

ISSN 2072-0297

МОЛОДОЙ УЧЁНЫЙ

МЕЖДУНАРОДНЫЙ НАУЧНЫЙ ЖУРНАЛ



50
ЧАСТЬ I
2020

16+

Молодой ученый

Международный научный журнал

№ 50 (340) / 2020

Издается с декабря 2008 г.

Выходит еженедельно

Главный редактор: Ахметов Ильдар Геннадьевич, кандидат технических наук

Редакционная коллегия:

Ахметова Мария Николаевна, доктор педагогических наук
Абдрасилов Турганбай Курманбаевич, доктор философии (PhD) по философским наукам (Казахстан)
Бердиев Эргаш Абдуллаевич, кандидат медицинских наук (Узбекистан)
Жураев Хусниддин Олтинбоевич, доктор педагогических наук (Узбекистан)
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
Авдеюк Оксана Алексеевна, кандидат технических наук
Айдаров Оразхан Турсункожаевич, кандидат географических наук (Казахстан)
Алиева Тарана Ибрагим кызы, кандидат химических наук (Азербайджан)
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
Игнатова Мария Александровна, кандидат искусствоведения
Искаков Руслан Маратбекович, кандидат технических наук (Казахстан)
Кайгородов Иван Борисович, кандидат физико-математических наук (Бразилия)
Калдыбай Кайнар Калдыбайулы, доктор философии (PhD) по философским наукам (Казахстан)
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
Кошербаева Айгерим Нуралиевна, доктор педагогических наук, профессор (Казахстан)
Кузьмина Виолетта Михайловна, кандидат исторических наук, кандидат психологических наук
Курпаяниди Константин Иванович, доктор философии (PhD) по экономическим наукам (Узбекистан)
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 (Казахстан)
Паридинова Ботагоз Жаппаровна, магистр философии (Казахстан)
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
Султанова Дилшода Намозовна, кандидат архитектурных наук (Узбекистан)
Титова Елена Ивановна, кандидат педагогических наук
Ткаченко Ирина Георгиевна, кандидат филологических наук
Федорова Мария Сергеевна, кандидат архитектуры
Фозилов Садриддин Файзуллаевич, кандидат химических наук (Узбекистан)
Яхина Асия Сергеевна, кандидат технических наук
Ячинова Светлана Николаевна, кандидат педагогических наук

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)
Буриев Хасан Чутбаевич, доктор биологических наук, профессор (Узбекистан)
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, декан (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)
Кошербаева Айгерим Нуралиевна, доктор педагогических наук, профессор (Казахстан)
Курпаяниди Константин Иванович, доктор философии (PhD) по экономическим наукам (Узбекистан)
Куташов Вячеслав Анатольевич, доктор медицинских наук, профессор (Россия)
Кыят Эмине Лейла, доктор экономических наук (Турция)
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)
Рахмонов Азиз Боситович, доктор философии (PhD) по педагогическим наукам (Узбекистан)
Ребезов Максим Борисович, доктор сельскохозяйственных наук, профессор (Россия)
Сорока Юлия Георгиевна, доктор социологических наук, доцент (Украина)
Узаков Гулом Норбоевич, доктор технических наук, доцент (Узбекистан)
Федорова Мария Сергеевна, кандидат архитектуры (Россия)
Хоналиев Назарали Хоналиевич, доктор экономических наук, старший научный сотрудник (Таджикистан)
Хоссейни Амир, доктор филологических наук (Иран)
Шарипов Аскар Калиевич, доктор экономических наук, доцент (Казахстан)
Шуклина Зинаида Николаевна, доктор экономических наук (Россия)

На обложке изображен Человек-невидимка — герой романа «Человек-невидимка» писателя Герберта Уэллса, написанный им в 1897 году.

В романе описывается судьба английского ученого-медика, а впоследствии физика Гриффина, который изобрел аппарат, делающий человека невидимым (и одновременно препарат, обесцвечивающий кровь). Правда, для полной невидимости человек должен был быть альбиносом, каковым Гриффин и был. Гриффин не хотел раньше времени обнародовать свое открытие, чтобы позже создать большую сенсацию. Однако обстоятельства сложились так, что из-за материальных затруднений он не мог продолжать свою работу. У него появилась идея «исчезнуть» и начать совершенно новую жизнь в качестве невидимки.

Уэллс раскрывает противоречия между развитием науки и техники и интересами человека в буржуазном обществе. Гриффин сам превращается в невидимку; будучи крайним индивидуалистом, равнодушным к судьбам человечества, он своим открытием не только не приносит пользы себе и другим, но ожесточается в борьбе со своими многочисленными врагами, увидевшими для себя опасность в загадочной и невидимой силе. Сосредоточенность героя на научных исследованиях ведет его к отчуждению от людей, нравственному огрублению, позволяет переступить через моральные запреты. Его чудесное открытие не приносит ему радости, люди начинают бояться невидимки. Гриффин погибает, становясь жертвой собственных экспериментов, непонятый и затравленный толпой гений-одиночка, забитый ею до смерти, проявившийся на глазах своих убийц. С ним погибает и его гениальное открытие. Гриффин, Человек-невидимка, бунтует против всего человечества именно потому, что материальные условия, в которых он находится, мешают закончить ему свои опыты. Эгоистом его сделала жизнь, среда, вечная

погоня за средствами для научной работы, отсутствие всякой поддержки, равнодушие и злоба окружающих.

Хотя невидимость в оптическом диапазоне — дело неопределенного будущего, текст Уэллса достаточно давно подвергается критике учеными. По мнению советского популяризатора науки Якова Перельмана, даже если бы удалось разработать практическую невидимость, такой человек был бы слеп. Если у невидимки показатель преломления тела становится равным аналогичному показателю у воздуха, хрусталик в этом случае потеряет возможность преломлять лучи света и фокусировать их на сетчатке глаза. Сама сетчатка также не сможет поглощать видимый свет палочками и колбочками по причине своей невидимости.

Научно-фантастические романы, к которым относится и роман «Человек-невидимка» (англ. *The Invisible Man*), принесли английскому писателю Герберту Уэллсу большую популярность. Это одно из знаменитейших произведений «научной фантастики» конца XIX века (Уэллс считается классиком научной фантастики, хотя термина такого в те времена не существовало, а сам Уэллс называл свои произведения «научными романами»). Задуманный еще в процессе работы над произведением «Машина времени» в 1896 году, «Человек-невидимка» был написан и печатался в июне-августе 1897 года в лондонском журнале «Пирсонз мэгэзин» и вышел отдельным изданием в сентябре того же года в Лондоне. Сразу после выхода роман имел огромный успех, причиной которого были как растущая известность Уэллса, так и оригинальность самого произведения. Эта фантастическая история, несколько раз переводимая на русский язык, пользуется огромной популярностью и в наши дни.

Екатерина Осянина, ответственный редактор

СОДЕРЖАНИЕ

АНОНСЫ

Отборы по программе «УМНИК» в рамках «Цифровой экономики»	1
---	---

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Антонов Д. А., Матюшенко А. А. Методика планирования процесса модернизации информационно-управляющей системы	3
Евстратов В. В. Создание программы визуализации псевдотрехмерного изображения с помощью рейкастинга	12
Ежов Г. В., Димитренко М. В. Алгоритмы нахождения пути, их сравнение и визуализация на базе движка Unity	15
Иванова М. А., Егорова Ю. А. Расширение функциональных возможностей в информационной системе «1С: Зарплата и кадры государственного учреждения»	24
Калашников Е. И., Иванова М. А. Процесс тестирования программного обеспечения, типы и методы тестирования	27
Кульмамиров С. А., Батыл Б. Пути исследования нечетких систем управления в среде fuzzyTECH Explorer Edition	31
Кунакбаева Э. Р. Основные проблемы информационной безопасности судов	39
Макушева О. Н., Щербинин Г. А. Компьютерные игры как источник образования	42
Санчес К. П., Карденас К. К. Перспективы проектирования бизнес-архитектуры в цепочке поставок с использованием технологии блокчейн	44

ТЕХНИЧЕСКИЕ НАУКИ

Абилжанова Н. Х., Абдухан А. Г. Исследование и совершенствование методики улучшения качества питьевой воды на станции водоподготовки из поверхностного источника	51
Авдеюк Д. Н. Способ повышения точности обработки сигнала рефлектометра	54
Боровикова А. С., Вдовина А. О., Медведева Л. Н. Разработка системы орошения на приусадебном участке в рамках проекта «Умный дом»	57
Золотов М. А. Способы воздействия на эффективность работы трехфазного сепаратора	59
Лёвина Е. В. Вероятностный анализ отказов газопроводов и оборудования, влияющих на надежность функционирования многоступенчатых распределительных систем	64
Малясов А. А. Пути разработки системы управления для беспилотного транспортного вертолета	66
Невыпрегайло И. Н. Опыт применения технологии бурения с контролем давления совместно с телесистемой с гидравлическим каналом связи при бурении горизонтальных скважин на Куюмбинском нефтегазовом месторождении	69
Радченко В. Е. Причины пожаров на предприятиях угледобывающей промышленности	71
Углова Е. В., Николенко М. А., Чирва Д. В., Саркисян Д. А., Керимов З. А., Оганян Ю. К. Концепция BIM-технологии при проектировании автомобильных дорог	72

А Н О Н С Ы

Отборы по программе «УМНИК» в рамках «Цифровой экономики»

Фонд содействия инновациям продолжает грантовую поддержку проектов в рамках национальной программы «Цифровая экономика Российской Федерации».

Прием заявок несколько раз в год идет на следующие конкурсы:

«УМНИК — Цифровая Россия» — всероссийский универсальный конкурс, направленный на поддержку проектов в области сквозных технологий цифровой экономики, реализуемый на базе пяти центров притяжения:

[Ростов-на-Дону](#) – для регионов Юга и Северного Кавказа;

[Москва](#) – для регионов Центральной России;

[Санкт-Петербург](#) – для регионов Северо-Запада;

[Казань](#) – для регионов Поволжья и Урала;

[Томск](#) – для регионов Сибири и Дальнего Востока.

Подать заявку можно на площадках конкурса.

[«УМНИК — VR/AR»](#) — всероссийский конкурс молодёжных инновационных проектов, которые создают и/или используют технологии виртуальной или дополненной реальности.

[«УМНИК — Цифровой нефтегаз»](#) — всероссийский конкурс молодёжных инновационных проектов по цифровому развитию нефтегазовой отрасли.

[«УМНИК — Электроника»](#) — всероссийский конкурс проектов, использующих сквозные цифровые технологии в области микроэлектронной промышленности, создания и развития электронной компонентной базы, устройств сенсорики и радиоэлектронной аппаратуры на ее основе.

[«УМНИК — Фотоника»](#) — всероссийский конкурс молодежных инновационных проектов в области фотоники, радиофотоники и оптоэлектроники.

Ожидается запуск традиционных конкурсов «УМНИК-Сбербанк», «УМНИК-МТС», «УМНИК – Цифровой прорыв», а также новых отборов совместно с ОАО «РЖД», ГК «Росатом» и другими партнерами.

Победа в конкурсе означает финансирование в размере 500 000 рублей для авторов инновационных проектов в возрасте 18-30 лет.

Следите за новостями Фонда на сайте fasie.ru

**ЖДЕМ ВАШИХ ПРОЕКТОВ
ДЛЯ УЧАСТИЯ В КОНКУРСЕ "УМНИК"
В РАМКАХ НАЦИОНАЛЬНОЙ ПРОГРАММЫ
"ЦИФРОВАЯ ЭКОНОМИКА РОССИЙСКОЙ
ФЕДЕРАЦИИ"!**

Победа в конкурсе означает финансирование в размере 500 000 рублей для авторов инновационных проектов в возрасте 18-30 лет.

Подробная информация - на сайте Фонда fasie.ru.

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Методика планирования процесса модернизации информационно-управляющей системы

Антонов Дмитрий Александрович, адъюнкт;
Матюшенко Алексей Алексеевич, кандидат военных наук, доцент
Военно-космическая академия имени А. Ф. Можайского (г. Санкт-Петербург)

Развитие или совершенствование информационно-управляющих систем (ИУС) неразрывно связано с комплексным использованием средств управления, отображения, документирования на базе современных компьютерных технологий. Перманентное совершенствование подобного рода средств неизбежно приводит к тому, что модернизируются пункты управления и связи между ними. Пункты управления объединяются в зависимости от характера связей между ними в организационные структуры ИУС, которые представляют собой их взаимосвязанную совокупность. К таким ИУС предъявляется целый ряд требований, в числе важнейших, обеспечивающих заданную эффективность их функционирования, считаются требования к технической надежности вычислительных средств пунктов управления, включая сюда и линии систем передачи данных и связи, а также в ряде случаев и требования к живучести указанных элементов и системы в целом. В статье представлены аналитические зависимости и методика планирования процесса модернизации ИУС с учетом допустимого расхода материальных и временных ресурсов. Сутью процесса модернизации является выбор рационального варианта структурного резервирования пунктов управления в ИУС при условии наличия ограничений на дополнительные финансовые и временные ресурсы. Предлагаемая методика позволяет выбрать рациональный вариант структурного резервирования до начала проведения модернизации, который обеспечит заданную эффективность функционирования ИУС и допустимый расход финансовых и временных ресурсов при её проведении. Задача этого типа относится к классу задач распределения неоднородного ресурса между потребителями.

На базе принципа декомпозиции ИУС на отдельные «конуса независимости», в статье в качестве основного метода решения задачи используется метод последовательного итерационно-циклического приближения к её рациональному решению.

Ключевые слова: организационная структура, задача распределения неоднородного ресурса, метод последовательного итерационно-циклического приближения к рациональному решению задачи, резервирующая связь, пункт управления, информационно-управляющая система, система передачи данных и связи

Введение

Вопросы повышения защищенности и надежности аппаратуры пунктов управления (ПУ) информационно-управляющей системы (ИУС), системы передачи данных (СПД) и связи, а также обслуживающего персонала рассматриваются в известных областях технических и военных наук и в существующей литературе отражены достаточно широко. Эффективность их решения характеризуется надежностными и стоимостными показателями. Решение задач структурного резервирования, хотя и позволяет обеспечить, как правило, требуемую живучесть системы управления, но в целом ряде случаев может быть экономически недоступным в силу ограниченных финансовых ресурсов. В этом случае решение задачи в целом представляется возможным за счет структурного резервирования ПУ без увеличения их числа в ИУС путем замены координационных или субординационных связей

в их иерархической организационной структуре резервирующими.

В большинстве случаев целесообразно и необходимо совершенствовать ИУС с линейной структурой, так как она обладает наибольшими резервами для повышения её эффективности при модернизации. Отчасти сказанное выше справедливо и для ИУС с функциональной структурой.

Решение задач повышения эффективности функционирования или живучести (защищенности и надежности) ИУС рассматривается в комплексе, исходя из существующих финансовых и временных ограничений. Задачам в подобной постановке, по мнению авторов, в существующей литературе не уделялось должного внимания. Сложность ее решения обусловлена тем, что в ней необходимо одновременно учитывать влияние затрат на структурное резервирование ИУС и затрат на повышение надежности аппаратуры ПУ ИУС, которые носят существенно нелинейный характер.

Математическая постановка задачи и выбор метода её решения

Эффективность применения современной ИУС определяется не только эффективностью функционирования каждого ПУ, но и зависит от того насколько они эффективно взаимодействуют между собой. Поэтому процесс совершенствования ИУС должен рассматриваться в комплексе всех мер по улучшению технической и программной оснащённости ПУ, изменению их функциональных связей. Вначале важно оптимально спланировать сам процесс модернизации ИУС и выбрать один из методов его организации: последовательный, параллельный и параллельно-последовательный. Выбор того или иного метода определяется необходимостью поддержания высокой готовности ИУС и директивным временем на её модернизацию.

Сутью процесса модернизации является выбор рационального варианта структурного резервирования пунктов управления в ИУС при условии наличия ограничений на дополнительные финансовые и временные ресурсы.

В данном случае под организационной структурой (организационной структурой) ИУС понимается вектор вида

$$\bar{S} = \langle P_i; P_{ij}; \delta_{ij}; \gamma_{ij} \rangle, \quad (1)$$

где P_i — показатель определяющий безошибочность и своевременность решения i -м ПУ задач управления; $i = \overline{1, v}$;

P_{ij} — показатель определяющий безошибочность и своевременность решения i -м ПУ задач управления j -го ПУ; $j = \overline{1, v}$; $j \neq i$;

$\delta = |\delta_{ij}|$ — матрица взаимного резервирования (назначения) ПУ ИУС, элемент которой $\delta_{ij} = 1$, если i -й ПУ в процессе решения задач управления резервирует j -й ПУ, а в противном случае $\delta_{ij} = 0$;

$\gamma = |\gamma_{ij}|$ — матрица коэффициентов участия, элемент которой γ_{ij} определяет степень участия

i -го ПУ во времени решения задач j -го ПУ.

Решение задачи следует рассматривать как поиск такого вектора $S^* \in S$, при котором эффективность $P(S^*)$ функционирования ИУС будет максимальна.

Учитывая это замечание, рациональный вектор организационной структуры ИУС может быть определен из выражения вида

$$P(S^*) \rightarrow \max_{S^* \in S} \quad (2)$$

$$\text{при условиях: } P_i^* \leq P_i^*; P_{ij}^* \leq P_{ij}^*; \quad (3)$$

$$C(\delta_{ij}^*) \leq C^0; |\delta_{ij}^*| \in |\delta_{ij}^0|; t(\delta_{ij}^*) \leq t^0,$$

где P_i^* — соответственно нижний предел показателя эффективности функционирования i -го ПУ заданного уровня иерархии ИУС;

P_{ij}^* — соответственно, нижний предел показателя эффективности функционирования i -го ПУ при решении задач управления j -го ПУ;

$|\delta_{ij}^0|$ — матрица назначения (допустимого для ИУС резервирования ПУ), определяемая характером их функциональных связей;

t^0 — временные ограничения на модернизацию ИУС;

C^0 — стоимостные ограничения на модернизацию ИУС;

$t(\delta_{ij}^*), C(\delta_{ij}^*)$ — соответственно, время и стоимость проведения мероприятий по модернизации ИУС.

В зависимости от характера решаемых задач модернизации ИУС возможны различные комбинации ограничений вида (3). Наиболее часто рассматриваются задачи выбора рациональной организационной структуры ИУС при условии наличия следующих ограничений:

1. При условии наличия ограничений на директивный срок t^0 модернизации организационной структуры ИУС.

2. При условии наличия стоимостных ограничений модернизации организационной структуры.

Возможны различные комбинации указанных задач. В практике планирования процессов модернизации ИУС нередко также встречаются обратные задачи минимизации временных и стоимостных затрат при условии, что показатели её эффективности будут не ниже требуемых значений [3,4,10,14].

Задачи первой группы возникают при необходимости оперативного повышения (сохранения) эффективности функционирования ИУС, связанной с периодическими или случайными изменениями состава ПУ. Поиск рациональной структуры ИУС производится при этом на основе использования имеющихся ресурсов, характеристики которых определяются матрицей $D = |\delta_{ij}|$.

Задачи второй группы возникают при необходимости долговременного планирования процесса совершенствования организационной структуры ИУС для достижения максимальной её эффективности функционирования. Задачи указанных групп используются, как правило, для долгосрочного оптимального планирования процесса совершенствования ИУС.

Результаты процесса совершенствования ИУС зависят от его организации. Если позволяет директивный срок проведения модернизации, выбирается последовательный метод её проведения.

Выбор этого метода определяется необходимостью поддержания высокой готовности ИУС.

Для множества ПУ и связей между ними удобно использовать матричную форму записи исходных данных и результатов расчетов.

Тогда в общем виде задачу можно сформулировать следующим образом. Примем известными:

– вероятностные характеристики живучести ПУ ИУС, в условиях воздействия внешних факторов в их точках дислокации являются элементами диагональной матрицы $\phi = |\phi_i|$, $\{i = \overline{1, v}\}$, где ϕ_i — вероятность выхода из строя i -го ПУ при воздействии внешних факторов;

– надежность аппаратуры отдельных ПУ (и связанных с ними участков линий СПД и связи)0, матрицей — строкой $A = |a_i|$, $\{i = \overline{1, v}\}$, где a_i характеризуют надежность (вероятность безотказной работы) i -го ПУ;

– стоимостные характеристики организации и проведения мероприятий по модернизации ИУС в виде квадратной матрицы $C = \|c_{ij}\|_{\text{кво}}$, $\{v = v; i, j = \overline{1, v(1, v)}\}$,

где

c_{ii} — стоимостные характеристики проведения мероприятий по увеличению вычислительных мощностей и подготовке резерва операторов на резервирующих (резервируемых) ПУ,

c_{ij} — стоимостные характеристики проведения мероприятий по организации резервирующей связи (СПД и связи) между i -м ПУ и j -м ПУ (или наоборот);

— допустимый уровень затрат, направляемых на модернизацию ИУС — C^0 .

Как правило, до проведения модернизации действующей ИУС известны трудозатраты (чел/час) на проведение мероприятий по её модернизации, что облегчает получение временных характеристик:

$$T = \|t_{ij}\|_{v \times v}, \{v = v; i, j = \overline{1, v(1, v)}\},$$

где t_{ii} — время проведения мероприятий по совершенствованию вычислительных средств, средств отображения и управления, по созданию и подготовке резерва операторов на резервирующих (резервируемых) ПУ,

t_{ij} — время проведения мероприятий по созданию (организации) резервирующей связи (СПД и связи) между i -м ПУ и j -м ПУ (или наоборот);

— директивное время, отпущенное на модернизацию ИУС — t^0 .

На практике допустимый уровень затрат и директивное время задаются вышестоящим органом управления.

Тогда эффективность решения комплексов задач управления ИУС P пунктами управления i с учетом показателей живучести и технической надежности ПУ, а также надежности линий СПД и связи будет равна

$$P = |(E - \phi)A + A(0)|, \tag{4}$$

где ϕ — диагональная матрица характеристик живучести ПУ ИУС;

A — матрица — столбец характеристик надежности ПУ;

$A(0)$ — матрица, в которой диагональные элементы равны нулю, а остальные элементы a_{ij} — характеристики надежности линий СПД и связи. Подобную матрицу в [7,8,9,12] называют матрицей отношений или связности;

E — единичная матрица.

В качестве «инструмента», для упрощения процесса решения поставленной задачи, используется квадратная матрица назначений $D = \|\delta_{ji}\|_{v \times v}; \delta_{ji} \in \{0,1\}$, которая позволяет получить искомым априорный план проведения модернизации.

Задачу повышения живучести и технической надежности ИУС целесообразно рассмотреть на примере одной из её подзадач (далее задачи): задачи структурного резервирования ПУ в ИУС (без увеличения их числа) при условии наличия ограничений на дополнительные финансовые и временные ресурсы. Решение этой задачи представляется возможным путем введения дополнительных резервирующих (координационных) связей в организационной структуре ИУС, а также путем повышения надежности аппаратуры ПУ ИУС.

Само понятие «резервирующая связь» между ПУ ИУС было впервые упомянуто в работах [1–6,9,12] для данной предметной области. Резервирующая связь между ПУ — это такая связь, которая обеспечивает решение 100% наиболее важных задач управления резервируемого ПУ резервирующим ПУ.

Сложность ее решения обусловлена тем, что необходимо одновременно учитывать влияние затрат на структурное резервирование ИУС и затрат на повышение надежности аппаратуры ПУ ИУС, которые носят существенно нелинейный характер [1–6,9,10,12,13,15].

Необходимо определить рациональный вариант решения рассматриваемой задачи — матрицу назначения $\delta^* \in D$ (априорный план модернизации организационной структуры), который обеспечивает максимальную эффективность функционирования ИУС — F при допустимых затратах средств и времени на её модернизацию.

Математически задача может быть представлена в следующем виде:

$$F = P \cdot D = \|p_{ij}\|_{v \times v} \cdot \|\delta_{ji}\|_{v \times v} \rightarrow \max; \delta^* \in D \tag{5}$$

при условии, что $C(\delta_{ij}^*) \leq C^0; t(\delta_{ij}^*) \leq t^0$,

где $D = \{\delta = \|\delta_{ji}\|_{v \times v}; \delta_{ji} \in \{0,1\}\}$, а $\delta^* \in D$ — искомая оптимальная матрица назначений, элементы которой могут принимать значения равные 1, если i -й ПУ резервируется j -м ПУ или 0, в противном случае. Искомая матрица назначений δ^* может быть получена методом последовательного итерационно-циклического приближения к рациональному решению путем последовательных итераций (этапов) при проведении модернизации ИУС.

В этом случае элементы матриц $\|p_{ij}\|$, $\|c_{ij}\|$ и $\|t_{ij}\|$ рассчитываются на момент времени t_z , то есть на моменты времени начала и конца каждого этапа (где индекс — z , это номер этапа). Используя t_{ij} , можно получить продолжительность 1 этапа проведения модернизации ИУС — ΔT_1 . Таким образом, $t_z = t_{z=0} + \Delta T_1$, где $t_{z=0}$ — время начала проведения 1 этапа модернизации. Очевидно, что продолжительности этапов проведения модернизации ИУС для каждого из этапов z зависят от выбранного направления модернизации и в общем случае различны. На практике их получают на основе ретроспективного анализа процессов создания, испытаний, эксплуатации и совершенствования организационной структуры модернизируемой или аналогичной ИУС.

Выбор возможного направления резервирования и, соответственно, резервирующего пункта управления $i(i = \overline{1, v})$, осуществляется таким образом, чтобы на каждом этапе модернизации обеспечивался бы максимум выражения (5). Итерационный процесс модернизации прекращается на таком шаге z , при котором выполняется одно из трех неравенств

$$P|_{t_z} \geq P^0; C|_{t_z} \geq C^0; t_z \geq t^0, \tag{6}$$

где $P|_{t_z}$ — эффективность функционирования ИУС на этапе z ;

$C|_{t_z}$ — расходы на проведение модернизации на этапе z ;

t_z — время на проведение модернизации на этапе z .

В матрицу $\delta^* |_{t_i}$ записывается её элемент $\delta_{ij}^* = 1$, соответствующий рассматриваемому направлению резервирования (модернизации) после каждой итерации. После завершения итерационного процесса получаем искомую матрицу назначений δ^* , а также затраты на реализацию каждого направления и время.

На базе принципа декомпозиции ИУС на отдельные «конуса независимости» за счет матрицы D уменьшается размерность матрицы $\|p_{ij}\|$, которая будет равна 2×2 . Это означает, что на первом этапе совершенствуются два ПУ и связи между ними. Выбор в этих парах ПУ и соответственно связей между ними означает наличие «1» на соответствующих позициях в матрице $\|\delta_{ji}\|$. Она одновременно указывает на элементы оргструктуры этой пары ИУС, которые модернизируются на данном этапе.

Пример. Оргструктура ИУС включает в себя три ПУ на одном уровне управления, показатели эффективности функционирования которых равны:

$$P = \begin{pmatrix} 0,9 & 0 & 0 \\ 0 & 0,7 & 0 \\ 0 & 0 & 0,89 \end{pmatrix}$$

Как видно, в составе ИУС имеется 2-й ПУ, который имеет низкую эффективность функционирования (например, по причине низкого уровня автоматизации). Связи между ПУ отсутствуют, так как оргструктура у ИУС линейная, то есть имеются только субординационные связи с ведущим ПУ (рис. 1). Совершенствуются ПУ нижнего уровня управления, которые свои задачи управления решают независимо друг от друга. Эффективность решения задач ИУС в целом до проведения модернизации тогда равна $P = \prod_{i=1}^3 P_i = 0,5607$. Требуемая эффективность функционирования ИУС $P_{\text{треб}} \geq 0,99$, а $P_{\text{треб ПУ}} \geq 0,95$.

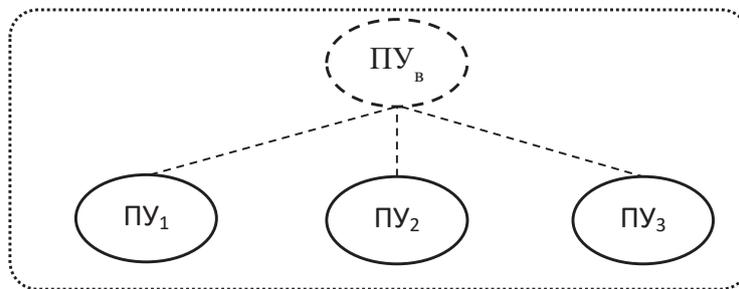


Рис. 1. Организационная структура ИУС до модернизации

Чтобы уменьшить размерность матриц P и D (число направлений совершенствования ИУС) на первом этапе модернизации вычеркиваем в них третий столбец и третью строку. Тем самым модернизируем на первом этапе первый и второй ПУ и связи между ними, что соответствует последовательному методу организации процесса.

Тогда матрица $P = \begin{pmatrix} 0,9 & 0 \\ 0 & 0,7 \end{pmatrix}$, а варианты матрицы $\|\delta_{ji}\|_{2 \times 2}$

соответствующие направлениям (вариантам) модернизации на первом этапе приведены в таблице № 1. Возможные расходы на модернизацию $C = \|c_{ij}\|_{\text{всв}}$ и время проведения $T = \|t_{ij}\|_{\text{всв}}$ считаем известными. Эти матрицы согласованы с матрицей P .

Таблица 1. Направления модернизации ИУС и их характеристики на первом этапе

№ п/п	Вид матрицы назначения D	Возможные направления модернизации	Вид матрицы P	Эффективность функционирования пары ПУ	Эффективность функционирования ИУС
1	$\delta_1^1 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$	Модернизация ВС и аппаратуры отображения и управления второго ПУ	$P = \begin{pmatrix} 0,9 & 0 \\ 0 & 0,95 \end{pmatrix}$	$p_{12} = p_1 \cdot p_2$ 0,855	$P_{\text{иус}} = p_1 \cdot p_2 \cdot p_3$ 0,76095
2	$\delta_2^1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	Введение двухсторонней резервирующей связи между первым и вторым ПУ	$P = \begin{pmatrix} 0,9 & a_{12} \\ a_{21} & 0,7 \end{pmatrix}$	$p_{12} = 1 - (1 - p_1)(1 - p_2)$ 0,97	$P_{\text{иус}} = p_{12} \cdot p_3$ 0,8633
3	$\delta_3^1 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$	Введение двухсторонней резервирующей связи между первым и вторым ПУ и модернизация второго ПУ	$P = \begin{pmatrix} 0,9 & a_{12} \\ a_{21} & 0,95 \end{pmatrix}$	$p_{12} = 1 - (1 - p_1)(1 - p_2)$ 0,995	$P_{\text{иус}} = p_{12} \cdot p_3$ 0,88555
4	$\delta_4^1 = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$	Введение двухсторонней резервирующей связи между первым и вторым ПУ и модернизация обоих ПУ	$P = \begin{pmatrix} 0,95 & a_{12} \\ a_{21} & 0,95 \end{pmatrix}$	$p_{12} = 1 - (1 - p_1)(1 - p_2)$ 0,9975	$P_{\text{иус}} = p_{12} \cdot p_3$ 0,947625

Анализ содержимого таблицы № 1 показывает, что наиболее эффективными являются направления модернизации (3 и 4), связанные с введением двухсторонней резервирующей связи между первым и вторым ПУ и модернизацией второго ПУ или обоих одновременно. Выбираем 4 направление.

На втором этапе модернизации вычеркиваем в матрицах P и D первый столбец и первую строку. Тем самым модернизируем на втором этапе третий ПУ и связи между ним и вторым

ПУ, при этом учитываем результаты первого этапа модернизации. Кроме этого, на втором этапе число направлений модернизации уменьшается до двух.

Тогда матрица $P = \begin{vmatrix} 0,9975 & 0 \\ 0 & 0,89 \end{vmatrix}$, а варианты матрицы

$\|\delta_{ji}\|_{2 \times 2}$ соответствующие направлениям (вариантам) модернизации на втором этапе приведены в таблице № 2.

Таблица 2. Направления модернизации ИУС и их характеристики на втором этапе

№ п/п	Вид матрицы назначения D	Возможные направления модернизации	Вид матрицы P	Эффективность функционирования пары ПУ	Эффективность функционирования ИУС
1	$\delta_1^2 = \begin{vmatrix} 0 & 1 \\ 1 & 0 \end{vmatrix}$	Введение двухсторонней резервирующей связи между вторым и третьим ПУ и модернизация третьего ПУ	$P = \begin{vmatrix} 0,9975 & a_{23} \\ a_{32} & 0,89 \end{vmatrix}$	$p_{23} = 1 - (1 - p_{12})(1 - p_3)$ 0,99973	$p_{иус} = p_{23} \cdot p_1$ 0,9497
2	$\delta_2^{2*} = \begin{vmatrix} 0 & 1 \\ 1 & 1 \end{vmatrix}$	Введение двухсторонней резервирующей связи между вторым и третьим ПУ и модернизация третьего ПУ	$P = \begin{vmatrix} 0,9975 & a_{23} \\ a_{32} & 0,95 \end{vmatrix}$	$p_{23} = 1 - (1 - p_{12})(1 - p_3)$ 0,99988	$p_{иус} = p_{23} \cdot p_1$ 0,9499

Анализ содержимого таблицы № 2 показывает, что эффективными являются оба направления модернизации (1 и 2), связанные с введением двухсторонней резервирующей связи между вторым и третьим ПУ и модернизацией третьего ПУ. Выбираем 2 направление.

На третьем этапе модернизации вычеркиваем в матрицах P и D второй столбец и вторую строку. Тем самым модернизируем на третьем этапе связи между третьим ПУ и первым ПУ (при этом оба ПУ в модернизации не нуждаются) и учитываем

результаты первого и второго этапов модернизации. Кроме этого, на третьем этапе число направлений модернизации

уменьшается до одного. Тогда матрица $P = \begin{vmatrix} 0,9975 & 0 \\ 0 & 0,99988 \end{vmatrix}$,

а вариант матрицы $\|\delta_{ji}\|_{2 \times 2}$ соответствующий направлению (варианту) модернизации на третьем этапе приведен в таблице № 3.

Таблица 3. Направления модернизации ИУС и их характеристики на третьем этапе

№ п/п	Вид матрицы назначения D	Возможные направления модернизации	Вид матрицы P	Эффективность функционирования пары ПУ	Эффективность функционирования ИУС
1	$\delta_1^{3*} = \begin{vmatrix} 0 & 1 \\ 1 & 0 \end{vmatrix}$	Введение двухсторонней резервирующей связи между первым и третьим ПУ	$P = \begin{vmatrix} 0,9975 & a_{13} \\ a_{31} & 0,99988 \end{vmatrix}$	$p_{13} = 1 - (1 - p_{12})(1 - p_{23})$ 0,999997	$p_{иус} = p_{13} \cdot p_{23} \cdot p_{12}$ 0,9974

Таким образом, требуемая эффективность функционирования ИУС для данного примера достигается на третьем этапе модернизации (рис. 2) за счет двух-кратного резервирования каждого ПУ двумя другими. Организационная структура ИУС соответствует матричной структуре [3,4].

В результате найдена рациональная структура связей резервирования между ПУ ИУС, при которой обеспечивается максимум эффективности функционирования ИУС при заданных

сроках модернизации её ПУ, а искомая матрица назначений (априорный план проведения модернизации) для данного примера имеет вид:

$$\delta_{иус}^* = \delta_4^{1*} \cup \delta_2^{2*} \cup \delta_1^{3*} = \begin{vmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{vmatrix}$$

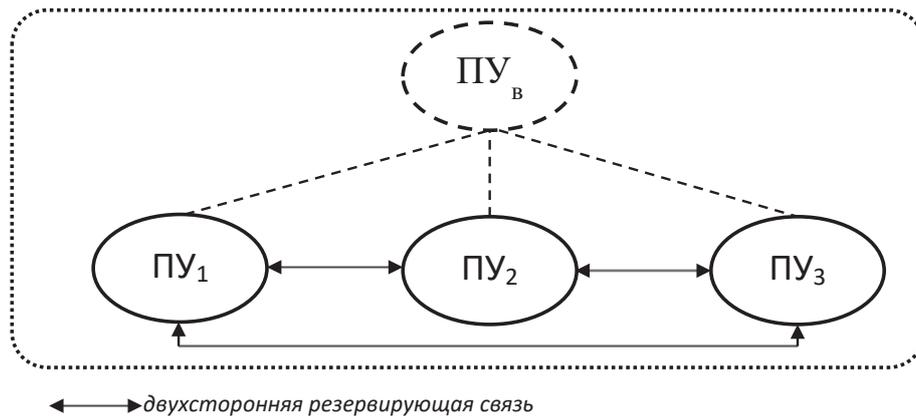


Рис. 2. Организационная структура ИУС после модернизации

Текстуально план кратко формулируется следующим образом:

- на первом этапе поочередно модернизируется первый и второй ПУ и параллельно создается резервирующая связь между ними;
- на втором этапе модернизируется третий ПУ и параллельно создается резервирующая связь между вторым ПУ и третьим ПУ;
- на третьем этапе создается резервирующая связь между первым ПУ и третьим ПУ.

Модернизация ИУС проводится в три этапа. Распределение поэтапно известных ресурсов $C = \|c_{ij}\|_{\text{xxx}}$ и $T = \|t_{ij}\|_{\text{xxx}}$ не представляет сложности. Однако не исключен вариант решения обратной задачи — минимизации расходов и времени проведения модернизации. План может быть отображен и в виде таблицы, обычного или сетевого графика.

Однако этот «красивый» результат может быть не получен, если недостаточно средств C^0 или времени t^0 на проведение модернизации. Поэтому лицу принимающему решение (ЛПР) следует обратить внимание на то, что если требуется глубокая модернизация с применением самых современных вычислительных средств и технологий для достижения максимально возможной эффективности функционирования ИУС, то и требуемый объем финансовых и временных ресурсов должен соответствовать поставленным целям.

Очевидно, что при реализации того или иного направления модернизации ИУС можно варьировать выделенными на это финансовыми и временными ресурсами. В процессе на каждом этапе модернизации можно уменьшить расходы за счет использования самых современных вычислительных средств и технологий, которые, как правило, дешевле и эффективнее тех, которые применялись ранее. Время проведения модернизации на каждом этапе можно также уменьшить за счет использования передовых технологий при монтаже, настройке и испытаниях вновь вводимых вычислительных средств, аппаратуры ПУ, систем ПД и связи, а также варьируя количеством и уровнем профессионализма специалистов. В случае необходимости можно кардинально уменьшить время, используя параллельный метод организации процесса модернизации.

Методика планирования процесса модернизации информационно-управляющей системы

Основными недостатками предлагаемого выше математического аппарата являются:

- высокая зависимость его от сложности организационной структуры ИУС;
- громоздкость выражений для ИУС со сложной структурой, затрудняющих их использование для проведения исследований и практических расчетов.

Однако первый недостаток легко устраним за счет декомпозиции сложной структуры на сравнительно простые двухуровневые организационные структуры, если есть необходимость учитывать финансовые и временные затраты, а также оценивать свойства ИУС, такие как живучесть и надежность. В этом случае применение этого математического аппарата вполне оправдано для проведения исследований.

На основе рассмотренного выше алгоритма и математического аппарата разработана схема методики планирования процесса модернизации ИУС, которая представлена на рис. 3.

Чтобы облегчить процесс исследования оргструктур в рамках методики, авторами была разработана *программно-реализованная модель оценки эффективности функционирования ИУС* с элементами визуализации. Она позволяет проводить оценку эффективности функционирования систем управления по трём группам показателей: вероятностным, временным и стоимостным, а результаты расчётов обладают достаточной точностью и достоверностью для того, чтобы на их основе можно было обосновать предложения по модернизации существующих ИУС.

На основе предлагаемого математического аппарата разработан алгоритм в [1] (приложение, п. 12), в соответствии с которым составлена программа модели на языке C++ Bilder (рис. 4).

Компьютерная модель имеет два «окна» (интерфейса) с элементами визуализации, используя которые можно исследовать и сравнивать между собой до восьми двухуровневых структур, так как результаты накапливаются и отображаются разными цветами.

В качестве *входных данных* для модели используются следующие переменные:

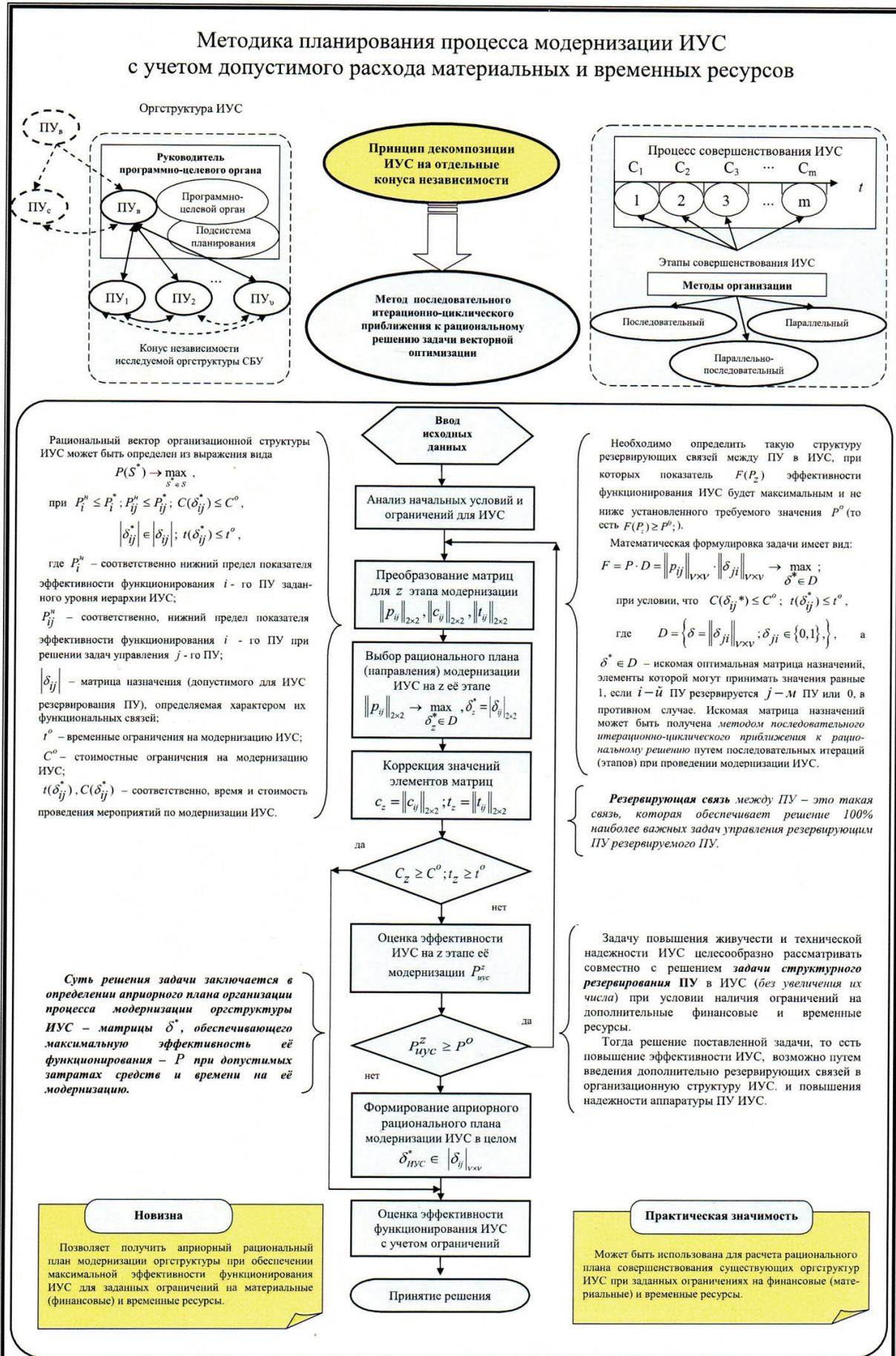


Рис. 3. Методика планирования процесса модернизации ИУС

1. Число ПУ на каждом уровне управления и матрица связей (связности) между ними;
2. Вероятности безошибочного и своевременного решения задач управления пунктами управления системы P_i ;
3. Вероятности безошибочной передачи информации каналами связи системы P_{ij} ;
4. Время решения задач управления пунктами управления T_{ny} , с;
5. Время передачи информации по каналам связи $T_{кс}$, с;
6. Стоимость создания (модернизации, эксплуатации) пунктов управления C_{ny} , усл.ед.;
7. Стоимость создания (модернизации, эксплуатации) каналов связи $C_{кс}$, усл.ед.;

В качестве **выходных данных** модели приняты следующие показатели:

1. Вероятность своевременного и безошибочного решения задач управления системой управления P_{ϕ} ;
2. Стоимость создания (модернизации, эксплуатации) системы управления C_{cy} , усл.ед.;

3. Время решения задач управления системой управления T_{cy} , с.
4. Эффективность функционирования системы управления \mathcal{E}_{fcy} .

Для упорядочения процесса расчётов в данной модели приняты следующие **допущения и ограничения**:

1. Вероятностные показатели задаются в числовой форме, в долях единицы, с точностью до сотых долей единицы: $P_i, P_{ij}, P_{\phi} \rightarrow [0...1]$;
2. Стоимостные показатели задаются в числовой форме, в количестве условных денежных единиц, с точностью до 1 у.е.: $C_{ny}, C_{кс}, C_{cy} \rightarrow [0...10^9 \text{ у.е.}]$;
3. Временные показатели задаются в числовой форме, в секундах, с точностью до секунды: $T_{ny}, T_{кс}, T_{cy} \rightarrow [0...10^6 \text{ с}]$;
4. Эффективность функционирования системы управления рассчитывается в процентах;
5. В организационной структуре системы управления допускается наличие субординационных, координационных и резервирующих связей;

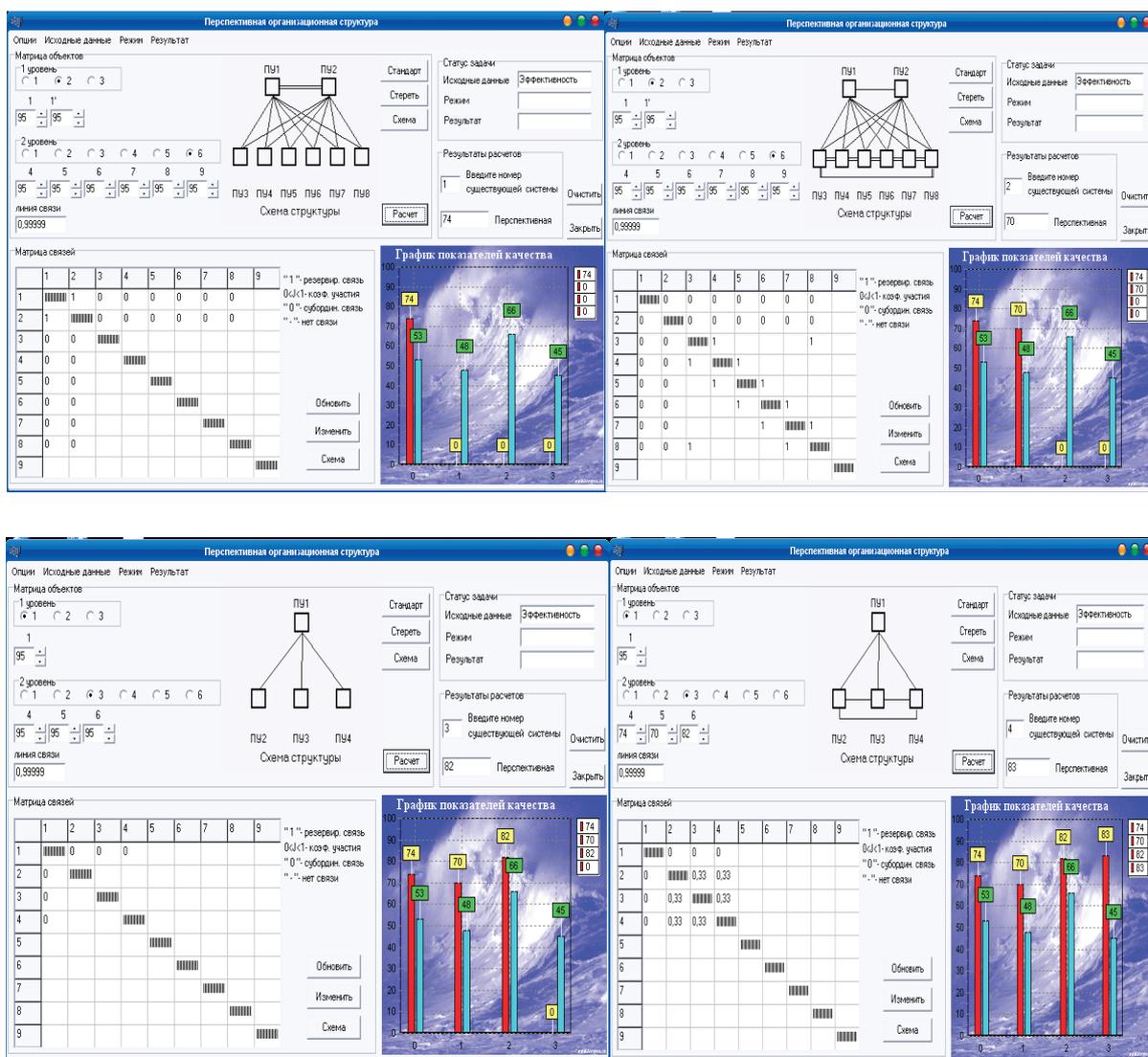


Рис. 4. «Окно» (интерфейс) модели с результатами оценивания эффективности функционирования ИУС для синтезируемых оргструктур (красного цвета) данные» и нажав кнопку «расчет» получаем оценку показателя в процентах

6. В организационной структуре системы управления допускается не более двух уровней управления;

7. Максимальное число пунктов управления в модели ограничено, не более чем по 9 ПУ на каждый уровень управления;

8. Для упрощения модели влияние окружающей среды на систему управления не учитывается, то есть функционирование системы зависит только от внутрисистемных процессов;

9. Все результаты расчётов округляются с точностью до сотых долей в сторону увеличения.

Во втором «окне» модели (в левом нижнем углу, рис. 3) в матрице связей (связности) есть возможность задавать тип связей между ПУ, то есть изменять организационную структуру исследуемой ИУС, которая отображается в верхней части «окна». «1» означает резервирующую связь. Новая организационная структура автоматически отображается в виде графа в *верхней центральной части* «окна». Затем выбрав оцениваемый показатель (в правом верхнем углу, — «исходные

На рисунке 4 зеленым цветом отображаются значения исходных показателей исследуемых структур ИУС.

Таким образом, разработанная модель в рамках предлагаемой методики значительно облегчает исследователю получение оценок эффективности функционирования систем

управления по трём группам показателей — вероятностным, временным и стоимостным, а результаты расчётов обладают достаточной точностью для того, чтобы на их основе можно было обосновать предложения по модернизации существующих систем управления. Кроме этого, отображаются в виде графа все исследуемые организационные структуры в соответствии с матрицей связности.

Заключение

В заключение следует отметить, что сформулированная выше, методика позволяет сочетать аналитические методы получения оценок показателей и результаты использования разработанной программно-реализованной компьютерной модели, что облегчает получение рационального априорного плана проведения модернизации (матрицу назначений) ИУС, обеспечивающего максимальную эффективность её функционирования. А также упрощает процесс распределения ресурсов, направленных на модернизацию (структурное резервирование ПУ) ИУС. Может использоваться для решения задачи выбора варианта структурного резервирования ПУ в процессе модернизации ИУС или создания перспективных.

Литература:

1. Матюшенко А.А. Модели и методы обоснования структур сложных систем ... управления. Монография. — СПб.: ВКА имени А. Ф. Можайского. 2015. — с. 86 ÷ 93 (н/с).
2. Матюшенко А. А., Хведелидзе Г.Э., Смирнов М.С. Методика формализации задач управления системы управления ... Труды ВКА им.А. Ф. Можайского, Третий выпуск, 641. Декабрь. — СПб.; ВКА имени А. Ф. Можайского, 2013. — с. 120–124.
3. Матюшенко А. А. и др. Особенности обоснования организационных структур при их многоструктурном анализе. — СПб.: Труды ВКА им. А. Можайского. Выпуск 630. Часть 2. Март 2011 г. — с 247–251. Инв. 335370.
4. Матюшенко А.А. Особенности оценивания показателей качества функционирования сложных систем управления для различных видов их организационных структур. Вестник Санкт-Петербургского университета МВД России № 1(33), январь 2007 г. — СПб.: УМВД, 2007.
5. Матюшенко А.А. Методика комплексного анализа организационных структур сложных систем управления. Известия Санкт-Петербургской лесотехнической академии. Выпуск 179. — СПб.: ГЛТА, 2007.
6. Матюшенко А.А. Методы комплексного анализа сложных систем управления ГПС МЧС. Вестник Санкт-Петербургского института государственной противопожарной службы. № 4, декабрь. — СПб.: ИГПС, 2006. — 12 с.
7. Системный анализ и принятие решений. Словарь — справочник. Учебное пособие для вузов/Под ред. В. Н. Волковой, В. Н. Козлова — М.: Высшая школа, 2004. — 616 с.
8. Владимирский Б. М., Горстко А. Б., Ерусалимский Я. М. Математика. Общий курс. Учебник для вузов. — СПб.: Издательство «Лань», 2002.
9. Кузнецов О.П., Адельсон-Вельский Г. М. Дискретная математика для инженера. — М.: Энергия, 1980.
10. Багрецов с. А., Везиров В.Н., Львов В.Н. и др. Технология синтеза организационных структур сложных систем управления. — М.: Академия оборонных отраслей промышленности, 1998. — 245 с.
11. Волкова В. Н., Денисов А. А. Основы теории систем и системного анализа: Учебник для студентов вузов. — СПб., Издательство СПбГТУ, 1997.
12. Шубинский И. Основы анализа сложных систем. Конспект лекций. ПВУРЭ ПВО, Пушкин — 1988.
13. Лагоша Б. А., Шаркович В. Г., Дегтярева Т. Г. Методы и модели совершенствования организационных структур. — М.: Наука, 1986.
14. Овсевич Б. Д. Модели формирования организационных структур. — Л.: Наука, 1979.
15. Справочник по исследованию операций./ Под общей редакцией Ф. А. Матвейчука — М.: Воениздат, 1979. с 102–107.
16. Михно М. К., Рыбаков Л. А. Формирование организационных структур АСУ. — М.: Экономика, 1977. — 86 с.

Создание программы визуализации псевдотрехмерного изображения с помощью рейкастинга

Евстратов Виталий Владимирович, студент
Камчатский государственный технический университет (г. Петропавловск-Камчатский)

В статье приводятся основные принципы и алгоритмы создания компьютерного изображения с помощью рейкастинга, а именно: графическое представление алгоритма броска луча и наложения текстур. Дана ссылка на полный исходный код проекта с инструкциями по его запуску.

Ключевые слова: raycasting, рейкастинг, псевдо 3D визуализация, graphic programming, графическое программирование, Wolfenstein.

Введение

Рейкастинг (англ. ray casting — бросание лучей) — один из методов рендеринга в компьютерной графике, при котором сцена строится на основе замеров пересечения лучей с визуализируемой поверхностью [1].

Термин впервые использовался в конце 70-х годов. Он применялся для описания метода параметрического моделирования с помощью комбинации примитивов, соединенных посредством операций объединения (+), разности (–) и пересечения (&).

В данной статье рейкастинг будет рассматриваться применительно к компьютерным играм. Конкретнее, будет рассмотрен способ отрисовки сцены в реальном времени в простейшей компьютерной игре.

Одной из первых игр, в которой была использована подобная технология псевдотрехмерной визуализации игрового пространства была Wolfenstein 3D (1992 год издания). Спустя почти 30 лет технологии визуализации в компьютерной графике, конечно, усложнились, однако, рейкастинг как технология имеет много общего с рейтрейсингом [2]. Рейтрейсинг, в свою очередь, благодаря увеличению вычислительных мощностей персональных компьютеров, в настоящее время становится современным способом визуализации некоторых объектов в играх, и уже довольно давно применяется в кинематографе для создания компьютерных спецэффектов.

Разработка своей собственной небольшой игры с применением технологии рейкастинга может помочь начинающим графическим программистам получить необходимый базовый опыт в создании компьютерных изображений и понимание

того, какие аспекты программы наиболее сильно влияют на её производительность.

Основной принцип рейкастинга в играх

Принцип рейкастинга заключается в том, что от наблюдателя (точки, из которой мы будем смотреть на сцену) бросаются лучи в необходимую область перед наблюдателем — область видимости. Каждый луч, столкнувшись с препятствием луч запоминает расстояние, которое он пролетел в данном направлении, прежде чем встретил препятствие. В зависимости от этого расстояния мы можем визуализировать это препятствие (стену) на экране. Размер визуализируемого объекта выбирается в соответствии с необходимой перспективой. На рисунке 1 показано графическое представление этого процесса: в области примерно 60° перед наблюдателем бросаются лучи, каждый луч пролетает какое-то расстояние, и чем оно больше — тем меньше высота отображаемой стены.

Далее будет описан общий алгоритм формирования таких изображений.

Соглашения для разработки собственной программы визуализации

- Представляемая сцена («карта») будет представлять собой двумерный массив символов. Символ «1» будет представлять препятствие, символ «0» — пустое пространство.
- Операционная система — Ubuntu.
- Для упрощенного создания графического окна воспользуемся надстройкой для Linux библиотеки LibX, MiniLibX.

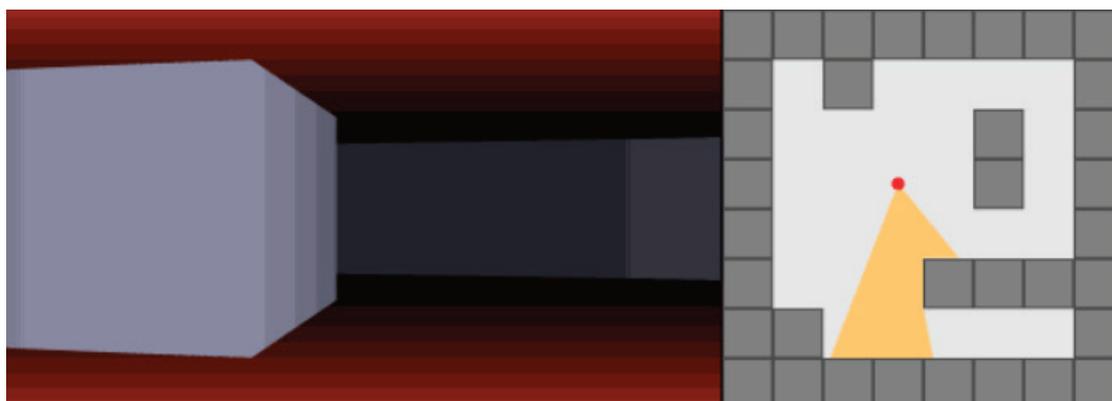


Рис. 1. Визуализация рейкастинга

Алгоритм отрисовки

Основной алгоритм отрисовки одной сцены заключается в следующем:

1. Определить ширину экрана в точках
2. Определить область видимости игрока в градусах (FOV — field of view)
3. Определить шаг угла (область видимости, деленная на ширину экрана)
4. Для каждого угла в интервале от 0 до FOV с полученным шагом выполнить бросок луча, и в зависимости от расстояния до препятствия отобразить вертикальную полосу на экране в соответствующей «колонке» дисплея

Основной алгоритм броска луча

Чтобы «бросить» луч необходимо вычислить расстояние между наблюдателем и препятствием в направлении луча. Если препятствие — это математический объект, который описывается каким-либо уравнением, а наблюдатель — точка на координатной плоскости, то чтобы рассчитать необходимое расстояние достаточно решить нужное уравнение. Однако, по нашим соглашениям препятствия — это «1» в двумерном массиве символов, поэтому нам нужно «шагать» лучом, пока мы не встретим стену.

Очевидным способом броска луча является перемещение начальной координаты луча в выбранном направлении на заданное константное расстояние [3] (рисунок 2).

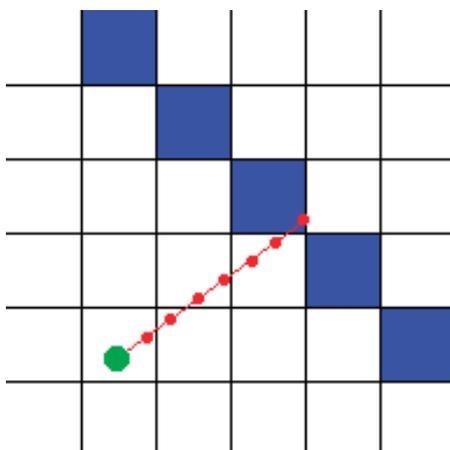


Рис. 2. Очевидное решение для броска луча

Однако, такое решение не является работоспособным для всех случаев. Основная причина в том, что под определенным углом луч будет проходить сквозь эту диагональную схему. Сокращение дистанции, на которую перемещается луч если и ча-

стично решает данную проблему, то вместе с этим сильно замедляет отрисовку изображения [3]. На рисунке 3 показан этот эффект.

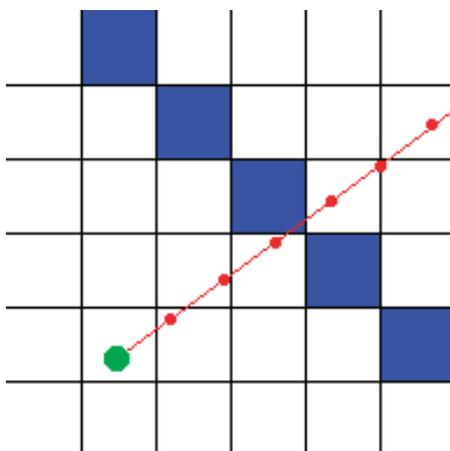


Рис. 3. Луч проскочил через стену

Правильным решением будет перемещать луч до следующей грани сетки. Это решение позволяет не нагружать отрисовку

долгими вычислениями, и в тоже время позволяет не промахнуться мимо стены. Визуализация представлена на рисунке 4.

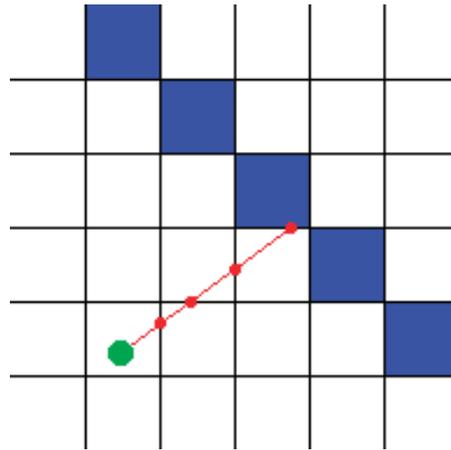


Рис. 4. Луч перемещается по граням сетки

Опираясь на эти идеи, мы можем реализовать алгоритм броска луча.

Примечание 1.

Перед отрисовкой стен мы можем нарисовать на экране линию горизонта и отделить «пол» от «потолка». Это поможет пользователям понимать, что верх и низ отличаются и сделать «игру» более красивой.

Примечание 2.

Поскольку мы проецируем изображение стены не на точку, из которой мы бросаем лучи, а на плоскость (монитор компьютера), то нам необходимо делать коррекцию расстояния до стен, чтобы убрать эффект рыбьего глаза. Необходимое расстояние до стены получается следующим образом:

$$\text{length_to_wall} = \text{ray_distance} * \cos(\text{cur_angle} - \text{middle_angle});$$

Здесь *cur_angle* — это угол текущей итерации относительно заданного FOV, а *middle_angle* — центральный угол FOV'a.

Примечание 3.

Для повышения эффективности алгоритма, вместо вычисления функции *cos()* во время отрисовки изображения можно заранее рассчитать таблицу косинусов и вместо вычисления косинусов брать заранее рассчитанное значение. Это может ускорить некоторые части программы в несколько раз [4].

Добавление текстур

Поскольку бросаемый луч останавливается на грани сетки, то мы можем узнать точку попадания луча относительно одной клетки. Вычислив относительное расстояние от левого края клетки до места попадания луча, мы можем узнать соответствующий столбец пикселей изображения и вывести его на экран. Перед выводом, очевидно, этот столбик пикселей текстуры необходимо растянуть или сжать (в зависимости от высоты изображения и высоты препятствия, которое мы отображаем). Графически этот процесс представлен на рисунке 5 [5].

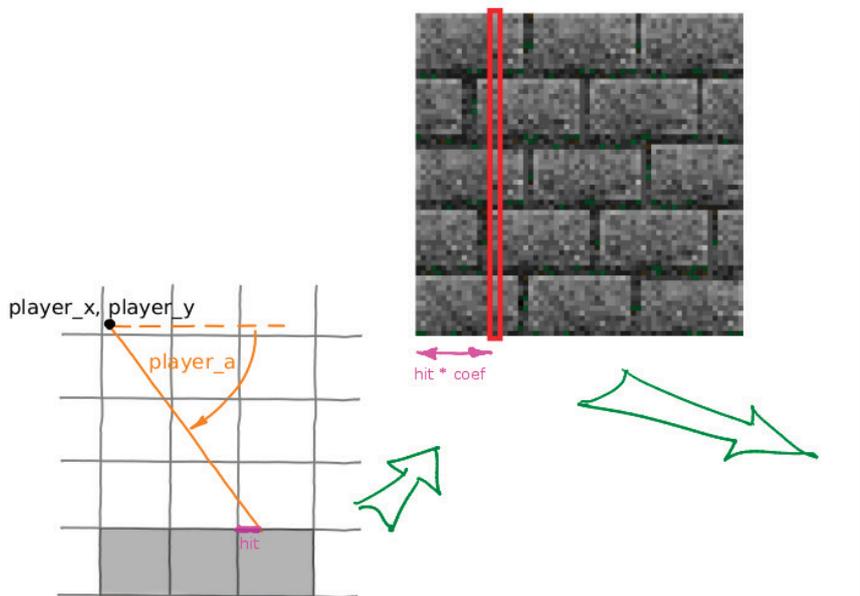


Рис. 5. Наложение текстур

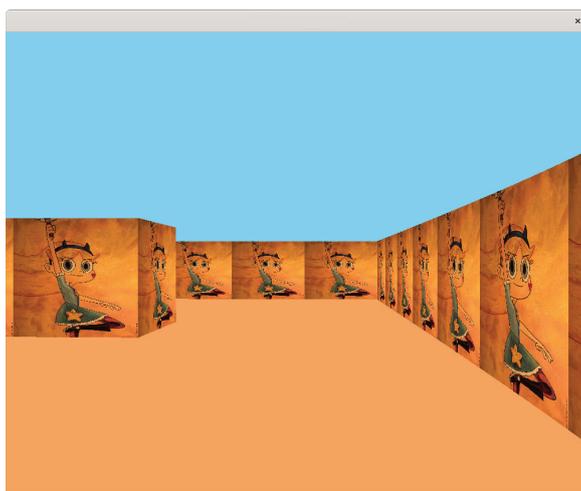


Рис. 6. Результат работы рейкастера

Заключение

Результат программы представлен на рисунке 6. Полный исходный код проекта можно посмотреть в github репозитории автора [<https://github.com/vesord/cub3d>].

Изучение принципов рендеринга компьютерного изображения является первым шагом на карьерном пути графического программиста.

Литература:

1. Рей кастинг. — Текст: электронный // Википедия свободная энциклопедия. — URL: https://ru.wikipedia.org/wiki/Ray_casting (дата обращения 05.12.2020)
2. Boulos, Solomon (2005). Notes on efficient ray tracing // SIGGRAPH 2005
3. Raycasting — Текст: электронный // Lode’s Computer Graphics Tutorial. — URL: <https://lodev.org/cgtutor/raycasting.html> (дата обращения 07.12.2020)
4. Евстратов В., Оценка скорости вычисления тригонометрических функций на Си / В. В. Евстратов // Молодой учёный № 46 (336) ноябрь 2020.
5. Рейкастинг — Текст: электронный // Блог компании «Библиотека программиста». — URL: <https://proglib.io/p/raycasting-for-the-smallest/> (дата обращения 08.12.2020)

Алгоритмы нахождения пути, их сравнение и визуализация на базе движка Unity

Ежов Глеб Владимирович, студент;

Димитренко Михаил Владимирович, Студент

Научный руководитель: Чернова Светлана Владимировна, старший преподаватель

Поволжский государственный университет телекоммуникаций и информатики (г. Самара)

В большом количестве прикладных задач, связанных с потребностью в оптимизации каких-либо ресурсов, есть возможность отображения начальных условий на граф. Вершины графа выступают в качестве обозначений контрольных пунктов или ключевых ресурсов, а рёбра графа могут подразумевать под собой не только расстояние, но и время, стоимость или объёмы ресурсов. Это делает графы универсальным инструментом для представления начальных условий в виде, который будет легко обрабатываться вычислительной машиной.

В таком случае появляется потребность в алгоритмах, которые смогут быстро получать оптимальный путь между двумя вершинами графа. Такой путь будет являться решением изначальной задачи. Алгоритмы, связанные с поиском оптимального пути между вершинами графа, названы алгоритмами поиска кратчайшего пути.

Путь в графе — это последовательность смежных вершин.

Кратчайший путь — это путь, который в результате вычисления его рёбер весовой функцией покажет наименьшее значение. Например, если все рёбра графа будут иметь единичный вес, то кратчайший путь — это наименьшее значение весовой функции.

В связи с изменчивостью постановки задачи, существует множество алгоритмов, каждый из которых создан для эффективного нахождения пути в определённых условиях. В этой статье мы рассмотрим три алгоритма, которые применяются особенно часто в решениях задач.

Алгоритм Breadth-First-Search (поиск в ширину):

Это один из методов полного обхода графа, путём последовательного перебора всех вершин. Сначала в очередь для проверки добавляется начальная вершина, а затем все смежные к ней. Далее при проверке каждой вершины, процесс добавления в очередь новых смежных вершин продолжается. То есть, перед переходом к проверке вершин на расстоянии $k+1$ от начала, сначала проверяются все вершины на уровне k .

Если представить работу алгоритма на графе, подобном по структуре дереву (см. рис. 1), то получим следующее:

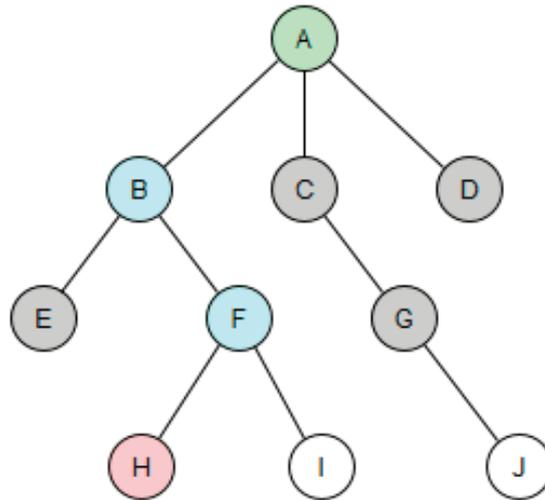


Рис. 1. Работа алгоритма «Поиск в ширину»

1. Добавление в очередь на проверку вершины A.
2. Проверка A и добавление в очередь вершин B, C и D.
3. Проверка B и добавление вершин E и F в конец очереди.
4. Проверка C и добавление в очередь G.
5. Проверка вершины D.
6. Проверка вершины E.
7. Проверка вершины F и добавление в очередь H и I.
8. Проверка G и добавление в очередь вершины J.
9. Проверка H и остановка алгоритма в случае наличия условия на остановку. Получили кратчайший путь: $A \rightarrow B \rightarrow F \rightarrow H$.

Главное преимущество этого алгоритма — это волновой обход всего графа, что помогает равномерно искать множество целей или одну цель, положение которой неизвестно. В то же время алгоритм не учитывает стоимость перехода от одной вершины к другой, поэтому он подойдёт для тех графов, рёбра которых не имеют веса.

Алгоритм поиска в ширину с прерыванием в случае нахождения конечной вершины на языке программирования Python:

```

frontier = Queue() # очередь для вершин графа
frontier.put(start )
came_from = dict() # словарь для информации о том, из какой
# вершины была получена текущая
came_from[start] = None

while not frontier.empty(): # алгоритм находится внутри тела
# цикла while
    current = frontier.get()

    if current == goal: # если текущая вершина конечная, то
# прервать цикл

```

```

for next in graph.neighbors(current): # цикл for,
заполняющий очередь и словарь новыми вершинами, относительно
текущей
    if next not in came_from:
        frontier.put(next)
        came_from[next] = current
    
```

Восстановление конечного пути:

```

current = goal
path = []
while current != start: # пока не будет достигнута начальная
вершина восстановить путь в массив
    path.append(current)
    current = came_from[current]
path.append(start) # добавить начальную вершину в массив
path.reverse() # выстроить элементы в последовательности от
начальной вершины до конечной
    
```

Алгоритм Дейкстры

Особенность этого алгоритма состоит в том, что, в отличие от «поиска в ширину», алгоритм Дейкстры учитывает «стоимость» перехода от одной вершины к другой, то есть каждое ребро графа получает свою цену. В качестве примера можно привести поиск кратчайшего дорожного пути с учётом автомобильного трафика.

Для учёта стоимости перехода от одной вершины к другой нужно модифицировать алгоритм поиска в ширину, добавив словарь для подсчёта стоимости пути к каждой из обнаруженных вершин, а также приоритетную очередь. С помощью приоритетной очереди алгоритм будет рассматривать самые выгодные маршруты в первую очередь, что позволит ему обходить «дорогостоящие» пути.

Рассмотрим работу алгоритма Дейкстры на графе (см. рис. 2):

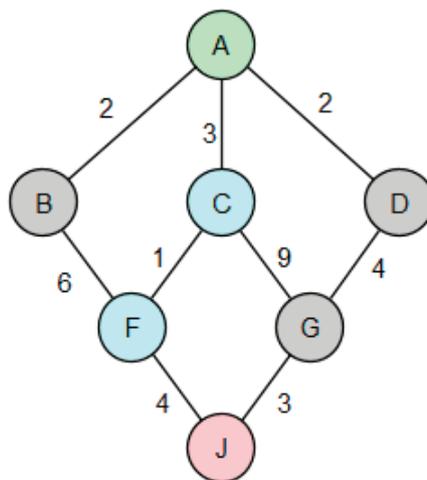


Рис. 2. Работа алгоритма Дейкстры

Приоритет очереди = цена. Чем меньше приоритет, тем выше вершина в очереди.

- 1) Добавление вершины A в очередь на проверку с ценой 0.
- 2) Проверка A и добавление в очередь вершин B (цена 2), C (цена 3) и D (цена 2).
- 3) Из очереди выбирается вершина с наименьшей стоимостью пути. В очереди стоит выше, чем D, поэтому проверяется вершина B. В очередь добавляется вершина F с общей ценой пути 8.
- 4) Далее в приоритете на проверку стоит вершина D. Проверка D, в очередь добавляется вершина G с общей ценой пути 6.
- 5) Проверка вершины C, в очередь добавляются вершины F и G с общими ценами 4 и 12 соответственно.
- 6) В текущей очереди находятся вершины F (цена 8), G (цена 6), F (цена 4) и G (цена 12). По правилу приоритетной очереди проверяется вершина F. В очередь добавляется вершина J с общей ценой 8.

- 7) Проверяется вершина G с ценой 6. В очередь добавляется ещё один путь до вершины J с общей ценой 9.
 8) Проверка J с ценой 8 и остановка алгоритма в случае наличия условия на остановку. Получили кратчайший путь: A → C → F → J.

Алгоритм Дейкстры с прерыванием в случае нахождения конечной вершины на языке программирования Python:

```
frontier = PriorityQueue() # очередь для вершин графа, где
высший приоритет получают вершины с наименьшей стоимостью пути
frontier.put(start, 0)
came_from = dict() # словарь для информации о том, из какой
вершины была получена текущая
cost_so_far = dict() # словарь, который хранит стоимость пути
от начала до обнаруженной вершины
came_from[start] = None
cost_so_far[start] = 0

while not frontier.empty(): # алгоритм находится внутри тела
цикла while
    current = frontier.get()

    if current == goal: # если текущая вершина конечная, то
прервать цикл
        break

    for next in graph.neighbors(current): # цикл заполнения
очереди и словарей новыми вершинами, относительно текущей
        new_cost = cost_so_far[current] + graph.cost(current,
next) # цена пути до обнаруженной вершины
        if next not in cost_so_far or new_cost <
cost_so_far[next]: # если это новая вершина или путь до этой
вершины выгоднее ранее найденного, то принять вершину к
рассмотрению и обновить стоимость пути до неё
            cost_so_far[next] = new_cost
            priority = new_cost
            frontier.put(next, priority)
            came_from[next]
```

Восстановление конечного пути:

```
current = goal
path = []
while current != start: # пока не будет достигнута начальная
вершина восстановить путь в массив
    path.append(current)
    current = came_from[current]
path.append(start) # добавить начальную вершину в массив
path.reverse() # выстроить элементы в последовательности от
начальной вершины до конечной
```

Минус алгоритма Дейкстры в том, что он тратит лишние ресурсы на проверку вершин, которые могут быть далеко от заранее известной цели и проверка которых нецелесообразна. Эту проблему решает следующий алгоритм.

Алгоритм A* (А звезда)

Этот алгоритм является модифицированной версией алгоритма Дейкстры. Помимо оценки пути с точки зрения стоимости переходов между вершинами, алгоритм A* учитывает положение цели, о которой известно заранее.

С практической точки зрения мы часто сталкиваемся с ситуацией, когда нам известно местоположение цели, поэтому алгоритм «А звезда» может оказаться эффективнее предыдущих двух, описанных выше.

Рассмотрим работу алгоритма «А звезда» на графе (см. рис. 3):

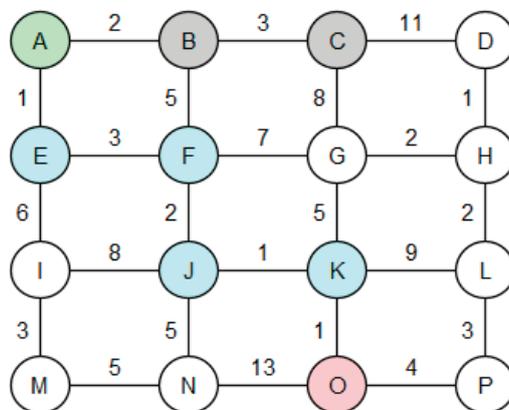


Рис. 3. Работа алгоритма «А звезда»

Приоритет очереди = цена + удалённость от цели. Чем меньше приоритет, тем выше вершина в очереди.

Удалённость от цели = $|X \text{ текущей вершины} - X \text{ цели}| + |Y \text{ текущей вершины} - Y \text{ цели}|$.

Координата цели (вершина O): (2, -3).

1) Проверка A (приоритет 0). Добавление в очередь вершин B (цена 2, удалённость 4, приоритет 6) и E (цена 1, удалённость 4, приоритет 5). Текущая очередь: E(5), B(6).

2) Проверка E (приоритет 5, получена из A). Добавление в очередь вершин F (цена 4, удалённость 3, приоритет 7) и I (цена 7, удалённость 3, приоритет 10). Текущая очередь: B(6), F(7), I(10).

3) Проверка B (приоритет 6, получена из A). Добавление в очередь вершин C (цена 5, удалённость 3, приоритет 8) и F (цена 7, удалённость 3, приоритет 10). Текущая очередь: F(7), C(8), I(10), F(10).

4) Проверка F (приоритет 7, получена из E). Добавление в очередь G (цена 11, удалённость 2, приоритет 13) и J (цена 6, удалённость 2, приоритет 8). Текущая очередь: C(8), J(8), I(10), F(10), G(13).

5) Проверка C (приоритет 8, получена из B). Добавление в очередь G (цена 13, удалённость 2, приоритет 15) и D (цена 16, удалённость 4, приоритет 20). Текущая очередь: J(8), I(10), F(10), G(14), G(15), D(20).

6) Проверка J (приоритет 8, получена из F). Добавление в очередь K (цена 7, удалённость 1, приоритет 8) и N (цена 11, удалённость 1, приоритет 12). Текущая очередь: K(8), I(10), F(10), N(12), G(14), G(15), D(20).

7) Проверка K (приоритет 8, получена из J). Добавление в очередь L (цена 16, удалённость 2, приоритет 18) и O (цена 8, удалённость 0, приоритет 8). Текущая очередь: O(8), I(10), F(10), N(12), G(14), G(15), L(18), D(20).

8) Проверка вершины O (приоритет 8, получена из K). Остановка алгоритма в случае наличия условия на остановку. Получили кратчайший путь: $A \rightarrow E \rightarrow F \rightarrow J \rightarrow K \rightarrow O$.

Функция, описывающая близость текущего местоположения и конечного (используется далее в коде алгоритма):

```
def heuristic(a, b):
    # Дистанция до цели в виде целого числа на графе,
    # представленном в виде сетки
    return abs(a.x - b.x) + abs(a.y - b.y)
```

Алгоритм A* с прерыванием в случае нахождения конечной вершины на языке программирования Python:

```
frontier = PriorityQueue() # очередь для вершин графа, где
# высший приоритет получают вершины с наименьшей суммой стоимости
# пути и дистанции до цели
frontier.put(start, 0)
came_from = dict() # словарь для информации о том, из какой
# вершины была получена текущая
```

```
cost_so_far = dict() # словарь, который хранит стоимость пути
от начала до обнаруженной вершины
came_from[start] = None
cost_so_far[start] = 0

while not frontier.empty(): # алгоритм находится внутри тела
цикла while
    current = frontier.get()

    if current == goal: # если текущая вершина конечная, то
прервать цикл
        break

    for next in graph.neighbors(current): # цикл заполнения
очереди и словарей новыми вершинами, относительно текущей
        new_cost = cost_so_far[current] + graph.cost(current,
next) # цена пути до обнаруженной вершины
        if next not in cost_so_far or new_cost <
cost_so_far[next]: # если это новая вершина или путь до этой
вершины выгоднее ранее найденного, то принять вершину к
рассмотрению и обновить стоимость пути до неё
            cost_so_far[next] = new_cost
            priority = new_cost + heuristic(goal, next)
            frontier.put(next, priority)
            came_from[next] = current
```

Восстановление конечного пути:

```
current = goal
path = []
while current != start: # пока не будет достигнута начальная
вершина восстановить путь в массив
    path.append(current)
    current = came_from[current]
path.append(start) # добавить начальную вершину в массив
path.reverse() # выстроить элементы в последовательности от
начальной вершины до конечной
```

Итог к разбору алгоритмов

Три этих алгоритма являются одними из наиболее часто применяемых способов нахождения оптимального пути в практических ситуациях. Конечно, алгоритм «А звезда» выделяется среди них особенно. Вбирая в себя технику алгоритма Дейкстры и, при этом, учитывая расстояние до известной цели, он позволяет быстро и точно находить кратчайший путь в чётко заданных условиях. Тем не менее, алгоритм «Поиск в ширину» и алгоритм Дейкстры также находят своё применение.

Алгоритм «Поиск в ширину» не только закладывает основной метод перебора вершин графа, но и сам по себе имеет практическое значение. Этим алгоритмом очень удобно находить одну или несколько целей, положение которых неизвестно заранее, в ситуациях, когда вес рёбер графа не имеет значения. Алгоритм Дейкстры схож в применении с алгоритмом «Поиска в ширину», при этом позволяет оценивать стоимость переходов между вершинами графа.

Алгоритмы нахождения кратчайшего пути являются важной частью дискретной математики и компьютерных наук в целом. Они позволяют находить путь не только в прямом понимании (кратчайший путь на картах внутри смартфонов и GPS-навигаторов), но и в значении наиболее выгодного пути, как это происходит в экономических задачах.

Визуализация на базе движка Unity

В качестве дополнения к подробному разбору алгоритмов нахождения кратчайшего пути приводим способ реализации одного из них на базе движка Unity. Реализация алгоритма на базе движка позволит автоматически перемещать игровые объекты на сцене от начальной позиции к конечной, учитывая наличие препятствий.

Визуализация алгоритма «Поиск в ширину» в сцене движка Unity:

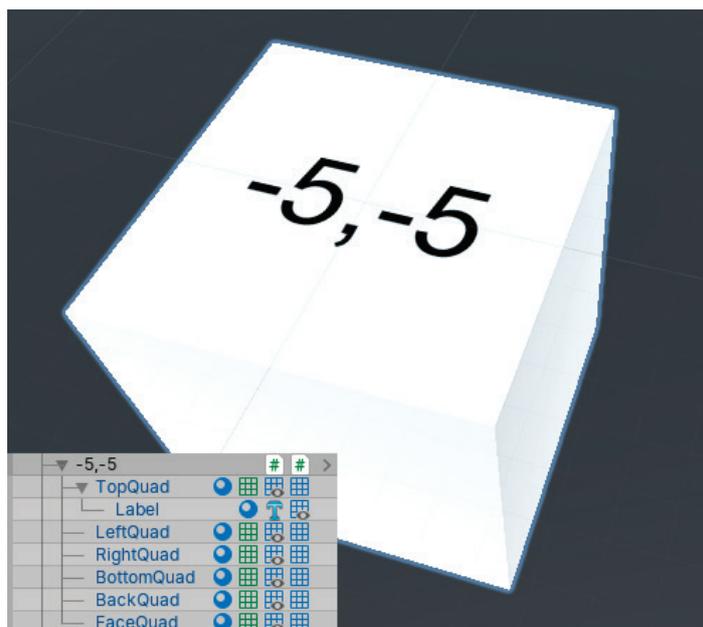


Рис. 4. Куб, выступающий в роли вершины

1. В качестве вершин у нас будут выступать кубы, составленные из четырёхугольников. «Label» на верхнем четырёхугольнике будет хранить в себе компонент Text Mesh для вывода положения вершины в пространстве (см. рис. 4). Mesh Renderer верхнего четырёхугольника будет окрашиваться в цвета, показывая его состояние в результате работы алгоритма.

2. К каждому кубу будет прикреплён скрипт Waypoint, который может возвращать положение куба в сетке, а также хранить его состояние в ходе работы алгоритма. Waypoint.cs:

```
using UnityEngine;

public class Waypoint : MonoBehaviour
{
    public bool isExplored = false;
    public Waypoint exploredFrom;

    const int gridSize = 10;
    Vector2Int gridPos;

    public int GetGridSize()
    {
        return gridSize;
    }

    public Vector2Int GetGridPos()
    {
        gridPos.x = Mathf.RoundToInt(transform.position.x / gridSize);
        gridPos.y = Mathf.RoundToInt(transform.position.z / gridSize);
        return new Vector2Int(gridPos.x, gridPos.y);
    }

    public void SetTopColor(Color color)
    {
        MeshRenderer topMeshRenderer =
transform.Find("TopQuad").GetComponent<MeshRenderer>();
        topMeshRenderer.material.color = color;
    }
}
```

3. Все кубы-вершины находятся внутри пустого объекта «Grid». Этот объект принимает скрипт Pathfinder, в котором заключается алгоритм нахождения кратчайшего пути «Поиск в ширину». Pathfinder.cs:

```
using System.Collections.Generic;
using UnityEngine;

public class Pathfinder : MonoBehaviour
{
    [SerializeField] Waypoint startWaypoint, endWaypoint;
    Dictionary<Vector2Int, Waypoint> grid = new Dictionary<Vector2Int, Waypoint>();
    Queue<Waypoint> queue = new Queue<Waypoint>();

    bool isRunning = true;
    Waypoint searchCenter;

    Vector2Int[] directions = {
        Vector2Int.up,
        Vector2Int.right,
        Vector2Int.down,
        Vector2Int.left
    };

    private void Start()
    {
        LoadBlocks();
        BreadthFirstSearch();
        ColorPath();
    }

    private void BreadthFirstSearch()
    {
        queue.Enqueue(startWaypoint);

        while (queue.Count > 0 && isRunning)
        {
            searchCenter = queue.Dequeue();
            HaltIfEndFound();
            ExploreNeighbours();
            searchCenter.isExplored = true;
        }
    }

    private void HaltIfEndFound()
    {
        if (searchCenter == endWaypoint)
        {
            isRunning = false;
        }
    }

    private void ExploreNeighbours()
    {
        if (!isRunning) { return; }

        foreach (Vector2Int direction in directions)
        {
            Vector2Int neighbourCoordinates = searchCenter.GetGridPos() + direction;
            if (grid.ContainsKey(neighbourCoordinates))
            {
                QueueNewNeighbours(neighbourCoordinates);
            }
        }
    }

    private void QueueNewNeighbours(Vector2Int neighbourCoordinates)
    {
        Waypoint neighbour = grid[neighbourCoordinates];
```

```

if (!neighbour.isExplored && !queue.Contains(neighbour))
{
    neighbour.SetTopColor(Color.blue);
    queue.Enqueue(neighbour);
    neighbour.exploredFrom = searchCenter;
}
}

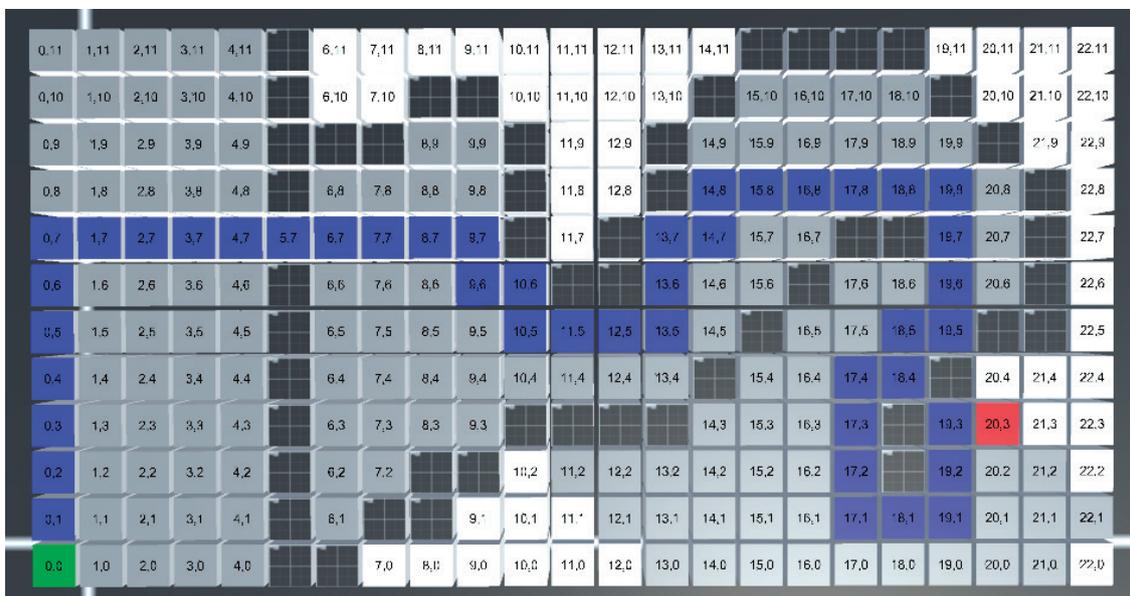
private void LoadBlocks()
{
    Waypoint[] waypoints = FindObjectsOfType<Waypoint>();
    foreach (Waypoint waypoint in waypoints)
    {
        if (grid.ContainsKey(waypoint.GetGridPos()))
        {
            Debug.LogWarning("Обнаружен перекрытый блок.");
        }
        else
        {
            grid.Add(waypoint.GetGridPos(), waypoint);
        }
    }
}

private void ColorPath()
{
    ColorTheTrail();
    startWaypoint.SetTopColor(Color.green);
    endWaypoint.SetTopColor(Color.red);
}

private void ColorTheTrail()
{
    Waypoint paintedWaypoint = searchCenter.exploredFrom;
    while (paintedWaypoint != startWaypoint)
    {
        paintedWaypoint.SetTopColor(Color.grey);
        paintedWaypoint = paintedWaypoint.exploredFrom;
    }
}
}

```

4. Дублируем кубы-вершины, расставляем их в виде сетки и выбираем в полях скрипта Pathfinder начальную и конечную вершину. После запуска сцены видим визуальное отображение работы алгоритма.



Расширение функциональных возможностей в информационной системе «1С: Зарплата и кадры государственного учреждения»

Иванова Мария Александровна, студент магистратуры;
Егорова Юлия Александровна, студент магистратуры
Калужский государственный университет имени К. Э. Циолковского

Данную статью авторы посвятили теме доработки типового решения «1С: Зарплата и кадры государственного учреждения». Приведено краткое описание существующего функционала. Произведен анализ причин, по которым зачастую требуется доработать стандартное решение. Рассмотрен пример из экзаменационного билета для специалистов 1С.

Ключевые слова: информационная система, 1С, государственное учреждение, документ, секретность, расширение функционала, пользователь.

Информационные системы затронули все сферы деятельности человека. Важные вопросы автоматизации касаются и таких сфер, как бухгалтерская и кадровая.

Из практики известно, что с увеличением объема данных повышается вероятность появления в информационных системах недостоверных данных, не соответствующих действительности, которые при использовании могут привести к негативным последствиям (сбой системы, нарушение технологического процесса, выход из строя оборудования, принятие неоптимальных управленческих решений и т.д.) [1]. В связи с этим в последние годы внутри государственных учреждений используются различные информационные системы для кадрового и бухгалтерского учета.

Создание и последующая поддержка таких информационных систем является одним из приоритетных направлений деятельности современной работы ведомств государственных учреждений РФ [2]. Такой точки зрения придерживается и фирма «1С», которая активно ведет разработку и обновление программных продуктов, в частности информационную систему «1С: Зарплата и кадры государственного учреждения». Технические разработки должны соответствовать законодательству РФ [5], а также поддерживать возможность выгрузки данных для контролирующих органов (Например: для Федеральной налоговой службы). В дальнейшем данные страхователей обрабатываются в автоматизированных информационных системах ведомств.

Однако, в настоящее время, вопросы проектирования систем управления базами данных автоматизации учёта взаиморасчетов с сотрудниками, работа которых предусматривает секретность, мало освещены в теоретических исследованиях и практических разработках, что определяет необходимость проведения исследований в этой области.

Из практического опыта было выявлено, что данная проблема сложна в решении на конкретных организациях, различающихся специфическими особенностями, в частности: имеющимся программным обеспечением «1С: Зарплата и кадры государственного учреждения». Для того, чтобы дорабатывать существующее решение программисты 1С проходят специальную сертификацию [6]: Аттестацию «1С: Специалист» по прикладному решению «1С: Зарплата и кадры государственного учреждения 8».

Рассмотрим следующий пример [6]: оплата труда (денежное содержание) работников, состоящих на государственной службе, состоит из следующих начислений: должностного оклада, ежемесячного денежного поощрения (устанавливается в количестве окладов), оклада за классный чин.

Дополнительно сотрудникам, работающим с секретными документами, выплачивается специальная доплата (за секретность). Доплата начисляется в процентах к денежному содержанию только за время фактической работы с секретными документами. Одновременно один и тот же работник может работать с разными секретными документами и получать суммарную доплату одним видом начисления.

Размер доплаты зависит от степени секретности и вида информации, которые представлены в таблице 1 [6].

Таблица 1. Показатели расчета секретности

Степень секретности	Государственная тайна	Служебная тайна
Секретные	60%	50%
Сов. секретные	80%	60%
Особой важности	100%	70%

Основные задачи:

1. Реализовать учет работы с секретными документами.
2. Создать настраиваемый отчет «Доплата за секретность», позволяющий за произвольное количество расчетных периодов получать следующую информацию (начальная настройка отчета должна соответствовать образцу из таблицы 2 [6]):

Таблица 2. Шаблон разрабатываемого отчета

	Месяц1				Месяц2			
	Гос. тайна		Служебная тайна		Гос. тайна		Служебная тайна	
	Дней	Доплата	Дней	Доплата	Дней	Доплата	Дней	Доплата
<Работник1> — всего				
...Степень секретности1				
Степень секретности2								

Очевидно, что в задании требуется создать конкретные документы-регистраторы периодов работы с секретными сведениями, поэтому требуется доработка типового решения.

Для создания требуемых начислений необходимо установить нужную настройку программы, как представлено на рис. 1.

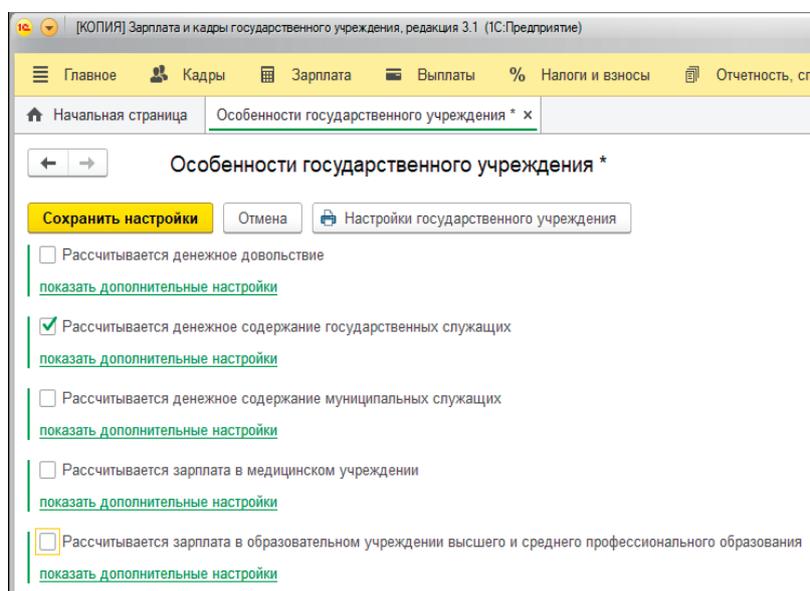


Рис. 1. Настройка особенностей государственного учреждения

Для реализации задачи потребовалось создать следующие объекты в конфигурации [3]:

- Подсистема «Секретность» (представлена на рис. 2)

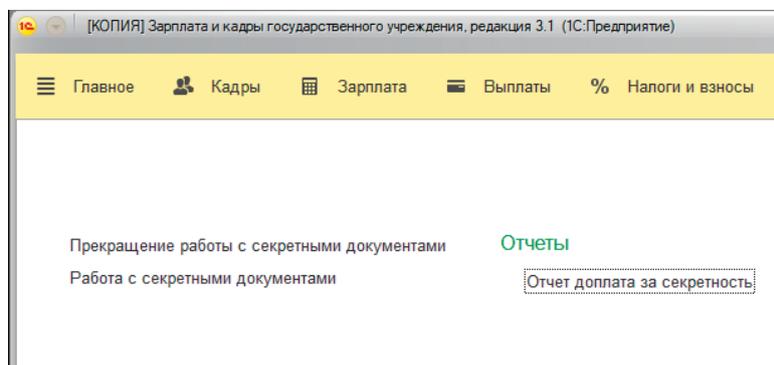


Рис. 2. Объекты подсистемы «Секретность»

- Справочник «Степени секретности», в котором будут храниться процентные показатели из таблицы 1
- Перечисления «Виды информации» со значениями Государственная тайна, Служебная тайна

- Документ «Работа с секретными документами» — для регистрации периодов привлечения работников к работам с секретными документами. Документ регистрирует степень секретности, вид информации и даты начала и окончания периода работы. Предусмотрена возможность ввода одного документа на несколько сотрудников сразу.
- Документ «Прекращение работы с секретными документами» — для регистрации досрочного прекращения доступа к секретным документам.
- Отчет «Доплата за секретность» (представлен на рис. 3)

Сотрудник	Ноябрь 2018				Декабрь 2018				Март 2019			
	Гос.тайна		Служебная тайна		Гос.тайна		Служебная тайна		Гос.тайна		Служебная тайна	
	Дней	Доплата	Дней	Доплата	Дней	Доплата	Дней	Доплата	Дней	Доплата	Дней	Доплата
Абаполова Дарья Сергеевна					5			4				
Секретные					3							
Сов. секретные					2							
Особой важности							4					
Иванов Иван Иванович			12	36 000								
Особой важности			12	36 000								
Кабанова Ирина Сергеевна	26	21 333,33	15	10 000	21	20 000	21	14 000	20	20 000	20	14 000
Секретные	7	4 000										
Сов. секретные	4	3 047,62										
Особой важности	15	14 285,71	15	10 000	21	20 000	21	14 000	20	20 000	20	14 000

Рис. 3. Настроенный отчет

На основе введенных документов «Работа с секретными документами» и «Прекращение работы с секретными документами» проводится автоматический расчет [4] доплаты за секретность по итогу месяца, что подтверждает рис. 4.

N	Сотрудник	Подразделение	Назначение	Результат	Финансир., расходы	Период	Основание
1	Кабанова Ирина...	Младший медицинский персонал	Должностной оклад	20 000,00	Б* (211)	01.12.2018 - 31.12.2018	
2	Кабанова Ирина...	Младший медицинский персонал	Надбавка за секретность	14 000,00	Б* (211)	01.12.2018 - 31.12.2018	Работа с секретными документами 0000000007 от 20.11.2018
3	Кабанова Ирина...	Младший медицинский персонал	Надбавка за секретность	20 000,00	Б* (211)	01.12.2018 - 31.12.2018	Работа с секретными документами 0000000008 от 20.11.2018

Рис. 4. Автоматический расчет надбавки на основе созданных документов

Таким образом, рассмотрены причины необходимости разработок в существующих информационных системах. Рекомендуется обучить пользователей таких программных продуктов работать с новым механизмом, а также разработать инструкцию доработанной подсистемы.

Литература:

1. Аверченков, В.И. Организационная защита информации: учебное пособие для вузов / В.И. Аверченков, М.Ю. Рытов.— Брянск: Брянский государственный технический университет, 2012.— 184 с.— Текст: электронный.
2. Смирнова, Е.В. Построение коммутируемых компьютерных сетей / Смирнова Е.В., Баскаков И.В., Пролетарский А.В., Федотов Р.А.— 2-е изд.—: НОУ «Интуит», 2016.— 428 с.— Текст: электронный.
3. Профессиональная разработка в системе «1С: Предприятие 8» / В.А. Ажеронков [и др.].— 2-е изд.— Москва: 1С-Паблишинг, 2012.— 2800 с.— Текст: непосредственный.

4. Радченко, М. Г. 1С: Предприятие 8.3. Практическое пособие разработчика. Примеры и типовые приёмы / М. Г. Радченко, Е. Ю. Хрусталева. — Москва: 1С-Паблишинг, 2013. — 952 с. — Текст: непосредственный.
5. КонсультантПлюс. — Текст: электронный // Справочно-правовая система «КонсультантПлюс»: [сайт]. — URL: <http://www.consultant.ru/>
6. Примеры билетов. — Текст: электронный // Фирма «1С»: [сайт]. — URL: <https://1c.ru/spec/questions.htm>

Процесс тестирования программного обеспечения, типы и методы тестирования

Калашников Егор Игоревич, студент магистратуры;
Иванова Мария Александровна, студент магистратуры
Калужский государственный университет имени К. Э. Циолковского

Тестирование программного обеспечения является наиболее важной фазой жизненного цикла разработки программного обеспечения. Тестируемое программное обеспечение проходит различные фазы, которые, согласно исследованию, являются тестовым анализом: планирование тестирования; подготовка тестового примера данных, среды; выполнение теста; регистрация ошибок; отслеживание и закрытие;

В прошлом было проведено много исследований для оптимизации всего процесса тестирования с целью улучшения качества программного обеспечения за минимальное время. После оценки всех доступных процессов тестирования было обнаружено, что для разных типов приложений используются разные модели разработки, и для тестирования используются разные методы тестирования. На основе исследований, проведенных в ходе изучения этой статьи, было проанализировано, что каждая компания модифицирует свой процесс тестирования в соответствии с потребностями и выполняет тестирование в зависимости от критичности приложений. Наиболее важные компоненты каждого приложения должны быть тщательно протестированы, чтобы убедиться, что их функции, производительность и функции безопасности работают должным образом. В этой статье говорится об обеспечении качества всех типов программных приложений путем выполнения определенных типов методов тестирования и оптимизированных процессов тестирования программного обеспечения. В соответствии с исследованиями и проведенными исследованиями типы тестирования могут быть разделены на три основных метода тестирования: функциональное тестирование, тестирование производительности и безопасности, а также основной процесс тестирования программного обеспечения, называемый анализом, подготовкой, выполнением и закрытием.

Ключевые слова: функциональное тестирование, тестирование производительности и безопасности (FPS), анализ, планирование и подготовка, выполнение и закрытие (APEC), методы тестирования программного обеспечения, жизненный цикл тестирования программного обеспечения (STLC), жизненный цикл разработки программного обеспечения (SDLC).

Software testing process, types and testing methods

Software testing is the most critical phase of the Software Development Life Cycle. Software under test goes through various phases, which as per the study are test analysis: test planning, test case data, environment preparation, test execution, bug logging and tracking and closure.

There is lot of research which has been done in past to optimize overall testing process with intent of improving quality of software in a minimum amount of time. After evaluating all available testing processes, it has been found that different development models are used for different types of applications and different testing techniques are performed to test the same. Based on the research during the study of this paper, it has been analyzed that each company modifies their testing process as per the needs and performs testing based on the criticality of the applications. The most critical components of each application have to be tested thoroughly to ensure their functional, performance and security features are behaving as expected. This paper talks about ensuring the quality of all types of software applications by performing certain types of testing techniques and optimized software testing processes. As per the study and research done testing types can be categorized under three major testing techniques which are Functional, Performance and Security Testing and major software testing process called as Analysis, Preparation and Execution and closure

Keywords: Functional, Performance and Security Testing (FPS), Analysis, Planning and Preparation, Execution and Closure (APEC), Software Testing Techniques, Software Testing Life Cycle (STLC), Software Development Life Cycle (SDLC).

Эта статья состоит из двух разделов. В первом разделе описывается оптимизированный процесс тестирования, в ко-

тором рассматриваются все этапы жизненного цикла тестирования, а во втором разделе рассматриваются типы тестирования.

Существует множество доступных типов тестирования, таких как тестирование «черного ящика», тестирование «белого ящика», тестирование на основе состояния, тестирование безопасности, тестирование на приемлемость, приемочное тестирование, системное тестирование, альфа- и бета-тестирование, а также тестирование на основе конфигурации, проверка и валидация.

Основываясь на проведенных исследованиях и исследованиях, эта статья классифицировала их всех по трем типам тестирования высокого уровня, а именно: Функциональность, Производительность и Безопасность (FPS). Последний раздел посвящен заключению, которое показывает актуальность нашего оптимизированного процесса тестирования программного обеспечения и FPS как основа для методов тестирования.

Оптимизированный процесс тестирования программного обеспечения

Этапы STLC связаны с обнаружением и исправлением любой ошибки с использованием различных методов тестирования программного обеспечения.

В этом разделе представлены обязательные этапы жизненного цикла тестирования, без которых жизненный цикл программного обеспечения не будет выполнен эффективно. Тестирование в основном дает критику или сравнение, которое определяет поведение системы в отношении ее спецификаций, механизмов, принципов, характеристик и соответствующих стандартов. Процесс тестирования программного обеспечения может быть настроен в соответствии с потребностями клиента или проекта. Процесс оптимизации, который можно использовать при тестировании программного обеспечения, — это анализ, планирование и подготовка, выполнение и закрытие. Процесс программного обеспечения обеспечивает поток системы и повышает гарантию продукта, который будет произведен.

Этап анализа тестирования

Первый этап, который является этапом анализа, является основным этапом процесса тестирования программного обеспечения. Эта фаза включает анализ функциональных и нефункциональных требований, например, бизнес-требования, функциональная документация и документ технической спецификации и т.д.

Сбор требований должен быть согласован с клиентами, чтобы идентифицировать фактические и ожидаемые результаты тестирования, которые в основном являются нефункциональными требованиями, такими как: удобство использования, масштабируемость, тестируемость, производительность и безопасность.

Все требования, которые не могут быть проверены из-за ограничений системы и среды тестирования, должны быть сообщены бизнес-команде.

На этом этапе команда тестирования рассматривает и анализирует требования и определяет тесты, которые должны быть выполнены, и устанавливает приоритеты для тестирования — членов команды.

Требования к среде тестирования включают требования к аппаратному и программному обеспечению, в соответствии с которыми необходимо тестировать необходимое программное обеспечение, и параллельно разработчики программного обеспечения начинают с планирования и разработки.

Этап планирования и подготовки теста

Этап подготовки теста включает подготовку плана тестирования, тестовый набор, данные теста и подготовку среды тестирования.

План тестирования является первым документом, который должен быть подготовлен, в котором описываются объем, цели, характеристики, подлежащие тестированию, функции, которые не должны тестироваться, типы тестирования, которые должны быть выполнены, роли и обязанности группы тестирования, критерии входа и выхода и предположения [1, с. 122].

Одновременно команды тестирования начинают готовить тестовые наборы и тестовые данные. Тестовый пример — это документ, в котором описаны шаги, необходимые для тестирования любой функциональности с ожидаемым и фактическим результатом. Если фактический результат не совпадает с ожидаемым, обнаруживается ошибка. Для каждого требования подготавливаются положительные и отрицательные тестовые примеры, что обеспечивается матрицей прослеживаемости требований (RTM). RTM — это документ, который сопоставляет требования с тестовыми примерами, чтобы гарантировать, что тестирование выполнено на 100%

Все действительные и недействительные наборы тестовых данных должны быть подготовлены для каждого тестового случая и подготовлен документ с тестовыми данными. Тестовые данные также генерируются на основе некоторых алгоритмов и инструментов. Подготовка тестового набора включает в себя различные этапы, которые начинаются с создания тестового набора, выбора тестового набора, оценки и определения приоритетов тестового набора [7, с. 57].

Эрик Д. Свайн предложил метод генерации тестовых случаев с использованием соответствующих диаграмм последовательности, а также определяет ограничения для определенных артефактов. В то же время методы генерации тестовых наборов полезны для выявления синхронизации и зависимости вариантов использования и сообщений, взаимодействия объектов и ошибок работы [8, с. 87].

Подготовка тестовой среды является одним из наиболее важных этапов, которые обычно готовятся отдельными средами обработки команды. После завершения части кода код проверяется инструментом управления конфигурацией, а затем готовится тестовая сборка, где тестировщики должны начать выполнение теста.

Этап выполнения теста

На этом этапе тестеры выполняют программное обеспечение в соответствии с контрольными примерами. Везде, где фактические и ожидаемые результаты не совпадают, тестировщик открывает ошибки и назначает их разработчикам.

Закрытие теста

Закрытие теста — это важный этап, который включает в себя все отчеты о тестировании, гарантирующие, что вся система, интеграция, приемочное тестирование пользователя пройдены, и принято решение о том, все ли требования проверены, и нет критической ошибки, ожидающей исправления ИЛИ проверки. Просмотр всех тестовых артефактов выполняется менеджером. Как только все артефакты рассмотрены и утверждены, выполняется выпуск программного обеспечения. Проводится дальнейший анализ первопричин, чтобы провести мозговой штурм по поводу того, что прошло хорошо, что не удалось, и областей улучшения. Существуют различные инструменты и методы анализа первопричин, по которым в прошлом было проведено много исследований.

Виды тестирования программного обеспечения

Существуют различные методы тестирования программного обеспечения согласно исследованию и изучению, такие как черный ящик, белый ящик, серый ящик, регрессия, надежность, удобство использования, производительность, модуль, система, интеграция, безопасность, дым, здравомыслие и объектно-ориентированное тестирование и т.д.

Невозможно выполнить все виды тестирования на программном обеспечении, поскольку для тестирования всегда выделяется определенное количество времени.

Функциональное тестирование очень распространено, и в прошлом было проведено немало исследований, поэтому только в редких случаях происходит сбой сайта из-за отсутствия функционального тестирования.

Эта раздел статьи будет посвящена основным методам тестирования, таким как функциональное [F], производительность [P] и тестирование безопасности [S]. Правильное сочетание тестирования должно быть включено со всех заголовков F, P и S.

Функциональность — это прежде всего аспект тестирования программного обеспечения, обеспечивающий качество программного обеспечения.

Проверка и валидация выполняются с использованием статического и динамического тестирования соответственно. Статическое тестирование включает в себя все виды проверок, проверок и прохождений. Динамическое тестирование или фактическая проверка включает все функциональные и нефункциональные типы тестирования.

Функциональное тестирование

Основным фактором качества в программном обеспечении является соответствие его требуемым функциональным возможностям и поведению. Функциональная часть программного обеспечения включает внешнее поведение, которое в основном определяет все требования пользователя. Разработка программного обеспечения высокого уровня производится таким образом, чтобы заказчик был удовлетворен на ранней стадии проектирования и разработки.

Существуют различные типы функционального тестирования, которые могут выполняться на различных уровнях тестирования, то есть модульное тестирование, интеграционное тестирование (тестирование сверху вниз и тестирование снизу вверх) и тестирование системы.

Существует много тестов, проводимых на различных уровнях тестирования, таких как тестирование черного ящика, тестирование белого ящика, тестирование серого ящика, регрессионное тестирование, нечеткое тестирование, тестирование варианта использования, предварительное тестирование, тестирование на дым., тестирование работоспособности, приемочное тестирование, альфа-тестирование, бета-тестирование и т.д.

Основные типы функционального тестирования кратко описаны в таблице 1:

Таблица 1. Описание типов функционального тестирования

№	Тип тестирования	Описание
1	Модульное тестирование	Самый низкий уровень тестирования в основном выполняется разработчиком для тестирования модуля кода
2	Интеграционное тестирование	Это делается для проверки связи между различными модулями, чтобы убедиться, что данные правильно передаются между различными компонентами. Это делается по принципу «сверху вниз» или «снизу вверх».
3	Тестирование системы	Система тестируется в целом, чтобы убедиться, что она ведет себя или функционирует так, как задумано и указано в документе с требованиями. Регрессионное тестирование проводится, чтобы убедиться, что в системе ничего не сломано после исправления ошибок и тестирования ошибок. Общее тестирование Smoke and Sanity проводится, чтобы убедиться, что все ссылки и функции работают и среда стабильна.
4	Тестирование белого ящика и черного ящика	Тестирование черного ящика проводится для того, чтобы убедиться, что результат приложения является правильным для всех различных типов положительных и отрицательных входных данных. Существуют различные типы типов тестирования «черного ящика», такие как разделение по классам эквивалентности, анализ граничных значений, определение ошибок и т.д. Белый ящик имеет дело с внутренней обработкой кода, чтобы гарантировать отсутствие избыточного кода, написанного в программном обеспечении. Это включает в себя тестирование строки кода, программы, потока, логики, цикла, структуры, функций, тестирования взаимодействия классов и другого внутреннего тестирования программы.

Тестирование производительности

Это один из типов нефункционального тестирования, который тестирует производительность программного обеспечения при всех благоприятных и неблагоприятных условиях. Включает в себя все связанные со временем параметры, такие как время загрузки, время доступа, время выполнения, время выполнения и т.д. Сюда также входят частота успеха, частота отказов, среднее время между отказами и общая надежность программного обеспечения.

Наиболее популярными типами тестирования, выполняемыми в тестировании производительности, являются стресс-тестирование и нагрузочное тестирование [9, с. 155].

Стресс-тестирование проводится, чтобы найти и понять верхние пределы пропускной способности в системе. Чрезвычайная нагрузка дается приложению для определения надежности системы.

Нагрузочное тестирование называется испытанием на выносливость. Это тестирование проводится, чтобы определить, может ли система выдерживать постоянную ожидаемую нагрузку.

Тестирование безопасности

Согласно текущим сценариям наиболее частыми являются атаки по времени и переполнению буфера. В объектно-ориентированных системах проблемы уровня проектирования включают обработку ошибок. Некоторые другие проблемы на уровне проектирования, такие как проблемы совместного использования и доверия, незащищенные каналы передачи данных, неправильные или отсутствующие механизмы контроля доступа, отсутствие аудита, неправильное ведение журналов и времени и ошибки порядка, также приводят к угрозам безопасности.

Программное обеспечение требуется для проверки функций безопасности, таких как строгая аутентификация, криптография и контроль доступа, а также некоторых других механизмов безопасности.

Гэри МакГроу предполагает, что уязвимость — это ошибка, которую злоумышленник может использовать [5, с. 45].

Тестирование безопасности программного обеспечения важно для защиты информации, услуг, навыков и ресурсов злоумышленников, а также стоимости потенциальных средств защиты [9, с. 243].

Тестирование безопасности в основном следует двум типам подходов:

Тестирование программного обеспечения относительно функциональных механизмов программного обеспечения;

Выполнение подхода, основанного на риске в соответствии с мышлением злоумышленников.

Тест на проникновение — это тестирование безопасности, в ходе которого оценщики пытаются обойти функции безопасности системы, основываясь на своем понимании структуры и реализации системы.

Еще один тип, который называется Fuzz Testing, был дан Бартоном Миллером из Висконсинского университета в 1988 году. Это методика тестирования программного обеспечения, при которой автоматически выдаются неверные, случайные и неожиданные данные для определения реакции программного обеспечения. Это хорошо для тестирования того программного обеспечения, где входные данные не имеют контроля над предопределенными данными. Этот метод тестирования используется только для поиска простых функций программного обеспечения, но не сложного программного кода.

Теперь, чтобы подвести итог вышеприведенным разделам тестирования, в таблице 2 приведены все типы тестирования.

Таблица 2. Описание всех типов тестирования

Тип тестирования	Метод
Функциональное тестирования	Черный ящик, Белый ящик, тестирование всех пар, таблицы переходов состояний, таблицы решений, тестирование на основе моделей, тестирование на основе использования, предварительное тестирование, тестирование на основе спецификаций, регрессионное тестирование, тестирование дыма, тестирование работоспособности.
Тестирование производительности	Нагрузка, стресс, выдержка и тестирование конфигурации.
Тестирование безопасности	Статический анализатор, атака методом грубой силы, SQL-инъекция и межсайтовый скриптинг (XSS), тестирование на проникновение и тестирование Fuzz.

Вывод

Целью данной работы было исследование различных этапов жизненного цикла тестирования программного обеспечения и различных типов тестирования. После анализа различных этапов жизненного цикла программного обеспечения выяснилось, что существуют четыре основных этапа жизненного цикла тестирования, которые можно отнести к категории «Анализ, планирование и подготовка, выполнение и закрытие». В данном документе предлагается общий

жизненный цикл тестирования программного обеспечения — АТЭС.

Поскольку всегда есть область применения, в дополнение к этой статье можно провести исследование в области тестирования программного обеспечения, чтобы предложить общую среду и методы тестирования для поддержки функционального тестирования, тестирования производительности и безопасности, для объектно-ориентированной среды разработки и других платформ, использующих некоторые алгоритмы без использования инструментов в минимальные сроки.

Литература:

1. Прессман, Р. С. Программная инженерия: практический подход / Р. С. Прессман.— 4-е изд.— 1997.— Текст: электронный.
2. Соммервилль, М. Разработка программного обеспечения / М. Соммервилль.— 5-е изд.— 1998.— Текст: электронный
3. Майерс, Г. Д. Искусство тестирования программного обеспечения / Г. Д. Майерс.— Текст: электронный.
4. МакГроу, Г. Программа защиты программного обеспечения / Г. МакГроу.— 2004.— Текст: электронный.
5. МакГроу, Г. Анализ рисков в разработке программного обеспечения / Г. МакГроу.— 2004.— Текст: электронный.
6. Бертолино, А. Практический подход к выводу интеграционных тестов на основе UML / А. Бертолино, Ф. Басаньери.— Текст: электронный // В материалах четвертой Международной недели качества программного обеспечения в Европе и Международной недели качества Интернета в Европе (QWE).— Брюссель, Бельгия., 2000.
7. Богдады, П. Предлагаемый метод генерации тестовых наборов на основе диаграмм действий / П. Богдады, Н. Бадр, Л. Хашем, М. Толба.— 2011.— Текст: электронный.
8. Свайн, Э. Д. Генерация тестовых примеров на основе сценария использования и последовательности / Э. Д. Свайн.— 2010.— Текст: электронный.
9. Бейзер, Б. Методы тестирования программного обеспечения. Технологическое созревание и исследовательские стратегии / Б. Бейзер.— 1990.— Текст: электронный.

Пути исследования нечетких систем управления в среде fuzzyTECH Explorer Edition

Кульмамиров Серик Алгожаевич, кандидат технических наук, академик общественной академии МАИН;
 Батыл Булдишин, студент
 Казахский национальный университет имени Аль-Фараби (г. Алматы)

Рассмотрена структура и исследованы параметры нечетких систем управления на основе теории нечетких множеств. Получены результаты моделирования функции нечеткого контроллера и оценки его функциональности путем его моделирования в программе fuzzyTECH Explorer Edition.

Ключевые слова: нечеткое моделирование, нечеткий контроллер, микропроцессорная система, терм, функция принадлежности, fuzzy-система.

В настоящее время массовым становится применение так называемых нечетких систем управления под обозначением «fuzzy-системы». Большинство этих систем всецело входит в современный класс интеллектуальных систем управления (ИСУ). Fuzzy-системы строятся на теории нечеткой логики (нечетких множеств), разработанной профессором Лотфи Заде [1965, 7].

ИСУ рождаются на исследовательских платформах искусственного интеллекта и больших данных (Big Data), где объект управления (ОУ) по структуре сложен и по логике его параметры не поддаются точным методам классической теории автоматического управления. Для анализа структуры и свойств ИСУ необходима огромная обработка больших зарегистрированных данных как априорное представление информации разного формата о поведении исследуемой системы. Применение fuzzy-систем наглядно демонстрируется рисунком 1.

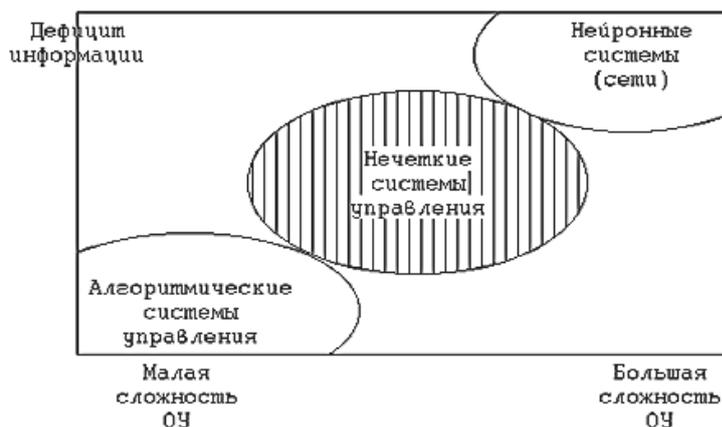


Рис. 1. Область применения нечетких систем управления

Принцип действия fuzzy-систем основан на применении свойств нечеткого логического вывода типа: «ЕСЛИ А..., ТО В...». Такие системы структурируются на базе знаний и элементах искусственного интеллекта. Обычно простейшее звено нечеткого управления содержит один восходящий нечеткий вывод (рисунок 2).

Схему вывода представим в виде:

Знание: «если X есть А, то Y есть В».

Факт: «X есть А'».

Вывод: «Y есть В'».

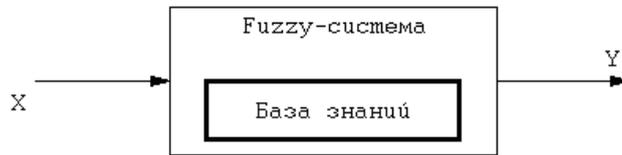


Рис. 2. Простейшее звено нечеткой системы

Здесь А, А' и В, В' — описание лингвистической переменной. Такое описание представляется в виде нечетких множеств $m(A)$, $m(B)$. Связь между входной переменной X и соответствующей ей лингвистической переменной А может быть представлена в виде характеристической функции (рисунок 3). Характеристическая функция является лингвистической переменной класса «Высокий». Это обозначение является термом лингвистической переменной.

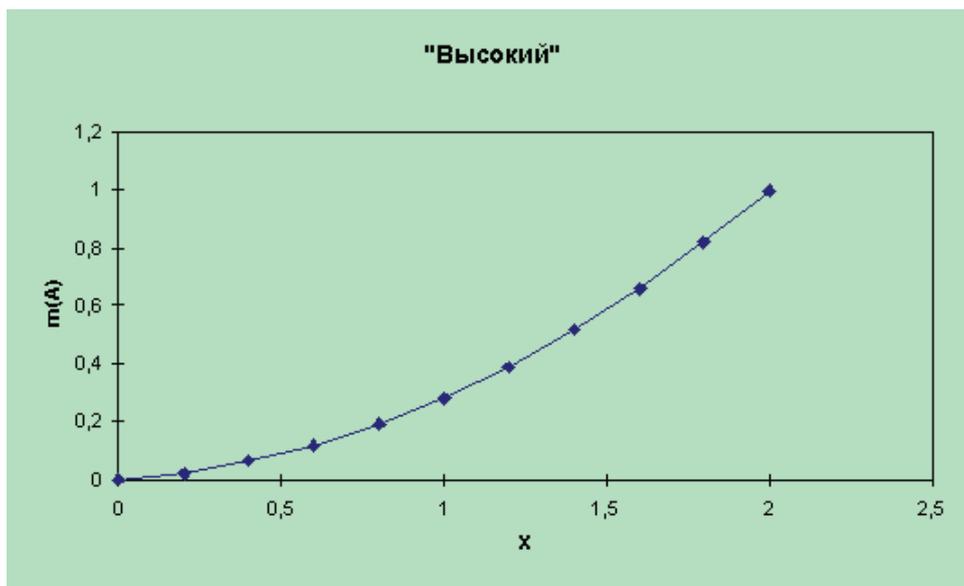


Рис. 3. Функция принадлежности лингвистической переменной «Высокий»

Здесь представленная функция имеет вид: $A = (x_i, a_i)$. Параметр x_i принадлежит к X. Переменная X является элементами множества, a_i — степенью принадлежности x_i к множеству А. При дискретном представлении функции принадлежности А будет иметь вектор вида: $A = (a_1, a_2, \dots, a_p, \dots, a_m)$, значения a_i находится в интервале (0, 1). Аналогично $B = (b_1, b_2, \dots, b_p, \dots, b_m)$, b_i также находится в интервале (0, 1).

В нечетком логическом выводе А (это причина) и В (это есть результат) связаны матрицей R. Такая матрица отражает нечеткие причинно-следственные отношения между А и В. Матрицу R следует представить как:

$$R = \begin{bmatrix} r_{11} & \dots & r_{1n} \\ \dots & r_{ij} & \dots \\ r_{m1} & \dots & r_{mn} \end{bmatrix}, \tag{1}$$

Профессор в своей теории Л. Заде определяет результат нечеткого вывода для параметра В' [1]:

$$B' = (b_1, b_2 \dots b_j \dots b_n) = (a_1, a_2 \dots a_i \dots a_m) \circ \begin{bmatrix} r_{11} & \dots & r_{1n} \\ \dots & r_{ij} & \dots \\ r_{m1} & \dots & r_{mn} \end{bmatrix} = A'R, \tag{2}$$

Из выражения (2) вытекает следующее выражение:

$$b'_i = \cup_i (a'_i \circ r'_{ii}), \tag{3}$$

Здесь приняты обозначения: \circ — логическое умножение (MIN); \cup — логическое сложение (MAX). Для непрерывной логики логическое умножение эквивалентно нахождению минимального значения, а логическое сложение — максимального значения.

Другой теоретик нечеткого множества Мамдани предложил другой алгоритм нечеткого отношения, наиболее часто используемый на практике как [2]:

$$r'_{ii} = a_i \cap b'_i, \tag{4}$$

где \cap — логическое умножение (MIN).

Реализация нечеткого управления в интеллектуальной системе управления с одним входом и одним нечетким выводом можно показать функциональной схемой с компенсационной емкостью (рисунок 4).

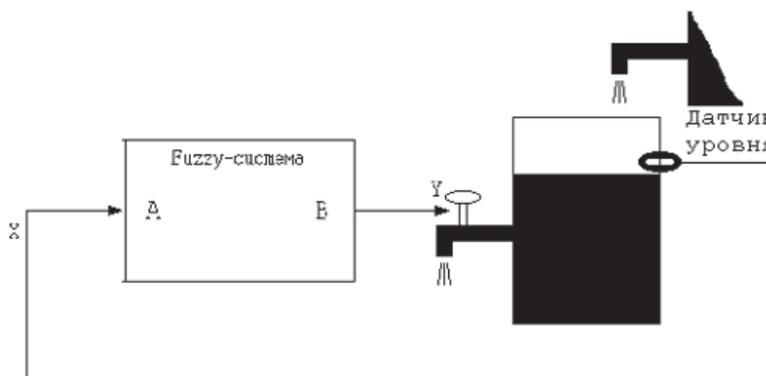


Рис. 4. Функциональная схема системы управления компенсационной емкостью

Такая система следит за протеканием вычислительного процесса, когда в компенсационной емкости должно быть достаточное количество жидкости. Значит, надо исключить ситуацию переполнения жидкости. Такая ИСУ отображает образ нечеткого вывода типа: «ЕСЛИ уровень »высокий«, ТО открыть клапан К».

Таким образом, здесь входная лингвистическая переменная пишется как: A = «Высокий», выходная переменная B = «Открыть».

Функции принадлежности, связывающие физическую входную переменную «Уровень» в метрах с лингвистикой «Высокий» и выходную переменную «Угол поворота вентиля» с лингвистикой «Открыт» показаны соответственно на рисунке 5, а и 5, б.

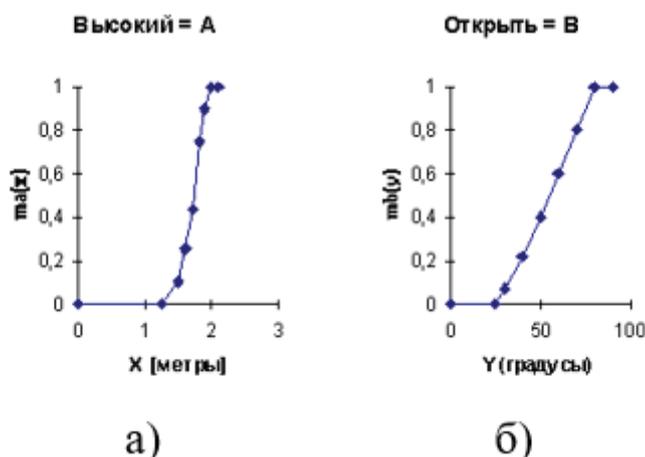


Рис. 5. Функции принадлежности «Высокий» и «Открыть»

По этим функциям входная и выходная переменные будут представлены в виде множеств:

«Высокий» = $m_A(x) = 0.1/1.5 + 0.3/1.6 + 0.7/1.7 + 0.8/1.8 + 0.9/1.9 + 1.0/2.0 + 1.0/2.1 + 1.0/2.2$;

«Открыть» = $m_B(y) = 0.1/30^\circ + 0.3/40^\circ + 0.7/50^\circ + 0.8/60^\circ + 0.9/70^\circ + 1.0/80^\circ + 1.0/90^\circ$.

В итоге нечеткий вывод «Если высокий, то открыть» реализуется в нечеткой системе на основе формул (3) и (4). Эта реализация продемонстрирована на рисунке 6.

Управление происходит в 3 этапа:

– фаззификация — превращение датчика уровня (например, 1.75 м) в лингвистическую переменную (A' — предельно высокий).

На рисунке 6 это представлено в виде множества $m_A(x)$ с учетом разброса измеряемого параметра x ;

– реализация нечеткого вывода по формуле:

$$b'_j = I_i(a'_i \text{ принадлежит } a_i) b_j. \quad (5)$$

Этот процесс показан на рисунке 6, б и в;

– дефаззификация — превращение выходного множества $m_{B'}$ (y) в выходной сигнал вида y . На рисунке 6, в дефаззификация множества показана в виде реализации метода центра тяжести (фигуры, заштрихованной на рисунке 6, в). Значение сигнала y' равно 70° .

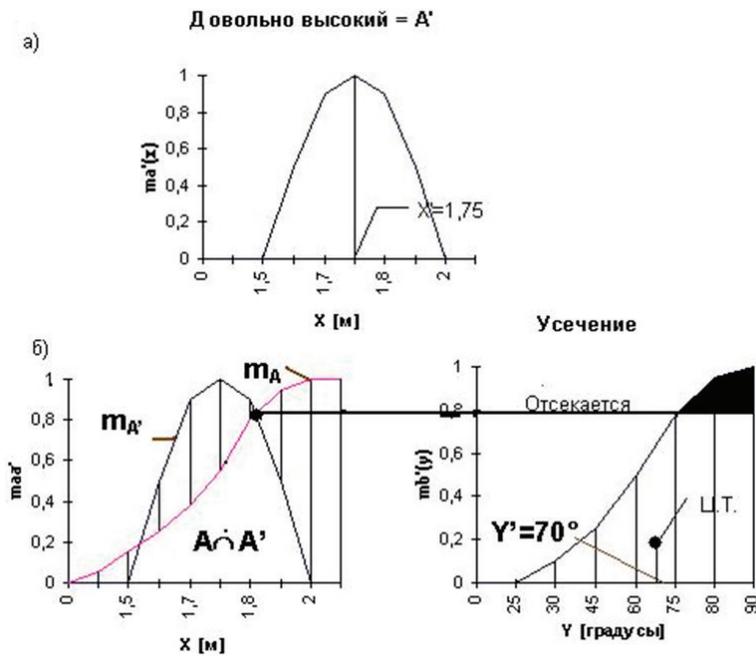


Рис. 6. Реализация принципа управления в fuzzy-системе

Нечеткое управление здесь может быть реализовано так называемыми нечеткими контроллерами. Такое устройство является машиной нечетких выводов (МНВ) [8], структура которой показана на рисунке 7.

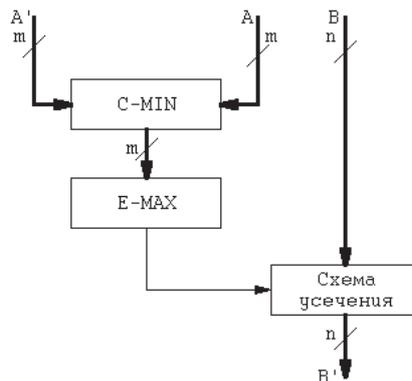


Рис. 7. Функциональная схема МНВ

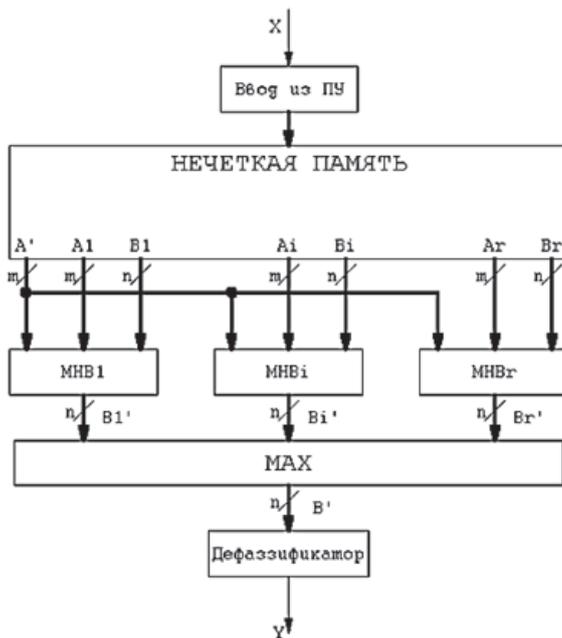


Рис. 8. Базовая архитектура нечеткого контроллера

Эта машина реализует нечеткий вывод вида: «ЕСЛИ А, ТО В» по формуле (5). Реализованный вывод указан на рисунке 6. Блок, реализующий функцию C-MIN, представляет пересечение множеств А и А'. Блок, реализующий функцию E-MAX, показывает из множества «А принадлежит А'» максимальный элемент с фиксированным значением. Такой процесс выполняет усечение множества В. Значит множество В превращается в множество В'.

Базовая архитектура нечеткого контроллера, реализующего нечеткие выводы (r), располагает r машинами нечетких выводов (рисунок 8). Кроме МНВ₁ базовый нечеткий контроллер располагает у себя нечеткую память. Здесь хранится база знаний и схема объединения результатов выводов.

Таким образом, нечеткое управление реализовано через обычные, так называемых специальное логико-арифметическое контроллеры. В таком контроллере нечеткие выводы генерируются вычислением значений выходного сигнала. Этот сигнал является лингвистической переменной и характеризуется через входные лингвистические переменные с операциями «И» и «ИЛИ». Переход от лингвистических переменных (термов) осуществляется через соответствующие характеристические функции, называемые функциями принадлежности (рисунок 9). Можно формулировать, что для реализации простейших алгоритмов нечеткого управления достаточно, чтобы лингвистические переменные содержали от 3 до 7 термов: например, Z-функция, р-функция, l-функция, S-функция.

Z-функция, р-функция, l-функция, S-функция



Рис. 9. Типовые функции принадлежности в нечеткой системе

Теперь можем с уверенностью представить, что разработка алгоритма нечеткого управления может содержать следующие 3 этапа:

Этап 1. Фаззификация — переход от физического значения к лингвистическим переменным и определение характеристических функций. Этот процесс осуществляется выполнением шагов:

Шаг 1. Для каждого терма лингвистической переменной определяется числовое значение физической величины (диапазоном значений). Такая величина лучше всего характеризует выбранный терм. Выбранным значениям присваивается характеристическое значение 1.

Шаг 2. Для терма выбирается диапазон значений физической переменной, где характеристическая функция имеет значение 0.

Шаг 3. После определения таких оптимальных значений выбираются промежуточные значения характеристических функций (рисунок 9).

Этап 2. Во втором этапе формулируются нечеткие правила. Нечеткие системы используют продукционные правила. Они удобны для описания зависимостей между лингвистическими переменными. Простое продукционное правило состоит из так называемого antecedента (ЕСЛИ...) и консеквента (ТО...) [6]. Antecedent содержит более одной посылки. В этом случае они объединяются через логику И или ИЛИ. Процесс формулировки нечеткого правила называется нечетким логическим выводом.

Процесс проходит в двух частях: обобщение и заключение. Например, для башенного крана (функциональная схема крана представлена на рисунке 10) входными переменными являются:

- расстояние от каретки до стены d [м];
- угол отклонения груза от вертикали j° .

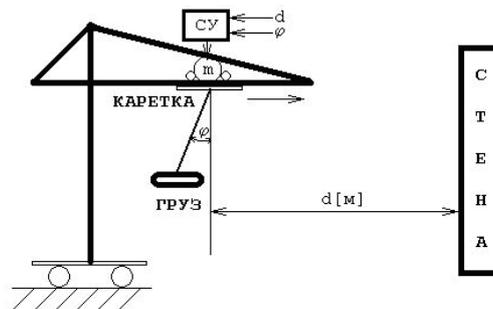


Рис. 10. Функциональная схема управления башенным краном

Для выходной переменной — мощность, подаваемая на двигатель каретки m . Теперь допустим, что каждая из этих переменных имеет 3 терма: нулевое, среднее и большое. Этим термам соответствуют свои характеристические величины:

- D_0 — нулевое (малое) расстояние;
- D_{cp} — среднее расстояние;
- D_6 — большое расстояние;
- j_0 — нулевой (малый) угол;
- j_{cp} — средний угол;
- j_6 — большой угол;
- M_0 — нулевая (малая) мощность;
- M_{cp} — средняя мощность;
- M_6 — большая мощность.

Все эти величины принимают значения на интервале (0–1). Например, одно из продукционных правил может быть сформулировано таким образом:

ЕСЛИ Дистанция = средняя И Угол = малый, ТО Мощность = средняя.

Это правило можно представить в виде логической формулы $M_{cp} = D_{cpj_0 = \min[D_{cp}, j_0]}$. После формулировки всех правил получится нечеткое значение выходной переменной со степенями принадлежности ее термов. В нашем примере — это M_0, M_{cp}, M_6 .

Чтобы исполнительное устройство крана сработало, надо перейти к значению выходного с игнала системы. Это — мощность крана m . В этом заключается сущность дефазификации.

Этап 3. Дефазификация. Наиболее распространены 3 способа дефазификации:

- центр максимума — это, когда выходная величина становится средним значением величин m_0, m_{cp}, m_6 . Тогда $m = (m_0 + m_{cp} + m_6) / 3$;

- наибольшее значение (MoM) — выбирается наибольшее значение величины: $m = \text{MAX}(m_0, m_{cp}, m_6)$;

- способ центраоида (CoA) — здесь значение выходного сигнала системы представляет координату центра тяжести площади фигуры через усечение характеристических функций по степеням принадлежности.

Теперь есть все условия для составления алгоритма функционирования интеллектуальной системы управления в виде башенного крана:

Этап 1. На этапе фазификации составим функции принадлежности для 3 переменных (рисунок 11).

Этап 2. Сформулируем продукционные правила. Перечислим комбинации входных переменных:

- $D_0j_0, D_0j_{cp}, D_0j_6, D_{cp}j_0, D_{cp}j_{cp}, D_{cp}j_6, D_6j_0, D_6j_{cp}, D_6j_6$; — M_0 ;
- $D_{cp}j_0, D_{cp}j_{cp}, D_{cp}j_6$; — M_{cp} ;
- D_6j_0, D_6j_{cp} ; — M_6 .

Эти комбинации разобьем на 3 группы. Группы соответствуют 3 термам выходного сигнала. На основании такого разбиения напишем выражения, соответствующие продукционным правилам определения выходного сигнала:

- $M_0 = D_0 \text{ И } D_{cp}j_6 = \max[D_0, \min(D_{cp}j_6)]$;

- $M_{cp} = D_{cp} j_{cp} И D_{\delta} j_{\delta} = \max [\min(D_{cp} j_{cp}); \min(D_{\delta} j_{\delta})];$
- $M_{\delta} = D_{\delta} (j_0 И j_{cp}) = \min [D_{\delta}; \max(j_0, j_{cp})].$

Производим вычисления для: $d = 25\text{м}$ и $j = 4^\circ$. Результаты заносим в таблицу 1.

Таблица 1. Результаты вычислений по формулам (6)

j_0	j_{cp}	j_{δ}	D_0	D_{cp}	D_{δ}	M_0	M_{cp}	M_{δ}
0.4	0.6	0	0.3	0.7	0	0.3	0.6	0

- $M_0 = \max [0.3; \min(0.7; 0)] = 0.3;$
- $M_{cp} = \max [\min(0.7; 0.6); \min(0; 0)] = 0.6;$
- $M_{\delta} = \min [0; \max(0.4; 0.6)] = 0.$

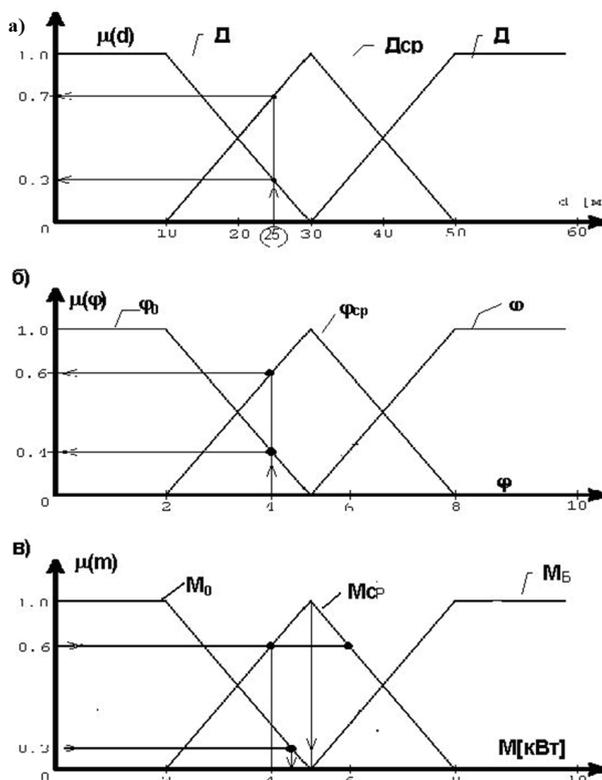


Рис. 11. Функции принадлежности входных величин (а, б) и выходного сигнала (в) системы управления башенным краном

Этап 3. Определим значения расчетов на этапе дефаззификации (рисунок 11):

- по СоМ получим $m = 4.5$ кВт;
- по МоМ и СоА получим $m = 5$ кВт.

Таким образом, для значений входных переменных d и j на двигатель следует подавать среднюю мощность, потому что величина степени принадлежности этого термина преобладает над другими.

В 2000 годы после бурного развития нечетких и интеллектуальных систем, особенно в Японии, многие технические специалисты штатов Америки и стран Европы явно заметили появления новой технологии в образе эксплуатации нечетких систем. Но к началу нового тысячелетия время было упущено. Мировым передовиком по нечетким системам стала Япония.

К концу 80-х годов в этой стране был налажен выпуск специальных нечетких контроллеров, разработанных по технологии СБИС (сейчас эта технология именуется как ПЛИС). Догонять реализацию японской технологии значило потерять десятки годы времени.

Американская фирма Intel принял альтернативное решение. В то время разнообразные промышленные контроллеры от MCS-51 до MCS-96 успешно применялись во многих цифровых платформах. Компания Intel решила предоставить разработчикам средство разработки приложений на базе этих контроллеров и включили туда возможность использования технологии по применению и внедрению нечетких систем (интеллектуализация контроллер).

Создание подобной системы позволили бы не осуществлять финансовые затраты на создание американских нечетких контроллеров. Такая созданная система получила название fuzzyTECH. В итоге такая нечеткая система была интеллектуализирована и завоевала огромную популярность в развитых странах: США, страны Европы, Япония).

Сейчас имеется многочисленные версии fuzzyTECH. Все они различаются по своим возможностям и стоимости. Приведем самый простой вычислитель из семейства fuzzyTECH — это пакет fuzzyTECH Explorer Edition.

Эта программная среда является одним из лучших средств обучения работе нечеткой системы. Она предоставляет пользователю широкие возможности по практическому освоению нечеткой технологии. Например, профессиональный пакет fuzzyTECH MCU-96 ориентирован на применение микроконтроллеров Intel семейства MCS-96.

Также имеется другой On-line вариант fuzzyTECH Edition. Эта программа может отладить реальную нечеткую систему, отлавливая ошибки прямо на ходу протекания процесса в системе. Приложение работает в реальном масштабе времени. Поэтому этот пакет незаменимое средство отладки созданных цифровых систем управления.

В составе программы fuzzyTECH аккумулированы многие алгоритмы и процедуры имеющейся нечеткой технологии: арбитражные функции принадлежности, современные методы дефаззификации и т. д.

Таким образом, среда FuzzyTECH — это полностью графическое средство поддержки разработки цифровых приложений и платформ на основе нечеткой логики. Оно является инструментом нечеткого моделирования и оптимизации реально созданного проекта. Пакет совместим с обычными средствами программной разработки. Графическое представление на экране всех этапов разработки поможет найти множество путей оптимизации или совершенствования проекта.

Пакет можно называть и генератором кода, который сможет поддерживать множество современного цифрового оборудования. Созданный пакетом fuzzyTECH исполнительный программный код можно конвертировать на язык программирования C.

Но ассемблерный код весьма уникален, наиболее корректно выполняется схемами аппаратной части и типу используемого микроконтроллера.

Покажем 5 стадии создания нечеткой системы используя программой fuzzyTECH:

1. Формулировка постановки задач в виде назначения лингвистических переменных, сопоставление термина к конкретной физической величине.
2. Формулирование сводов правил на основе стратегии управления системой (дефаззификация выходных данных).
3. Оптимизация структуры исследуемой (анализ свойств и поведения нечеткой системы через подготовленные данные или тестирование программной модели системы).
4. Организация процедуры проведения оптимизации параметров прямо в среде Интернет (тестирование структуры системы к реальному объекту и оптимизация структуры системы в реальных условиях).
5. Реализация нечеткой системы (выбор варианта генерируемого кода для конкретного микропроцессора может, такой вариант может быть записан в ПЗУ реальной системы управления).

В пакете fuzzyTECH все этапы разработки осуществляются через нажатие соответствующих кнопок мыши. Внутренний код fuzzyTECH создан на языке FTL (Fuzzy Technology Language). Это аппаратно независимый объектно-ориентированный язык для разработки нечетких систем. Он имеет открытую архитектуру и может быть расширен в зависимости от потребностей пользователя.

Перечислим состав пакета программы fuzzyTECH: редакторы (Editors), отладочный модуль (Debugger), анализ созданной структуры (Analyzer), специальный генератор кодов (Code Generation — реализация нечетких систем), опции (Options — конфигуратор с инструментальным интерфейсом).

При запуске программы fuzzyTECH появляется два окна (рисунок 12): Окно Редактора проекта и Окно лингвистических переменных (LV). Редактор Проекта содержит структуру системы нечеткой логики (интерфейсы и блоки правил), LV-окно содержит список всех лингвистических переменных.

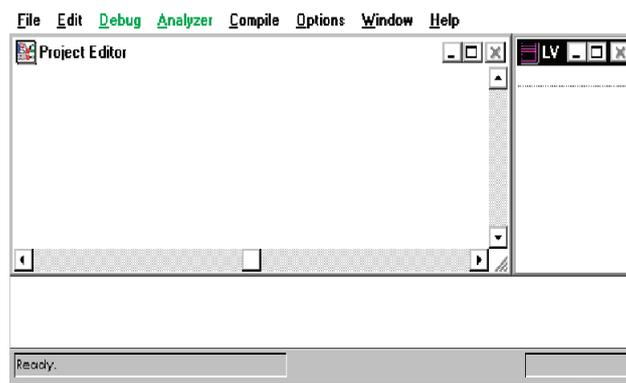


Рис. 12. Окна программы fuzzyTECH

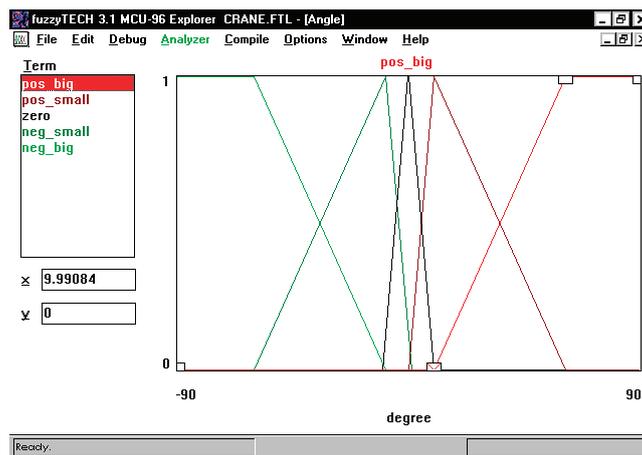


Рис. 13. Вид окна fuzzyTECH-редактирования термов

В теории нечетких множеств наиболее часто используется 3 или 7 количество термов. Обычно 1 или 2 термина не используются. Причина этого явления ясна. Человеческий разум чаще всего оценивает и представляет реальные числа (т.е. число имитации лингвистических переменных). Чаще всего в человеческом уме удобно раскладывается 3 величины: две величины низкий и высокий пределы, а также середина. Это число как терм наиболее удобно запоминается мозгом человека. Точно также можно комментировать самое большое количество термов как арабское число 7. Этот факт можно пояснить тем, что запоминание чисел включает краткосрочную память мозга. Человеческий мозг наиболее лучше работает не более 7 символическими величинами. Также известно, что переменные с «центром» или крайними «нулями» наилучше работают с нечетным количеством термов (рисунок 13).

Авторы статьи считают, что пакет программ fuzzyTECH является удачной и востребованной разработкой для углубления в теорию Искусственного интеллекта. Методика этой программы сотни раз может облегчить проектирование и создание микропроцессорных систем управления. Мы уверены, что цифровые системы в будущем поколении будут функционировать на основе теории нечеткой технологии.

Литература:

1. Микропроцессорные системы автоматического управления / Бесекерский В. А. и др.— Л.: Машиностроение, Лен. отд. 1988–365с.
2. Боборыкин А. В. и др. Однокристалльные микроЭВМ, М.: МИКАП, 1994.
3. Гребнев В. В. Однокристалльные микроЭВМ (микроконтроллеры) семейства MCS-96, 1995.
4. Микропроцессоры в системах автоматического управления. INTEL 8XC196MC: Учеб. пособие / Бутырин Н. Г. и др.: СПбГТУ, СПб., 1995, 115с.
5. Изерман Р. Цифровые системы управления: пер. с англ.— М.: Мир, 1984–541с.
6. Микропроцессорное управление электроприводами станков с ЧПУ. / Тихомиров Э. Л. и др.— М.: Машиностроение, 1990.
7. Заде Л. Понятие лингвистической переменной и его применение к принятию приближенных решений., М.: Мир, 1976, 165с.
8. Прикладные нечеткие системы / Тэрано Т., Асаи К., Сугено М., 1993.

Основные проблемы информационной безопасности судов

Кунакбаева Эльвира Рашитовна, студент магистратуры
Камчатский государственный технический университет (г. Петропавловск-Камчатский)

В данной статье рассматриваются основные вопросы информационной безопасности, такие как принципы защиты информации, шифрование. Особое внимание уделяется методу защиты шифрованием, использованию электронных подписей.

Ключевые слова: защита информации, криптография, криптоанализ, шифрование, дешифрование, электронная подпись.

Национальная безопасность страны и ее региона в современных условиях напрямую зависит от уровня информационной безопасности и существующих условий информаци-

онной безопасности. Информационная безопасность играет ключевую роль в обеспечении жизненно важных интересов Российской Федерации и ее регионов. Революция в области ин-

формационных технологий (появление глобального интернета, мобильных средств связи и т.д.).

Качественно расширились технические возможности специалистов в области информационной безопасности. Информационная безопасность во многом зависит от безопасности технических структур (компьютерных систем управления, баз данных и т.д.). Выход из строя может привести к нарушению интегрированной системы безопасности.

Существует три основных принципа, которые должна обеспечивать информационная безопасность:

- Целостность данных-предотвращение сбоев, приводящих к потере информации, а также предотвращение несанкционированного создания или уничтожения данных;
- Конфиденциальность информации;
- Доступность информации для всех авторизованных пользователей.

Защита информации — это комплексная задача, направленная на обеспечение безопасности, достигаемая за счет внедрения систем информационной безопасности. Проблема информационной безопасности многомерна и сложна, включает в себя множество важных задач. Кроме того, безопасность информации усугубляется проникновением технологий обработки и передачи данных во все сферы жизни общества, в первую очередь в компьютерные системы.

При разработке компьютерной системы сбои или ошибки в ее работе могут привести к серьезным последствиям, вопросы компьютерной безопасности становятся приоритетными. Сегодня в секторе судоходства защита информации в сети почти так же важна, как и обеспечение безопасности и сохранности судов.

Рекомендации МАКО (Международная ассоциация классификационных обществ) являются результатом эффективного отраслевого сотрудничества и содержат необходимые рекомендации по развитию и поддержанию информационной безопасности судов. МАКО — международное объединение классификационных обществ, ставящее своей целью выработку стандартов и правил в отношении обеспечения безопасности морских перевозок [3].

Примеры рисков информационной безопасности включают:

- Риск раскрытия или искажения информации;
- Риск отключения питания;
- Риск кражи или утечки;
- Риск изменения судовых тарифов.

Планы и процедуры управления рисками информационной безопасности судоходных компаний должны соответствовать существующим процедурам управления рисками кодекса ISM и спецификации ISPS.

Ниже представлен рисунок, показывающий состав функциональных элементов системы информационной защиты:



Эти функциональные элементы должны быть включены во всю систему управления рисками. Они отражают необходимые действия и ожидаемые результаты для эффективного управления информационными рисками ключевых систем, участвующих в морских операциях.

От береговых менеджеров до членов экипажа системы информационной безопасности должны использоваться на всех уровнях компании и сочетаться с системами безопасности и охраны для обеспечения эффективности работы судна.

Уязвимыми могут быть следующие корабельные системы:

- Система ходового моста;
- Грузовая система;
- Системы управления транспортом и энергетикой;
- Система контроля доступа на борту;
- Система обслуживания пассажиров;
- Публичная сеть;
- Система социального обеспечения членов экипажа;
- Системы радиосвязи и связи.

Для обеспечения информационной безопасности использовались различные методы. Большинство из них основаны на науке криптографии, методе обеспечения конфиденциальности и достоверности информации (целостности и авторства). Суть защиты зашифрованных данных заключается в передаче информации в зашифрованном виде, в котором она вряд ли будет прочитана посторонними людьми. Традиционно криптография включает в себя методы использования ключей для шифрования и дешифрования, современная наука предлагает десяток различных открытых алгоритмов шифрования и состоит из 4 частей: симметричная криптография, криптография с открытым ключом, электронные подписи и управление ключами. Симметричная криптографическая система использует один и тот же ключ для шифрования и дешифрования. В такой системе можно выполнить преобразование текста: замена, по некоторым правилам заменить один символ другим символом; замена, заменить символ.

В соответствии с определенными аналитическими правилами аналитическое преобразование; комбинированное преобразование, то есть в соответствии с определенным порядком с использованием набора основных методов преобразования [2].

Опасность использования этого метода заключается в том, что, если получатель не знает ключа, он также должен быть передан вместе с зашифрованным текстом по каналу связи, что поставит под угрозу передаваемые данные. В этом отношении система открытых ключей является наиболее безопасной. В такой системе получатель генерирует два взаимосвязанных ключа, один из которых является открытым и публикуется отправителю, а закрытый ключ известен только получателю.

Система открытых ключей использует необратимую функцию, которая исключает возможность использования открытого ключа для расшифровки текста или обнаружения закрытого ключа. Эти системы опираются на следующие типы необратимых переменных: вычисление корня алгебраического выражения, разложение чисел на простые множители и вычисление логарифмов в конечной области [1, 2]. Электронные подписи позволяют проверить информацию, сверив подпись с образцом (если таковой имеется), а также указать авторство.

К сожалению, этот метод аутентификации не является надежным, поэтому мы рекомендуем использовать его в сочетании с другими алгоритмами шифрования. Мало выбрать криптографическую систему для защиты информации, но нужно еще и управлять ключами, то есть генерацией, накоплением и распределением ключей. Вы можете генерировать ключи случайным образом или на основе пользовательских данных. В случае пользовательских данных один и тот же зашифрованный входной текст имеет один и тот же результат, в отличие от случайных методов, поэтому он наиболее уязвим.

Ключи накапливаются в специальной базе данных, а ключи, используемые для шифрования, дешифрования и проверки подписи, строго разделены. Каждый пользователь имеет свой собственный набор ключей. Распределение ключей использует систему шифрования с открытым ключом.

Существует два возможных варианта: распространение открытого ключа и распространение секретного ключа с использованием шифрования с открытым ключом [4].

Средства защиты зашифрованной информации или SCSI-с этим словом сталкиваются не только специалисты по информационной безопасности, но и обычные люди.

Одним из определений термина SCSI, используемых в различных официальных документах, является компьютерное средство, выполняющее преобразование информации в шифрование для обеспечения ее безопасности. При этом под компьютерными средствами понимается совокупность программных и технических элементов системы обработки данных, которые могут работать независимо или в составе других систем.

Зашифрованные информационные инструменты используются в различных областях, непосредственно связанных с бизнес-процессами. Это включает в себя надежное хранение, защищенные каналы связи и защищенный электронный документооборот (в этой области используется ЭЦП).

Помимо криптографии, широко используются и другие методы защиты информации, но наиболее актуальным и используемым в настоящее время является парольный метод защиты информации. Исходя из того, что существует несколько типов криптографических систем, сфера применения метода расширяется, поскольку в случае, когда системы с открытым ключом использовать не следует, на помощь могут прийти симметричные криптографические системы.

Что касается решения вопросов информационной безопасности, то с появлением новых средств защиты также появились новые, более сложные методы несанкционированного доступа и другие угрозы, поэтому проблема защиты информации, всегда будет открытой и актуальной [5].

В заключении, следует отметить следующие моменты. К требованиям закона нужно относиться серьезно. Исходя из опыта реализации нового закона, особенно в сфере транспортной безопасности, можно прогнозировать, что вступление закона в силу может привести к увеличению бюрократических процедур, количества документов, необходимых для транспортной инфраструктуры и / или транспортных субъектов, а также количества контрольно-надзорных мероприятий. Определение, содержащееся в законе, позволяет представить, что очень широкий круг объектов, включая суда, береговые и портовые системы, может быть отнесен к транспортным КИИ. Стандарты ИМО рекомендуют осуществлять управление киберрисками путем естественного расширения существующих подходов к управлению безопасностью на море и безопасностью судов [6].

Однако российское законодательство предусматривает целый ряд вопросов безопасности в отдельных, не связанных между собой нормативных правовых актах, и реализовать комплексный подход к обеспечению безопасности объектов водного транспорта практически невозможно [7]. Имея это в виду, легко прогнозировать ситуацию при отсутствии стан-

дартов кибербезопасности ИМО и интеграции российского законодательства в области информационной безопасности. Судовладельцы и субъекты инфраструктуры морского и речного транспорта обязаны соблюдать международное (ИМО) и российское законодательство [8].

Иными словами, необходимо обеспечить параллельную и одновременную работу двух систем информационной без-

опасности. Следует отметить, что эта проблема затронет весь транспортный сектор, а не только морской и речной транспорт. Мы считаем, что Министерство транспорта Российской Федерации, подведомственные ему федеральные органы, а также транспортное сообщество должны проанализировать ситуацию, обеспечить безопасность транспортной информации и сформировать четкую позицию по этому вопросу.

Литература:

1. Кибербезопасность на бескрайних морях: [блог]//Positive Technologies. Хабрхабр: [сайт]. URL: <https://habrahabr.ru/company/pt/blog/303198/> (дата обращения 19.11.2020).
2. Кибербезопасность на бескрайних морях: [блог]//Positive Technologies. Хабрхабр: [сайт]. URL: <https://habrahabr.ru/company/pt/blog/303198/> (дата обращения 18.11.2020).
3. Allianz проанализировал безопасность судоходства в мире//Allianz: [сайт]. URL: <https://www.allianz.ru/ru/press/news/article22445986> (дата обращения 07.11.2020).
4. Кибербезопасность на бескрайних морях: [блог]//Positive Technologies. Хабрхабр: [сайт]. URL: <https://habrahabr.ru/company/pt/blog/303198/> (дата обращения 18.11.2020).
5. Руководство ИМО предупреждает о кибер-уязвимости рулевого устройства//Crewmarket.net: [сайт]. URL: <http://news.crewmarket.net/2016/05/rukovodstvo-imopreduprezhdaet-o-kiber-uyazvimos tirulevogo-ustrojstva.html> (дата обращения 17.10.2020).
6. Международная морская организация: [сайт]. URL: [http://www.imo.org/en/OurWork/Facilitation/FALCommittee/Facilitation/FAL41-17-Tableofcontents\(Secretariat\).pdf](http://www.imo.org/en/OurWork/Facilitation/FALCommittee/Facilitation/FAL41-17-Tableofcontents(Secretariat).pdf) (дата обращения 19.11.2020).
7. Международная морская организация: [сайт]. URL: [http://www.imo.org/en/OurWork/Security/WestAfrica/Documents/ResolutionMSC.428\(98\)-MaritimeCyberRiskManagementinSafetyManagementSystems.pdf](http://www.imo.org/en/OurWork/Security/WestAfrica/Documents/ResolutionMSC.428(98)-MaritimeCyberRiskManagementinSafetyManagementSystems.pdf) (дата обращения 16.10.2020).
8. Международная морская организация: [сайт]. URL: [http://www.imo.org/en/OurWork/Security/Guide_to_Maritime_Security/Documents/MSC-FAL.1-Circ.3-GuidelinesOnMaritimeCyberRiskManagement\(Secretariat\).pdf](http://www.imo.org/en/OurWork/Security/Guide_to_Maritime_Security/Documents/MSC-FAL.1-Circ.3-GuidelinesOnMaritimeCyberRiskManagement(Secretariat).pdf) (дата обращения 19.11.2020).
9. Международная палата судоходства: [сайт]. URL: <http://www.ics-shipping.org/docs/default-source/resources/safety-security-and-operations/guidelines-on-cyber-security-onboardships.pdf?sfvrsn=16> (дата обращения 19.11.2020).
10. Федеральный закон от 09.02.2007 № 16-ФЗ (ред. от 06.07.2016) «О транспортной безопасности» (с изм. и доп., вступ. в силу с 21.12.2016). [Электронный ресурс] URL: http://www.consultant.ru/document/cons_doc_LAW_66069/ (дата обращения 19.10.2020.)
11. Федеральный закон от 26.07.2017 № 187-ФЗ «О безопасности критической информационной инфраструктуры Российской Федерации». [Электронный ресурс] URL: http://www.consultant.ru/document/cons_doc_LAW_220885/ (дата обращения 09.10.2020).

Компьютерные игры как источник образования

Макушева Ольга Николаевна, кандидат экономических наук, доцент;

Щербинин Георгий Артурович, студент

Московский государственный институт культуры

Сегодняшний век является веком информационных технологий. Казалось бы, еще вчера мы пользовались простенькими телефонами, чтобы поддерживать связь с близкими людьми, но уже сегодня этот гаджет заменяет целый персональный компьютер. Информатизация не обошла стороной и игры, они встали на ступень выше и стали компьютерными. Компьютерная игра — это компьютерная программа, служащая для организации игрового процесса, или геймплея, связи с партнерами по игре, или сама выступающая в качестве партнера [1].

Многие люди, по большей части старшего поколения, считают компьютерные игры бессмысленной тратой времени,

процессом деградации личности и даже «воплощением зла». В статье автор постарается не только опровергнуть данный стереотип, но и доказать на примерах, что видеоигры могут нести не только развлекательный, но и образовательный характер.

Игровое обучение — форма учебного процесса в условных ситуациях, направленная на воссоздание и усвоение общественного опыта во всех его проявлениях: знаниях, навыках, умениях, эмоционально-оценочной деятельности [2]. Также это можно назвать всевозможными формами образования без принуждения, образовательные развлечения, включая образование посредством развлечения, или эдьютеймент. Игровое об-

учение несет древнейшие корни: дети обучаются, познают окружающий мир посредством игр, принимая на себя различные роли: отца, мамы, фельдшера, пожарного, благородного рыцаря и прочие. Соответственно, игра обучает, развивает, воспитывает, социализирует, развлекает и даёт отдых. Но исторически одна из первых её задач — обучение. К важнейшим свойствам игры относят тот факт, что в игре, преодолевая трудности, дети получают навык преодоления экстремальных ситуаций. Причём высокий уровень активности достигается без принуждения. Игровое обучение также имеет свои признаки, особенности: *осуществляется человеком по желанию, с удовольствием от самого процесса деятельности, а не из-за поощрения или положительной оценки; инициативность; парадоксальное пересечение прямых и косвенных правил, сопрягающих содержание игры с элементами общественного опыта; эмоционально напряженная, приподнятая, состязательная, конкурентная деятельность* [3]. Некоторые исследователи даже указывают на то, что участники игры, решая игровые задачи, увлечены и не отвлекаются на противодействия со стороны.

Сегодня эти же игры изменили свой образ, преобразились, прошли стадию информатизации и приобрели статус компьютерной игры, но при этом не потеряли своих основных признаков и свойств. Но тогда почему возникает столько негатива вокруг видеоигр? Дело в том, что в последнее время по телевизору, в социальных сетях, в газетах и иных источниках информации можно наткнуться на новости следующего характера: «Школьник пытался сбросить одноклассника с крыши школы», «подросток расстрелял сотрудников учреждения N и т.д.». Это социально-психологический вопрос. В первую очередь, на чем делают акцент люди, это увлечения компьютерными играми подростками. Но самое интересное состоит в том, что мало кто интересуется жизнью внутри семьи подростка, ведь проще свалить всю вину на видеоигры, они же делают психику неустойчивой, а ребенка чрезмерно агрессивным. Существуют различные психологические типы людей: кто-то может быть высокоэмоциональным, а кто-то более прохладным, сухим,— и этот фактор нельзя исключать, поскольку каждому ребенку нужно внимание, особенно эмоциональному. Если же родители, родственники или друзья не уделяют ему внимания, то ребенок начинает постепенно терять способность к социализации, он пытается найти выход, чтобы почувствовать себя кому-то нужным. Здесь ему на помощь приходит его внутренний мир, мечтания или компьютерные игры, в которых он прячется от отвергшего его социума, пытается найти для себя место. А еще хуже ситуация обстоит, если ребенка не только забывают, но и проявляют к нему агрессию со стороны, допустим, одноклассков. Это окончательно ломает его психологическое состояние. И здесь уже все зависит от темперамента: ребенок может, как было упомянуто выше, углубиться в свой внутренний мир и прийти к идее самоубийства, а может резко отреагировать, избив до полусмерти обидчиков. Соответственно, можно сделать вывод, что не игры виноваты в проблемах человека, а именно люди. *«Из всех существ, которые были созданы, человек — самое отвратительное. Из всех живущих на Земле, ему одному свойственна злоба, — самый низменный из всех инстинктов, пороков, самый мерзкий. Че-*

ловек — единственное на свете существо, способное причинять боль просто так, без оснований, сознавая, что он её причиняет» — Марк Твен.

Далее в качестве примеров будут приведены крупные проекты (видеоигры) с образовательным контентом.

Assassin's Creed: Интерактивный тур. Серия игр Assassin's Creed известна различными историческими событиями: крестовые походы, эпоха Ренессанса, война за независимость в США и другие. Игрокам предоставлялась возможность не только узнать сюжет каждой игры, но и узнать, как выглядел мир, окунуться в захватывающую дух атмосферу. При этом, если игрок находил в игровом мире какую-либо достопримечательность (допустим, Колизей в Риме), ему в базу данных игры приходила краткая историческая справка об этом месте. В 2017 году разработчики создают Assassin's Creed Origins, погружая игрока в атмосферу Древнего Египта. При этом они убрали вышеупомянутую базу данных и добавили новый режим — Интерактивный тур. Можно сказать, это виртуальный музей, который позволяет узнать достоверную информацию о географии, значимых объектах, достопримечательностях, особенностях быта жителей Древнего Египта и о жизни известных исторических личностей. Разработчики на своем сайте отмечают: *«Он подойдет учителям и ученикам, опытным геймерам и тем, кто далек от мира видеоигр. Тур можно проходить в комфортном для себя темпе, прокладывая маршрут самостоятельно или под руководством историков и других специалистов»*. Интерактивный тур завоевал награды в номинациях «Лучшая обучающая игра» и «Выбор игроков» на Games for Change 2019 года. Его активно используют учителя, чтобы привить ученикам интерес к истории. Возможность перемещаться по Древним Египту позволяет учащимся получить максимально наглядное представление об архитектуре и культуре прошлого. Также разработчики выпустили новую часть, в которой можно уже погулять и исследовать окрестности Древней Греции [4].

This War of Mine — уникальное произведение на тему войны. Она рассказывает про осажденный город, на улицах которого постоянно идут бои, но акцентируется внимание прежде всего на мирных жителях, которым в таких условиях приходится выживать. Игрокам предстоит сталкиваться с нехваткой ресурсов, сложными моральными выборами, которые определяют исход персонажа. Матеуш Моравецкий, премьер-министр Польши, включил игру в школьную программу в качестве внеклассного чтения. Прохождение игры будет рекомендовано старшеклассникам, изучающим историю и культуру, в частности, специализирующимся на социологии, философии и этике. Моравецкий прекрасно уверен, что игра идеально подходит для того, чтобы учащиеся узнали больше о людях [5]. Ужасы войны и жестокость людей теперь можно познать не только через рассказы, но и через компьютерную игру.

Minecraft Education Edition. Minecraft — популярна игра среди детей, в которой можно построить что угодно, целая виртуальная песочница: начиная с любого здания в реальную величину и заканчивая моделью работающего компьютера [6]. Издание Education Edition позволяет изучить любой предмет, начиная математикой и заканчивая химией, есть даже возмож-

ность пройти экскурсию по любому воссозданному месту: например, музей Царицыно, воссоздал свой заповедник во время карантина, чтобы в будущем проводить различные программы в виртуальном мире [7].

В 2014 году CEO агентство Insight ONE опубликовало статистику популярности компьютерных игр в России. Результат

вышел внушительным: 58% россиян играют в игры, средний возраст игрока — 30 лет, 68% всех игроков взрослые люди (старше 18 лет), 45% геймеров — женщины, 89% игроков — родители, которые играют со своими детьми [8]. За 2018 год количество увлеченных компьютерными играми увеличилось.

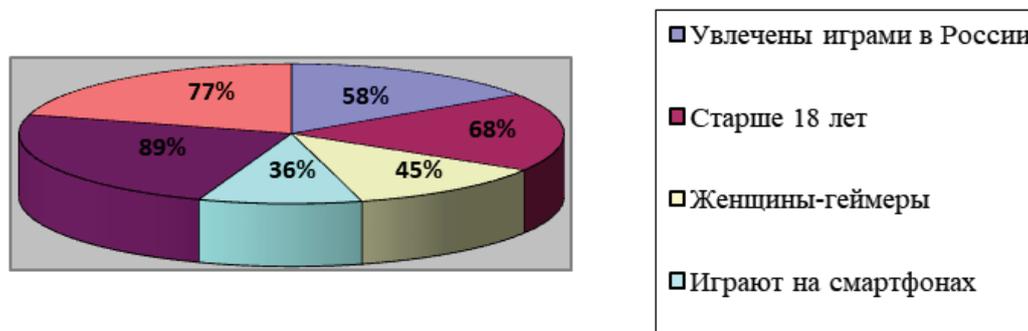


Рис. 1. Популярность компьютерных игр в России

Можно предположить, что за период пандемии в 2020 году процент геймеров продолжал расти в разы, поскольку людям как-то хотелось скрасить свой досуг. Таким образом, игры можно рассматривать в качестве источника образования, поскольку они не утратили своих первоначальных функций, признаков и свойств, а лишь сделали шаг в эру информационных технологий, прошли стадию информатизации. Поэтому не следует верить стереотипам, что видеоигры нарушают психику

человека и делают его агрессивным, все зависит от окружающих людей, их действий, но многие почему-то этого не замечают. Компьютерные игры можно однозначно назвать одним из источников образования. Они как информационные технологии в музейном деле: не полностью заменяют какой-либо предмет, а дополняют, приукрашают его и дают возможность не только посмотреть на него с другой стороны, но и даже взаимодействовать.

Литература:

1. <https://dic.academic.ru/dic.nsf/ruwiki/977059>
2. <https://nsportal.ru/shkola/obshchepedagogicheskie-tekhnologii/library/2013/11/26/igrovoe-obuchenie-statya>
3. https://ru.wikipedia.org/wiki/Игровое_обучение
4. <https://www.ubisoft.com/ru-ru/game/assassins-creed/discovery-tour>
5. https://www.igromania.ru/news/94880/This_War_of_Mine_dobavyat_v_polskuyu_shkolnuyu_programmu.html
6. <https://education.minecraft.net/>
7. <https://tsaritsyno-museum.ru/events/special/p/minecraft-tsaritsyno/>
8. <http://security.mosmetod.ru/internet-zavisimosti/statistika>

Перспективы проектирования бизнес-архитектуры в цепочке поставок с использованием технологии блокчейн

Санчес Кальвера Педро Висенте, студент магистратуры;
Карденас Карденас Кристиан Эпифанио, студент магистратуры
Донской государственный технический университет (г. Ростов-на-Дону)

Настоящая статья призвана показать общие перспективы реализации бизнес-архитектуры с использованием технологии блокчейн в цепочке поставок в рамках проекта (внедрение блокчейн технологии в сети снабжения в колумбийском контексте). Сперва будет дано краткое описание основных элементов, составляющих технологию блокчейн, затем будут даны различные примеры, происходящие между вышеупомянутыми элементами, и, наконец, предложен способ проведения экономического поиска возможного применения.

Ключевые слова: электронная торговля, банковские транзакции, отслеживание продуктов, защита потребителей, интернет вещей IoT, искусственный интеллект AI, машинное обучение, блокчейн, цепочка поставок, API, KPI.

В эпоху цифровых технологий постоянная эволюция человека достигла других уровней. Переход от создания видеогриг к возможности имитировать ситуации, которые раньше казались невозможными, например, перенос транзакций, совершаемых с помощью бумажных денег в цифровой мир (Bitcoin-Ethereum-Corda). Это привело к тому, что компании и микропредприятия стали разрабатывать виртуальные среды для развития своего бизнеса, где стремятся обеспечить большой поток взаимодействий между участниками и элементами, своевременный анализ и информационную безопасность, а также масштабируемость и инновации в своих бизнес-моделях.

Развитие новых технологий росло экспоненциально, что означает, что правительства и правительственные учреждения не могли осуществлять адекватное регулирование использования и объема децентрализованных технологий и, среди прочего, становится невозможным артикуляция каждого элемента. Напротив, если это будет сделано в каком-то случае, это правило уже устареет из-за степени развития, которую этот тип технологии имеет на основе данных через проблемы безопасности, а также технических возможностей для продвижения, которые у них есть.

Риск, связанный с применением этого типа технологии в финансовой системе, цепочках поставок и других секторах, высок, поэтому он воспринимается как угроза при проведении раз-

личных разработок, выходящих за пределы любой контролирующей экосистемы, либо со стороны проблемы безопасности, доступ к информации, нарушение конфиденциальности, затраты на внедрение и высокие технологические возможности, требуемые каждой системой и подсистемой, как следствие затрудняют следующий шаг к трансформации среды и к выполнению процессов, тем не менее, рост экосистемы блокчейнов имеет место и в дальнейшем будет увеличиваться.

Компании, которые не проводят реструктуризацию или изменения в своих бизнес-моделях, со временем будут вынуждены уйти с рынка [1]. Постоянное появление новых конкурентов, сильная устойчивость, адаптация к изменениям и инновации знаменуют собой новую эру в цепочках поставок, а также в других секторах экономики.

Согласно Всемирному экономическому форуму, вот тенденции в внедрении новых технологий в опросе нескольких компаний к 2025 году, где показана вероятность применения или принятия этих [2]; облачные вычисления, блокчейн, электронная коммерция, IoT, большие данные задают тон (см. Рисунок 1), поэтому мы заинтересованы в том, чтобы показать в этом документе различные элементы, которые следует учитывать в архитектуре бизнес-модели, и насколько серьезна их возможная интеграция между каждой из этих технологий.

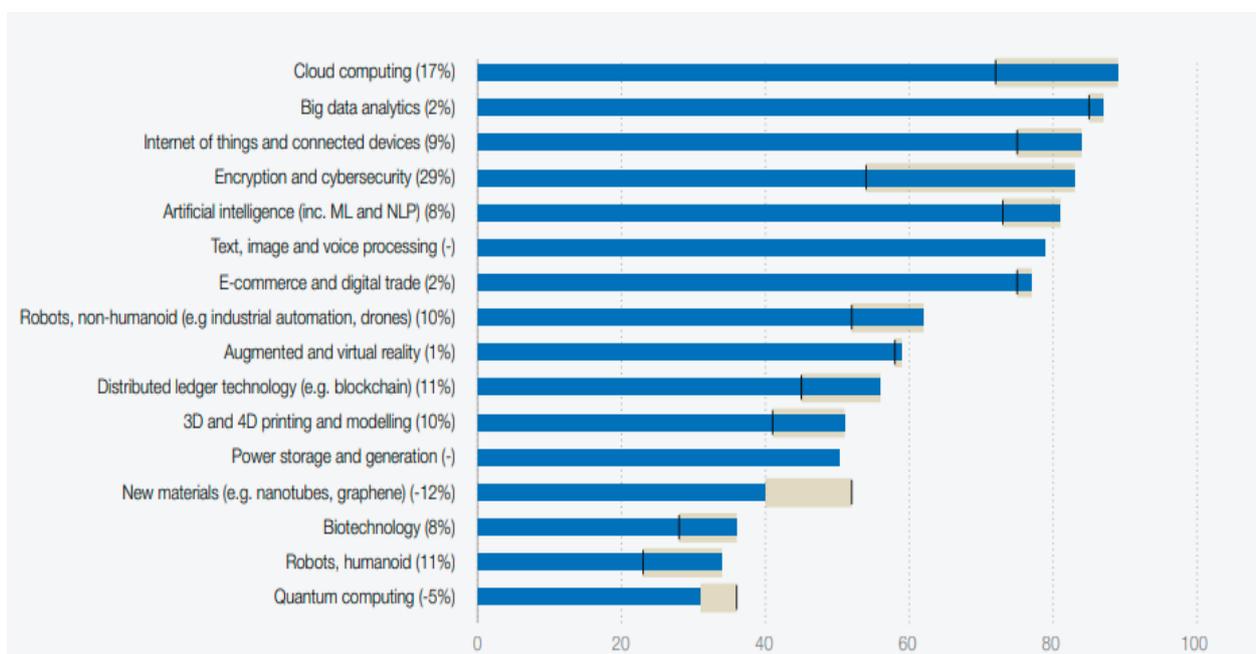


Рис. 1. Технологии, которые, вероятно, будут приняты к 2025 году (по доле опрошенных компаний), В. Э. Ф. [2]

Основные технические элементы

Управление производственно-сбытовыми цепочками представляет собой серьезную проблему для каждого из их участников из-за сложности их процессов, информационных потоков,

продуктов и транзакций, поэтому эти потоки требуют прослеживаемости, безопасности и прозрачности. В настоящее время цепочки поставок используют централизованные системы в качестве решения для управления производственно-сбытовыми процессами, однако эти процессы уязвимы для проблем взлома,

ошибок и коррупции [3]. В качестве умной альтернативы для решения этих ситуаций используются технологии DLT (Distributed ledger technologies) или технология блокчейн — инструмент, который обещает внести изменения в другие сферы, такие как финансовый сектор, цепочки поставок и другие рынки.

Технология блокчейн — это распределенная база данных общих публичных или частных записей или бухгалтерских книг всех цифровых событий, которые были выполнены и переданы участвующим агентам блокчейн. [4]

Важные элементы, которые мы должны включить или учитывать в архитектуре блокчейна, заключаются в следующем:

Криптография

Несомненно, сердцем блокчейна является криптография, которая позволяет использовать различные методы шифрования данных с целью сохранения конфиденциальности информации, поступающей в сеть, делая их безопасными и надежными с использованием подписанных ключей и цифровых фимов.

Сети блокчейнов криптографически связаны, а также хранят все записи транзакций, совершаемых участниками сети, чтобы понять рабочие процессы, прежде всего их объем, масштабируемость и безопасность сети должны быть определены. Они состоят из следующих элементов и процессов:

1. Узлы — это любое устройство или компьютер, подключенный к децентрализованной сети, позволяющей взаимодействовать и передавать информацию.
2. Запись — это запись всех транзакций, сделанных и распределенных на каждом узле в сети блокчейн.
3. Майнинг — это процесс создания нового легитимного блока с помощью приложения Proof-of-Work. Есть люди, которые посвящают свои узлы «добыче» новых блоков. [4]
4. Консенсус — это алгоритм, в котором все или большинство узлов в сети блокчейн согласны с одной и той же операцией или транзакцией, выполняемой в сети.

Смарт-контракты

Это термин, предложенный Ником Сабо в 1994 году [4]. Смарт-контракты — это небольшие компьютерные программы, которые пишутся в строках кода, они самовосстанавливаются, на них записываются соглашения между двумя или более заинтересованными сторонами, описанные здесь соглашения распределяются по всей сети блокчейнов, обеспечивая выполнение и проверку каждого из соглашений без вмешательства человека.

Система отсчета корпоративной архитектуры

Для проведения глубокого исследования цепочки поставок необходимо знать потоки информации, финансовых ресурсов, продуктов, а также людских ресурсов. Затем устанавливаются отношения, которые существуют между каждым из участвующих участников. В дальнейшем определяется архитектура проектирования применения многослойной технологии в случае блокчейна, как показано на рисунке 1 [5]:

- на уровне 1 регистрируются все участники, отслеживая любую логистику в конкретном;
- на уровне 2 каждый пользователь вводится с использованием различных типов биометрической безопасности, в зависимости от того, какую роль он осуществляет в цепочке поставок. Также запланирован смарт-контракт, который обеспечивает последовательность операций в цепочке;
- на уровне 3 ведется управление информационными индикаторами (логистика, качество, финансовые операции), также сохраняется информация для сгенерированного в блоке;
- на уровне 4 производится сбор физической информации с использованием различных систем IoT, здесь генерируются доказательства достоверности и прозрачности информации.



Рис. 2. Архитектура логистической системы блокчейн

Чтобы реализовать архитектуру блокчейна в любой бизнес-модели, необходимо искать иной тип интеграции между различными доступными технологическими платформами блокчейна, такими как Hyperledger, Ethereum, Corda и другими, и существующими бизнес-архитектурами, такими как ERP (планирование ресурсов предприятия) и CRM (система управления взаимоотношениями с клиентами); разработка программного уровня API (промежуточного программного обеспечения), которые уже были разработаны многими компаниями, для обеспечения связи или моста между существующими бизнес-моделями. Эти сервисы поддерживаются облачными сервисами, которые оцениваются по следующим принципам больших данных: объем, скорость, разнообразие, точность и ценность.

Для этой перспективы существуют различные подходы, как это предлагается в [6], где они предлагают визуализировать расположение облачной ERP-системы с элементами больших данных, адаптированными для RFID, в качестве метода управ-

ления запасами в цепочке поставок в режиме реального времени, кроме того, в этом документе подчеркивается важность интеграции процессов аналитической иерархии (ANP) в качестве метода многокритериального принятия решений (MCDM), с помощью которого можно оценивать различные компромиссы различных показателей, проводимых в цепочке поставок.

Согласно исследованию [7], прокси-уровень блокчейна создается для создания модуля между сетью или платформой блокчейна и интегрированными службами API, что делает все блокчейны одинаковыми, повышая производительность сервера. Автор описывает, что создать общий прокси-слой очень сложно, так как необходимо учитывать тип бизнес-модели или предлагаемого решения и его возможности. Однако реализация микросервисов с использованием технологии блокчейн позволит масштабировать эти решения для всей бизнес-модели (см. рисунок 3).

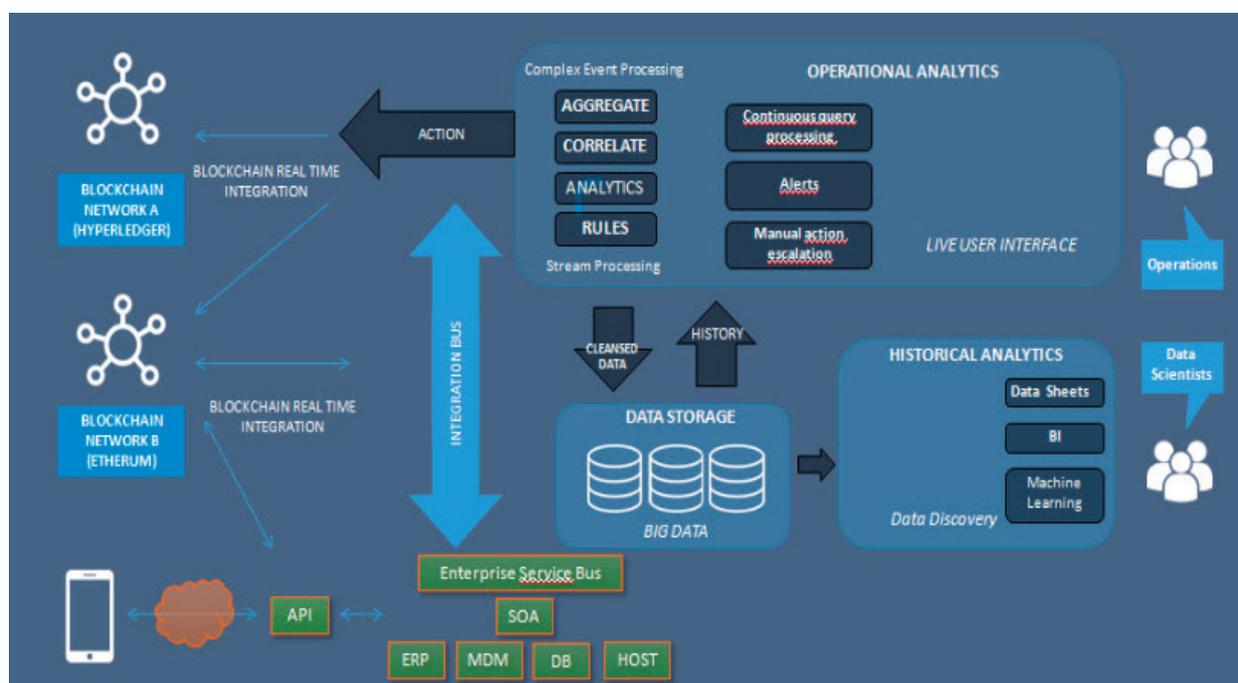


Рис. 3. Блокчейн в эталонной архитектуре. [8]

Эта бизнес-модель позволит:

- Упрощение сложностей, используя смарт-контракты и сеть блокчейнов;
- Взаимодействовать различным подмодулям блокчейна;
- Отслеживать процессы в реальном времени;
- Запуск процессов внутри и вне сети;
- Создать безопасную иерархию и уменьшать провалы сети блокчейна.

С точки зрения цепочки поставок смарт-контракты предлагают множество возможностей, поскольку они обеспечивают совместимость между такими технологиями, как IoT и AI. Например, выполнение следующих действий в торговле автоматически:

Хотя ожидания от смарт-контрактов для автоматизации процессов высоки, они создают проблемы, которые бросают вызов дизайнерам архитектуры, выходящие за рамки простой архитектуры сети, как описано [10], где указываются следующие проблемы:

1. Проблема исправления: проблема исправления или обновления каждого смарт-контракта относится к рассмотрению версий, которые будут применяться до его реализации, некоторые авторы выполняют проверку информации, применяя теорию игр в решении смарт-контракта с использованием методов согласования и проверки протокола.
2. Уязвимости в системе безопасности. Учитывая, что экосистемы блокчейнов состоят из миллионов узлов, когда риск регистрируется в смарт-контракте, вся цепочка оказывается

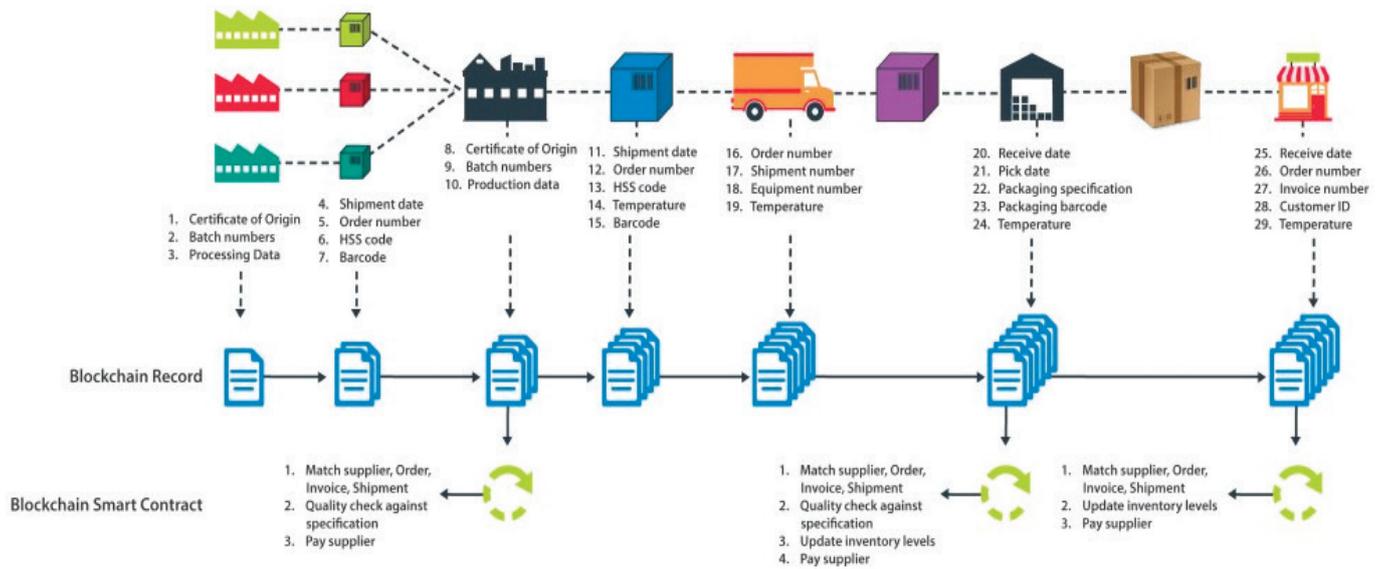


Рис. 4. Смарт-контракты в цепочке поставок [9]

уязвимой, теряя капитал и время; особенно в общедоступных сетях, поэтому рекомендуется проводить автоматические тесты безопасности, с помощью которых можно выявить ошибки и уязвимости в смарт-контракте.

3. Ошибки программного обеспечения: здесь делается ссылка исключительно на функциональные возможности, в которых используются различные методы для выявления ошибок, направленных на выполнение функциональных требований.

4. Вопросы конфиденциальности данных: это очень актуальная тема, поскольку реестр копирует всю информацию во всей цепочке, что делает ее деликатной темой. В этой ситуации рекомендуется интегрировать безопасное оборудование и конфиденциальные приложения как для пользователя, так и для смарт-контрактов.

5. Ограничения производительности, объем и скорость передачи данных — одна из величайших проблем, с которыми сталкивается технология блокчейн, в настоящее время растет сообщество, которое проводит исследования по улучшению факторов производительности, чтобы уменьшить пробелы интеграции с другими технологиями.

Что ожидается от цепочек поставок, использующих смарт-контракты, децентрализованные автономные операции и одновременную аутентификацию всех транзакций, выполняемых в рамках логистических операций; [10] увеличение инфраструктуры технологии IoT позволит шире использовать контракты на основе блокчейна.

Отношения с цепочками поставок и блокчейн

Помимо применения технологии блокчейн во всех вышеупомянутых аспектах необходимо провести оценку приложения с более финансовым подходом, для проекта [11] ожидается сравнение бизнес-архитектуры с использованием ключевых показателей эффективности или KPI (Key Performance Indicator).

KPI являются ключевыми показателями эффективности в компании для измерения транспорта с точки зрения достижения предложенных как конкретных, так и оперативных целей. Эти KPI будут оцениваться в соответствии с изученной цепочкой поставок или там, где установлена технология блокчейн.

В настоящее время один случай оценки эффективности KPI в цепочке поставок используется Walmart, где было обнаружено, что многие из их поставщиков не соответствуют показателю OTIF (On Time, In Full). [12]

OTIF (вовремя, в полном объеме) — это показатель, который измеряет количество партий, поставляемых по указанному графику, т.е. все суммы, согласованные с клиентом на согласованную дату, должны быть выполнены, имея в виду, что эта дата также не должна быть ожидаемой, поскольку она сопряжена с штрафами и расходами на техническое обслуживание запасов и душевые распределительных центров [13]

Например, walmart использует этот индикатор:

Оценка OTIF = дела, полученные в окне / общее количество заказанных дел.

И для расчета заказанных общих случаев используется:

Общее количество заказанных дел = дела доставлены рано + дела доставлены коротко + дела доставлены вовремя + дела доставлены с опозданием.

Поскольку мы видим, что это показатель, отражающий состояние поставок и удовлетворенность клиентов, большинство проблем, наблюдаемых при расчете этого показателя, заключается в том, что транзакции, извлеченные из любой ИТ-системы, зависят от манипуляций с человеком, перемещения товаров, попыток мошенничества с системой [12] и в других случаях с помощью предположений [13], в качестве решения этой проблемы используется интеграция с сетью блокчейн, используя смарт-контракты и поддерживаемые технологии IoT.

Внедрение технологии блокчейн обеспечит улучшение цепочки поставок, измеряемое индикатором OTIF, поскольку

в соответствии с ее характеристиками неизменности и децентрализации она обеспечивает большую прозрачность и прослеживаемость транзакций для заинтересованных сторон с учетом как крупных, так и небольших заказов.

Это обеспечивает лучшую рентгенографию в управлении цепочками поставок и позволяет лучше принимать решения по выявлению изменений в процессах (таких, как влияние хлыста на спрос), сокращению узких мест, отображению времени поставок, реагированию в режиме реального времени и оценке целесообразности автоматизации других процессов.

Современным примером сказанного выше можно посмотреть решение блокчейн (food truth), разработанная компанией IBM, Walmart и применяется в цепи поставок, (лист-зеленый), который использует платформу блокчейн из hyperledger, который позволяет пользователям (заинтересованные стороны) проводить отслеживание всех сделок, совершенных в процессе продукта, а также сеть хранит данные такие как температура, от фермы до магазина [12] и для клиентов предлагает систему хранения данных в облаке (SaaS), связь с другими интерфейсами (API) и общение с ERP-системами компаний; одна из проблем этого решения заключается в том, чтобы можно было отследить пищу, используемую в транспортных средствах.

Выводы

Вычислительная мощность в любой области или децентрализованной системе важна, так что меньше времени на обработку выполняется в каждой из транзакций, выполняемых в ней, еще одним элементом уравнения является интеграция с другими приложениями, которые помогают решать раз-

личные проблемы в зависимости от искомого решения, подтверждения транзакции и формы оплаты.

Ожидается, что новые платежные системы будут интегрированы посредством смарт-контрактов вместе с интеграцией виртуальных валют, напрямую конкурируя с традиционными компаниями, такими как Master Card или Visa; также стремиться к тому, чтобы эти транзакции были приняты и могли быть использованы участниками цепочки поставок.

Миграция компаний в цифровые экосистемы неизбежна, поэтому спрос на новые рабочие места в будущем или после пандемии требует профессионалов с аналитическими навыками, инновациями, технологическими навыками проектирования и решения проблем и т.д.

Дизайн бизнес-архитектуры, применяемой в сети блокчейн, требует глубоких знаний о роли, которую играет каждый из элементов в ней, а также взаимосвязях или потоках информации, транзакций и товаров в цепочке поставок в настоящее время изучается, однако процесс внедрения этой технологии должен происходить еще одним способом, а не микропроцессами в сети поставок, чтобы технология была масштабируемой и была достигнута возможность взаимодействия с другими технологиями.

Программирование в облаке даст возможность управлять сетями блокчейн, однако этап создания инфраструктуры является важной проблемой для разработчиков, исследователей и отраслей, где также необходимо оценить проблемы сетевой безопасности и скорости транзакций, с целью создания ценности для любой отрасли; Точно так же оценка технологий блокчейна, принятых в цепочке поставок, может быть выполнена путем применения KPI в рамках подтверждения цепочки.

Литература:

1. Особенности национальной цифровизации: TRAFFIC надежды.— Текст: электронный // LOGIRUS: [сайт].— URL: https://logirus.ru/articles/solution/osobennosti_natsionalnoy_tsifrovizatsii-_traffic_nadezhdy.html (дата обращения: 03.12.2020).
2. Всемирный, э.ф. Отчет о будущем рабочих мест 2020 / э.ф. Всемирный.— Текст: электронный // World Economic Forum: [сайт].— URL: http://www3.weforum.org/docs/WEF_Future_of_Jobs_2020.pdf, (дата обращения: 02.12.2020).
3. Системная архитектура для обеспечения прозрачности социальной устойчивости цепочки поставок на основе блокчейна // Робототехника и компьютерно-интегрированное производство / В. Г. Венкатеш.— Текст: электронный // Science Direct: [сайт].— URL: <https://doi.org/10.1016/j.rcim.2019.101896> (дата обращения: 03.12.2020).
4. Давила, К. Интернет вещей от шумихи до реальности / К. Давила, Я. Тарнов.— 2-е изд.— Швейцария: springer, 2019.— 269–297 с.— Текст: непосредственный.
5. Хело, П. Блокчейны в операциях и цепочках поставок: модель и эталонная реализация / П. Хело.— Текст: электронный // Scienc Direct: [сайт].— URL: <https://doi.org/10.1016/j.cie.2019.07.023> (дата обращения: 03.12.2020).
6. Дев, Н. К. Многокритериальная оценка ключевых показателей эффективности цепочки поставок в реальном времени с учетом архитектуры больших данных / Н. К. Дев.— Текст: электронный // Scienc Direct: [сайт].— URL: <https://doi.org/10.1016/j.cie.2018.04.012> (дата обращения: 08.12.2020).
7. Шадаб, М. Интеграция с технологией блокчейн / М. Шадаб.— Текст: электронный // ADEPTIA: [сайт].— URL: <https://adeptia.com/blog/integrating-blockchain-technology> (дата обращения: 25.11.2020).
8. Блокчейн в эталонной архитектуре.— Текст: электронный // Prosynergy: [сайт].— URL: <http://prosynergy.com.pe/2019/06/12/blockchain-arquitectura-referencia/> (дата обращения: 01.12.2020).
9. Блокчейн — как технология будущего будет использована в логистике.— Текст: электронный //: [сайт].— URL: <https://www.ablcompany.ru/news/blokcheyn-kak-tehnologiya-budushchego-budet-ispolzovana-v-logistike> (дата обращения: 01.12.2020).
10. Хева, Т. Обзор смарт-контрактов на основе блокчейна: приложения, возможности и проблемы / Т. Хева.— Текст: электронный // Science Direct: [сайт].— URL: <https://doi.org/10.1016/j.jnca.2020.102857> (дата обращения: 15.11.2019).

11. Санчес, П. внедрение блокчейной технологии в сети снабжения в колумбском контексте: специальность 23.04.01 «Технология транспортных процессов»: диссертация на соискание ученой степени кандидата технических наук / П. Санчес; ДГТУ. —, проект в настоящее время находится в разработке. — с. — Текст: непосредственный.
12. применение технологии блокчейн в цепочке поставок. — Текст: электронный // Hyperledger: [сайт]. — URL: <https://www.hyperledger.org/learn/publications/walmart-case-study> (дата обращения: 01.12.2020).
13. ОТИФ КПЭ — в срок в полном объеме. — Текст: электронный // QPR ProcessAnalyzer: [сайт]. — URL: <https://www.qpr.com/kpi/otif-on-time-in-full> (дата обращения: 02.12.2020).

ТЕХНИЧЕСКИЕ НАУКИ

Исследование и совершенствование методики улучшения качества питьевой воды на станции водоподготовки из поверхностного источника

Абилжанова Назгуль Хасенаевна, студент магистратуры;

Абдухан Алима Галымжанкызы, магистрант

Научный руководитель: Сидорова Наталья Васильевна, кандидат технических наук, ассоциированный профессор
Казахский национальный исследовательский технический университет имени К. И. Сатпаева (г. Алматы, Казахстан)

Водоподготовка — процесс сложный и требует тщательного продумывания. Существует очень много технологий и нюансов, которые прямо или косвенно повлияют на состав водоподготовки, поэтому разрабатывать технологию, оборудование, этапы следует очень тщательно. Для решения проблемы получения питьевой воды, отвечающей всем требованиям и СанПиН, синтезированы принципиально новые полимерные материалы, обладающие уникальными свойствами. Это полимеры пространственно-глобулярной структуры (ПГС-полимеры). ПГС-полимеры по эффективности и механизмам очистки заменяют все ранее известные фильтрующие материалы.

Ключевые слова: водоподготовка питьевой воды, поверхностные источники, полимеры пространственно-глобулярной структуры, фильтрация воды.

Водоподготовка питьевой воды очень важна. Источниками водоснабжения являются подземные и поверхностные источники, качества воды в которых из-за антропогенного воздействия в последнее время значительно ухудшилось. Потребление воды такого качества негативно воздействует как на здоровье человека, так и на инженерное оборудование жизнеобеспечения. Решение этих проблем обеспечивает применение технологического оборудования очистки воды. Так как состав и степень загрязнение воды изменяется в широком диапазоне, то технология очистки воды может включать различные типы и оборудования. Создание очистителей воды универсального применения является актуальной задачей. При этом важными являются как стоимостные, так и эксплуатационные показатели оборудования.

Главная цель водоподготовки питьевой воды — очистить воду от грубодисперсных и коллоидных примесей, солей жесткости.

Целью работы является модернизация технологий водоподготовки за счет дополнительного узла.

К поверхностным источникам относятся реки, озера, искусственные водохранилища, пруды. Общими свойствами воды поверхностных источников являются низкая минерализация, большое количество взвешенных веществ, высокий уровень микробного загрязнения, колебания расхода воды в зависимости от времени года и метеорологических условий. Величина активной реакции большинства поверхностных источников находится в диапазоне рН 6,5–8,5.

В последние годы практически на всей территории Республики Казахстан резко ухудшилось качество питьевой воды.

Загрязнены водозаборы, особенно функционируют старые, сильно корродирующие системы водопровода, в питьевую воду в больших количествах попадают отходы производств и жизнедеятельности человека. Загрязнено большинство колодцев, грунтовых и подземных вод. Во многих регионах существует опасность возникновения чрезвычайных ситуаций, связанных с залповыми выбросами вредных веществ и их попадание в питьевую воду со всеми вытекающими последствиями (массовые отравления, эпидемии и т.д.), прорывы дамб, наводнения, засухи, последствия ядерных взрывов, разрушения систем водоснабжения в результате землетрясений и других природных явлений. Такие чрезвычайные ситуации уже неоднократно возникали, хотя и не предавались широкой огласке (ртуть в воде канала Иртыш-Караганда, в грунтовых водах г. Павлодара, бор в питьевой воде г. Актюбинска, радионуклиды и цветные металлы в питьевой воде Восточно-Казахстанского экологическими катастрофами: засухи, последствия ядерных взрывов, Центрально-Казахстанского регионов, гербициды в питьевой воде Приаралья и т.д.).

Вероятность локальных катастроф с питьевой водой постоянно возрастает. В этой связи весьма высокоэффективных, компактных очистных устройств индивидуального и коллективного назначения, с помощью которых население имеет возможность получать доброкачественную питьевую воду из загрязненных источников (реки, озера, ручьи, арыки и т.д.).

Методы водоподготовки должны выбираться при сопоставлении состава исходной воды и ее качества, регламентированного нормативными документами или определенного

потребителем воды. После предварительного подбора методов очистки воды анализируются возможности и условия их применения, исходящие из поставленной задачи. Чаще всего результат достигается поэтапным осуществлением нескольких методов. Таким образом, важными являются как выбор собственно методов обработки воды, так и их последовательность.

В последние годы все большее значение в очистке водных и не водных систем приобретают мембранные методы — (электролиз, обратный осмос, ультрафильтрация, микрофильтрация, нанофильтрация). Обратный осмос (размеры пор 1–15 А, рабочее давление 0,5–8,0 МПа) применяется для деминерализации воды, задерживает практически все ионы на 92–99%, а при двухступенчатой системе и до 99,9%. Нанофильтрация (размеры пор 10–70 А, рабочее давление 0,5–8,0 МПа) используется для отделения красителей, пестицидов, гербицидов, сахарозы, некоторых растворенных солей, органических веществ, вирусов и др. Ультрафильтрация (размеры пор 30–1000 А, рабочее давление 0,2–1,0 МПа) применяется для отделения некоторых коллоидов (кремния, например), вирусов (в том числе полиомиелита), угольной сажи, разделения на фракции молока и др. Микрофильтрация (размеры пор 500–20000 А, рабочее давление от 0,01 до 0,2 МПа) используется для отделения некоторых вирусов и бактерий, тонкодисперсных пигментов, пыли активных углей, асбеста, красителей, разделения водо-масляных эмульсий и т.п.

Основные достоинства мембранных методов очистки — низкая энергоёмкость, высокая степень очистки, возможность удаления большой группы примесей, относительная простота аппаратного оформления. Основным элементом рассматриваемых устройств — разделительные мембраны (фильтры), их свойства и качество, которые определяют эффективность всего процесса. В настоящее время в процессах ультра-, микро- и гиперфильтрации используют в основном мембраны целлюлозного типа. Этим мембранам присущи серьезные недостатки: низкая химическая устойчивость, и относительно низкая удельная производительность.

Разработанные на основе пространственно-глобулярной структурных полимеров фильтры во многом лишены указанных недостатков. В них высокие разделительные свойства, высокая удельная производительность и химическая стойкость сочетаются с простотой синтеза, доступностью и дешевизной исходных материалов.

Иониты образуют большой класс фильтрующих ионообменных материалов. Применяются для умягчения, деминерализации, обескремнивания воды, выделения из водных растворов отдельных веществ и т.п. Иониты — твердые, практически нерастворимые полиэлектролиты, природные, искусственные или синтетические.

Синтетические иониты на органической основе — высокомолекулярные соединения, молекулы которых содержат ионогенные группы, способные к диссоциации и обмену подвижных ионов на другие ионы, содержащиеся в растворе.

Процесс ионного обмена приближенно можно представить в виде составляющих: перенос ионов раствора к поверхности гранулы ионита;

- диффузия ионов внутри гранулы;
- протекание реакции ионного обмена;
- диффузия вытесняемого из гранулы противоиона изнутри к поверхности гранулы;
- перенос противоиона от поверхности гранулы ионита в объем раствора.

Для решения проблемы получения питьевой воды, отвечающей требованиям, синтезированы принципиально новые полимерные материалы, обладающие уникальными свойствами. Это полимеры пространственно-глобулярной структуры (ПГС-полимеры) характеризуются узким распределением пор по размерам 10% и высокой химической активностью и стойкостью, а также обладают высокой механической прочностью. ПГС-полимеры по эффективности и механизмам очистки заменяют все ранее известные фильтрующие материалы. Это «ситовый» эффект, позволяющий задерживать механические примеси и крупные включения, зарядная фильтрация — интенсифицирует процесс очистки за счет действия электростатических сил, когда загрязнители, обладающие отрицательным зарядом (кроме Fe и Al), взаимодействуют с положительными или отрицательными зарядами микроглобул ПГС-полимеров (в зависимости от типа фильтроэлемента), ионообмен и сорбция, позволяющие связывать поливалентные катионы, к которым относятся катионы цветных и тяжелых металлов, и коалесценция, позволяющая нейтрализовать электрический заряд, присущий всем загрязнителям, и во много раз интенсифицировать процесс осаждения частиц. Такая комбинация механизмов очистки воды в одном устройстве позволяет создать эффективные компактные очистители малых габаритов и с низкими эксплуатационными показателями.

Технология получения изделий из ПГС-полимеров существенно отличается от технологии получения обычных известных полимеров. В реакторе готовят реакцию смесь, где основным по массе компонентом является растворитель, а затем в специальных формах протекает формирование изделий глобулярной структуры в режиме спонтанного глобулообразования. Сырьем для ПГС-полимеров служат обычно используемые мономеры: фенолы, амины, альдегиды и т.д.

Наиболее эффективно применение ПГС-полимеров в качестве материалов технологического назначения в таких процессах как ионный обмен, сорбция, фильтрация, микро-, ультра- и гиперфильтрация, коалесценция, диспергирование. Пространственно-глобулярные фильтры с порами от 0,01 до 20 мкм работают при скоростях пропускания 800–3000 уд./об/ч в режимах фильтрации ионного обмена.

Эффективность работы ПГС-ионитов в 100 и более раз выше, чем обычных зерненных ионитов. При этом содержание твердых частиц в очищенной жидкости не превышает 0,2 мг/дм³, а концентрация растворимых веществ находится в пределах от 0,1 до 1,0 мг/дм³. Степень концентрирования извлекаемого вещества в элюат составляет 600–100000 в зависимости от характера очищаемого раствора и вида вещества.

ПГС-полимеры представляют собой или объемные изделия: листы, ленты, трубы, стержни, блоки различной конфигурации кубического объемом до нескольких десятков сантиметров. Заданная структура достигается процессе поли-

меризации или поликонденсации исходной гомогенной смеси веществ с специальных условиях под влиянием физико-химических коллоидно-химических факторов. В итоге образуется полимерный материал, состоящий из микроглобул заданного размера, соединенных между собой в точках контакта.

Пористость можно варьировать в пределах 35–90%, а размеры пор задавать в диапазоне 0,01–20 мкм. Технологией процесса обеспечивается отклонение размера пор от заданного номинала не более ±10%.

Площадь внутренней поверхности ПГС-фильтров достигает 150 м² на 1г пространственно-глобулярного изделия. ПГС-фильтры изготавливают из стирола, винилпиридина, метилметакрилата, акрилонитрила, бутадиена, фенолов, карбамида, меламина, альдегидов, эпоксидных соединений.

На сегодняшний день успешно применяется и освоен выпуск 10 типов ПГС-полимеров, которые обладают однотипной структурой и пористостью, но отличаются наличием характерных функциональных группировок (ионообменные, электроннообменные, комплексообразующие, коалесцентные), которые придают ПГС-полимерам определенные свойства.

Конструктивное исполнение фильтроэлементов позволяет изготавливать компактные фильтрационные установки,

производительность которых определяется количеством размещенных в них ФЭЛ. В зависимости от назначения и особенностей эксплуатации установки очистки воды на базе картриджных ФЭЛ из ПГС-полимеров комплектуются корпусами, выполненными из нержавеющей стали или пищевой пластмассы.

Освоен выпуск фильтрационных установок (патронных фильтров ФП) следующих типов и производительностей:

- ФУ производительностью 0,25 м³/час в пластмассовом корпусе УФМ-0.25П;
- ФУ производительностью 0,3 м³/час в пластмассовом корпусе УФМ-0.3П
- ФУ производительностью 0,5 м³/час в корпусе из нержавеющей стали УФМ-0,5НВ;
- ФУ производительностью 1,2 м³/час в корпусе из нержавеющей стали УФМ-1,2Н;
- ФУ производительностью 15,0 м³/час в корпусе из нержавеющей стали УФМ-15,0Н;
- Фильтрационные установки производительностью свыше 1,2 м³/час компонуются из модулей УФМ-1,2Н;
- Фильтрационные установки производительностью свыше 15,0 м³/час компонуются из модулей УФМ-15,0Н.

Таблица 1

Тип установки	ФП-1	ФП-2	ФП-3	ФП-4	ФП-5	ФП-6
Количество фильтрующих элементов, шт.	1	2	3	4	5	6
Производительность, м ³ /ч	0,3	0,6	0,9	1,2	1,5	1,8
Поверхность фильтрации, м ²	0,05	0,10	0,15	0,20	0,25	0,30
Режим работы	непрерывный, циклический					
Продолжительность регенерации, мин., не более	8–10					
Рабочее давление исходной среды, МПа, не более	0,3					
Температура исходной среды, град. С, не более	100					

Выводы

Решение проблемы доброкачественной питьевой водой — может быть достигнуто путем использования специальных видов ПГС-полимеров. Широкий набор ПГС-полимеров различной геометрической формы и размеров, химического состава, параметров пористой структуры, сорбционных

и ионообменных свойств позволяет создать на их основе весь потребный спектр проблем очистных устройств, необходимых для решения обеспечения населения доброкачественной питьевой водой в практически любых условиях.

Перспективность применения ПГС-фильтров: простая технология производства; возможность переработки больших объемов растворов при сохранении высокой селективности процесса;

Литература:

1. Елизарова Т. В., Михайлова Л. А. Гигиена питьевой воды, учебное пособие, Чита, 2007. — 4с.
2. Беликова С. Е. Библиотека Акватерм Водоподготовка, 2007
3. Nino Zinna Методы фильтрации воды в системах водоподготовки — библиотека научных статей, из журнала RCI № 4/2005 https://www.abok.ru/for_spec/articles.php?nid=2975
4. Материалы четвертого Международного конгресса и технической выставки «Вода: экология и технология» «ЭКВАТЕК — 2000» (29 мая — 2 июня 2000 г.).-М., 2000.
5. Луков А. Н., Горбачев Е. А., Дементьев В. Н. Метод определения оптимальной технологии на водопроводной станции с помощью функций Кобба-Дугласа. //Городское хозяйство и экология. 2000. — № 2. — С. 43–48.
6. СанПиН 2.1.4.10749–01 «Питьевая вода. Гигиенические требования к качеству воды
7. Елизарова, Т. В. Гигиена питьевой воды: уч. пособие / Т. В. Елизарова, А. А. Михайлова. — Чита: ЧГМА, 2014. — 63 с.

Способ повышения точности обработки сигнала рефлектометра

Авдеюк Данила Никитович, аспирант
Волгоградский государственный технический университет

При использовании рефлектометра для определения места повреждения на линии электропередачи есть необходимость выделения значения сигнала с удаленных участков, который имеет свойство затухать. В связи с чем, есть необходимость приведение сигнала к одному значению по амплитуде. В работе рассматривается процесс приведения сигнала к одному виду с повышением точности путем введения блока автоматической регулировки.

Ключевые слова: рефлектометр, блок автоматического регулирования усиления, погрешность, точность.

Процесс дискретизации сигнала, получаемого в результате работы рефлектометра, происходит путем заполнения тактовыми импульсами полученный сигнал с устройства.

Согласно теореме Котельникова, всякий непрерывный сигнал, имеющий ограниченный частотный спектр, полностью определяется своими дискретными значениями в моменты отсчета, отстоящие друг от друга на интервалы времени [1, с. 27] (рисунок 1).

В реальном сигнале фронты прямоугольного сигнала имеют не мгновенные возрастания и затухания, то в систему необходимо добавить триггер Шмитта, который будет срабатывать по заданной амплитуде сигнала, тем самым выделять начало и окончания прямоугольного импульса (рисунок 2).

В связи с тем, что в зависимости от дальности прохождения сигнала меняется его амплитуда, отсюда возникает проблема выделения начала и окончания прямоугольного сигнала из реального. Так же, т.к. триггер Шмитта срабатывает по строго заданному порогу, то при прохождении сигнала с малой амплитудой, этот сигнал будет теряться [2, с. 80] (рисунок 3).

Также стоит отметить, что в зависимости от дальности прохождения сигнала меняется и значение его амплитуды, т.е. происходит затухание сигнала (рисунок 4).

В связи с чем, возникает необходимость в стабилизации амплитуды возвращенного сигнала, т.е. нужно использовать автоматическую регулировку усиления (АРУ). Таким образом, АРУ должна обеспечить относительное постоянство напряжения сигнала на входе.

АРУ (рисунок 5) содержит в себе усилитель информационного сигнала (1), усилитель контура обратной связи по аддитивной компенсации дрейфа нуля (2), является блоком вычитания из информационного сигнала в нулевую составляющую, усилитель контура обратной связи автоматической регулировки усиления, выполняющий функцию элемента сравнения и усилителя (3), RC цепь

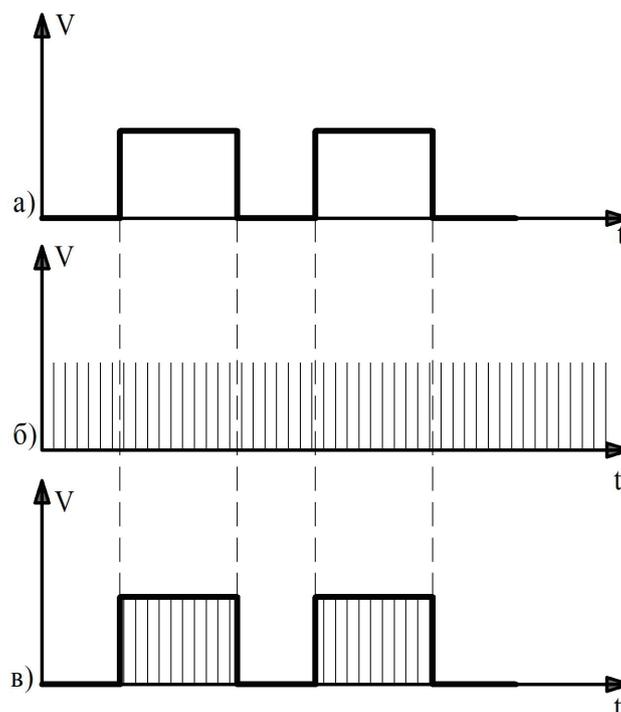


Рис. 1. Пример заполнения тактовыми импульсами:

а) Прямоугольный сигнал от рефлектометра; б) Тактовые импульсы; в) Заполнение импульсами сигнала

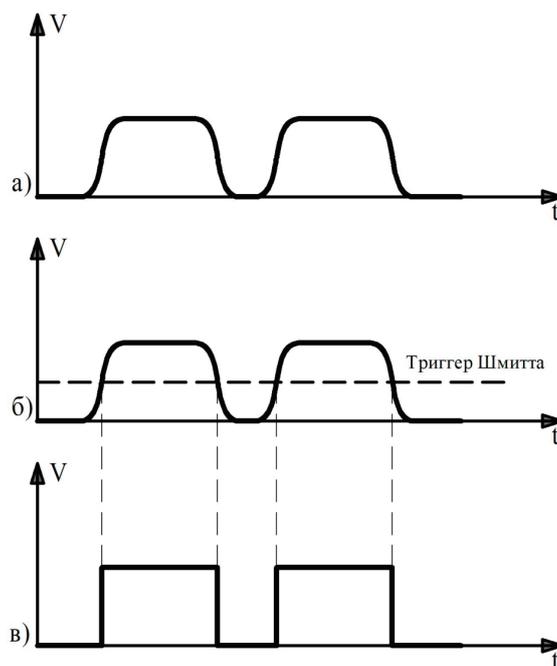


Рис. 2. Преобразования реального сигнала в прямоугольный:
 а) Реальный сигнал; б) введение триггера Шмитта; в) Прямоугольный сигнал

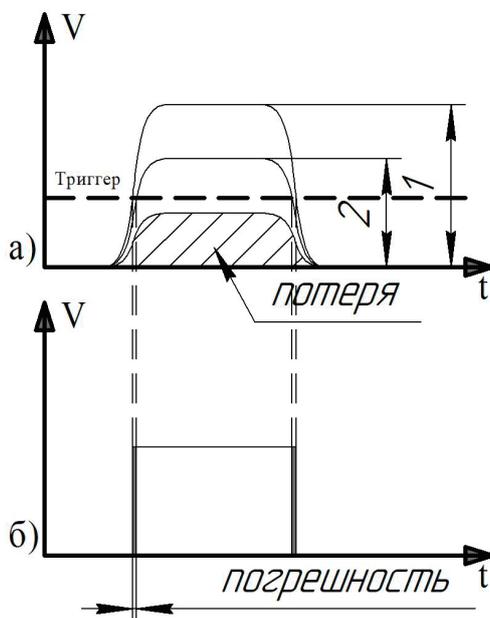


Рис. 3. Потери и погрешности: а) 1-сигнал на короткой дистанции, 2-сигнал на средних дистанциях, потеря — сигнал на дальних дистанциях, который имеет амплитуду ниже порога срабатывания триггера; б) демонстрация погрешности

пикового детектора, который преобразует максимальное значение сигнала относительно напряжения питания (4), т.е. постоянную составляющую-нулевую сигнала относительно общей шины, которая с помощью установленной обратной связи должна поддерживаться на постоянном малом значении, RC цепь пикового детектора, который преобразует максимальное значение сигнала относительно общей шины в постоянную составляющую, которая при помощи АРУ должна поддерживаться постоянно (5), оптрон, который предназначен для гальванической развязки управляющего сигнала и резистора в цепи обратной связи усилителя, т.е. выполняет функцию управления коэффициентом усиления усилителя (6), потенциометр, который выполняет функцию задатчика амплитуды импульсного сигнала (Рздч) [3].

Благодаря блоку АРУ появляется возможность привести разные значения амплитуд к единому значению, которое позволит триггеру стабильно срабатывать. (рисунок б)

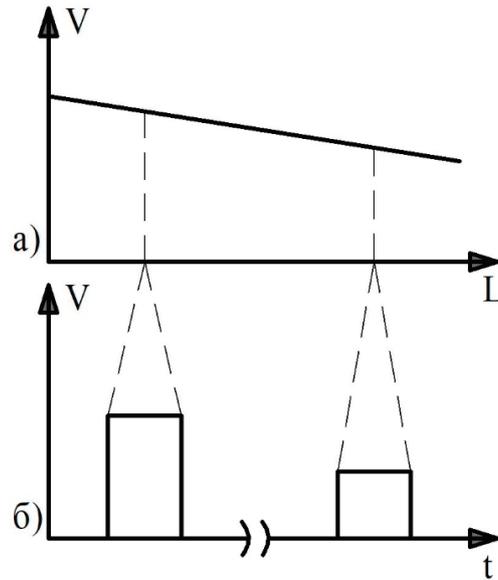


Рис. 4. Затухание сигнала: а) демонстрация затухания сигнала в зависимости от расстояния L ; б) демонстрация импульсов

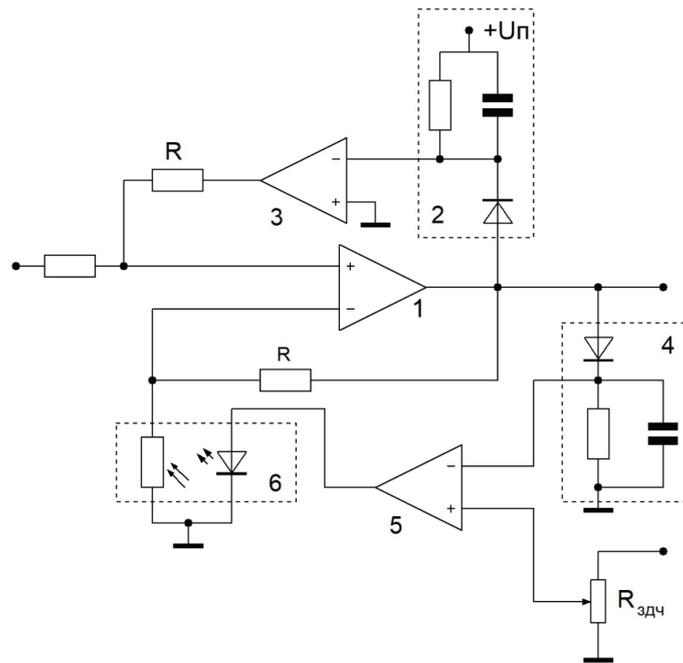


Рис. 5. Блок АРУ

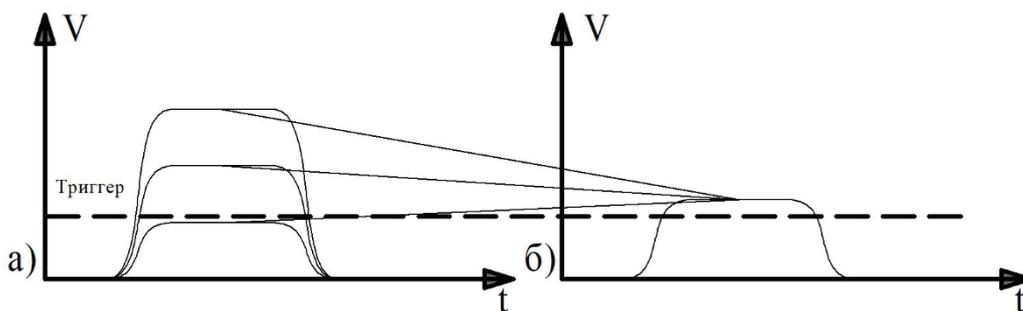


Рис. 6. Стабилизация значения сигнала: а) сигналы до блока АРУ; б) сигналы после блока АРУ

В результате введения блока АРУ удастся добиться стабильного значения амплитуды сигнала, в результате чего, достигается повышение точности.

Литература:

1. Шилин, А. Н. Автоматическая коррекция методической погрешности в импульсных рефлектометрах = Automatic correction of methodological errors in a reflectometer / А. Н. Шилин, А. А. Шилин, Н. С. Артюшенко // Актуальные проблемы электронного приборостроения (АПЭП-2016): матер. междунар. науч.-техн. конф. (г. Саратов, 22–23 сент. 2016 г.) / редкол.: А. А. Захаров (отв. ред.) [и др.]; Саратовский гос. техн. ун-т им. Гагарина Ю. А., IEEE.— Саратов, 2016.— Т. 2.— С. 26–29.
2. Гонжал, М. И. Измерительные преобразователи с автоматической коррекцией погрешности / М. И. Гонжал, А. Н. Шилин // Научные исследования и их практическое применение. Современное состояние и пути развития 2007: сб. науч. тр. по матер. междунар. науч.-практ. конф., 1–15 окт. 2007 г. / Одес. нац. морской ун-т [и др.].— Одесса, 2007.— Т. 2.— С. 78–82.
3. Пат. 2712771 Российская Федерация, МПК G01R31/11 Интеллектуальное устройство для измерения расстояния до места повреждения линий электропередачи / А. Н. Шилин, А. А. Шилин, Над, С. Кузнецова, Д. Н. Авдеюк; ФГБОУ ВО ВолгГТУ.— 2020.

Разработка системы орошения на приусадебном участке в рамках проекта «Умный дом»

Боровикова Анастасия Сергеевна, студент магистратуры;

Вдовина Алена Олеговна, студент магистратуры;

Медведева Людмила Николаевна, доктор экономических наук, профессор

Волжский политехнический институт (филиал) Волгоградского государственного технического университета

В статье представлена разработка организации капельного орошения кустов винограда на приусадебном участке, приведены расчеты стоимости проекта.

Ключевые слова: умные технологии, капельное орошение, автоматический полив, экономическое обоснование проекта.

Сегодня все чаще мы употребляем слова: умные технологии, умные вещи, умный дом. Умный дом — это комплексная система автоматизации управления различными устройствами, в том числе и на приусадебном участке. Умные технологии повышают уровень комфорта и безопасности человека, обеспечивают экономию ресурсов, в том числе и воды [1, 2]. Чтобы добиться хорошего урожая необходимо обеспечить полив растений с учетом оросительной нормы. Определить величину оросительной нормы нетто в конкретном году t можно с помощью уравнения:

$$M_t^{op} = \frac{R_t - \bar{R}_{0t} \times L (M_{Mt} + O_{Cl})}{\bar{R}_{0t} \times L} \times 100, \quad (1)$$

где, M_t^{op} — оросительная норма, м³/га;

R_t — радиационный баланс деятельной поверхности, в году t , кДж / см² в год;

M_{Mt} — количество влаги, полученное за счет орошения, см / га;

\bar{R}_{0t} — «индекс сухости» в естественных условиях.

Расход воды на формирование 100 кг урожая принято называть коэффициентом водопотребления. Орошение положительно влияет на экологические факторы, которые определяют рост и развитие винограда. Для формирования 100 кг урожая виноградному растению в условиях Волгоградской области потребуется: 20–30 м³ Один из вариантов полива — использование капельного орошения, полив винограда капельными

трубками, которые можно разместить в закрытом грунте или на поверхности, прикрепив к лозе. Целью исследования является разработка автоматической схемы полива винограда, размещенного на расстоянии 1 м друг от друга в проекте «Умный дом». В ходе исследования был подготовлен проект капельного орошения приусадебного участка (рис. 1) [3,4,5].

К скважине подключается насосная станция, которая подает воду в емкость объемом 1000 л (объем емкости был выбран из расчета: сделано 20 гребней по 10 м вдоль которых проходит лента с шагом 30 см, водовыход каждого отверстия ленты 1 л/сутки), насос будет поддерживать нужное давление в системе, как только откроется кран — давление упадет, и насос автоматически включится, начиная подавать воду в емкость. Когда кран закроется, то давление подачи воды повысится, насос автоматически отключится. В связи с этим, установка емкости на высоту 1,5–3 м не требуется. В бак врезают переходник, на который накручен шаровый кран диаметром 32 мм. После крана устанавливается дисковый фильтр тонкой очистки для предотвращения засорения линий капельного полива. После фильтра устанавливается контроллер, отвечающий за автоматический полив — это может быть электромагнитный клапан, облаченный в герметичный корпус с электронным таймером. Благодаря его настройкам можно установить необходимую периодичность полива растений, не только на один день, но и на целый поливочный сезон. Такой способ можно отнести к системе «Умный дом», которая благодаря специальным датчикам с чувствительным элементом

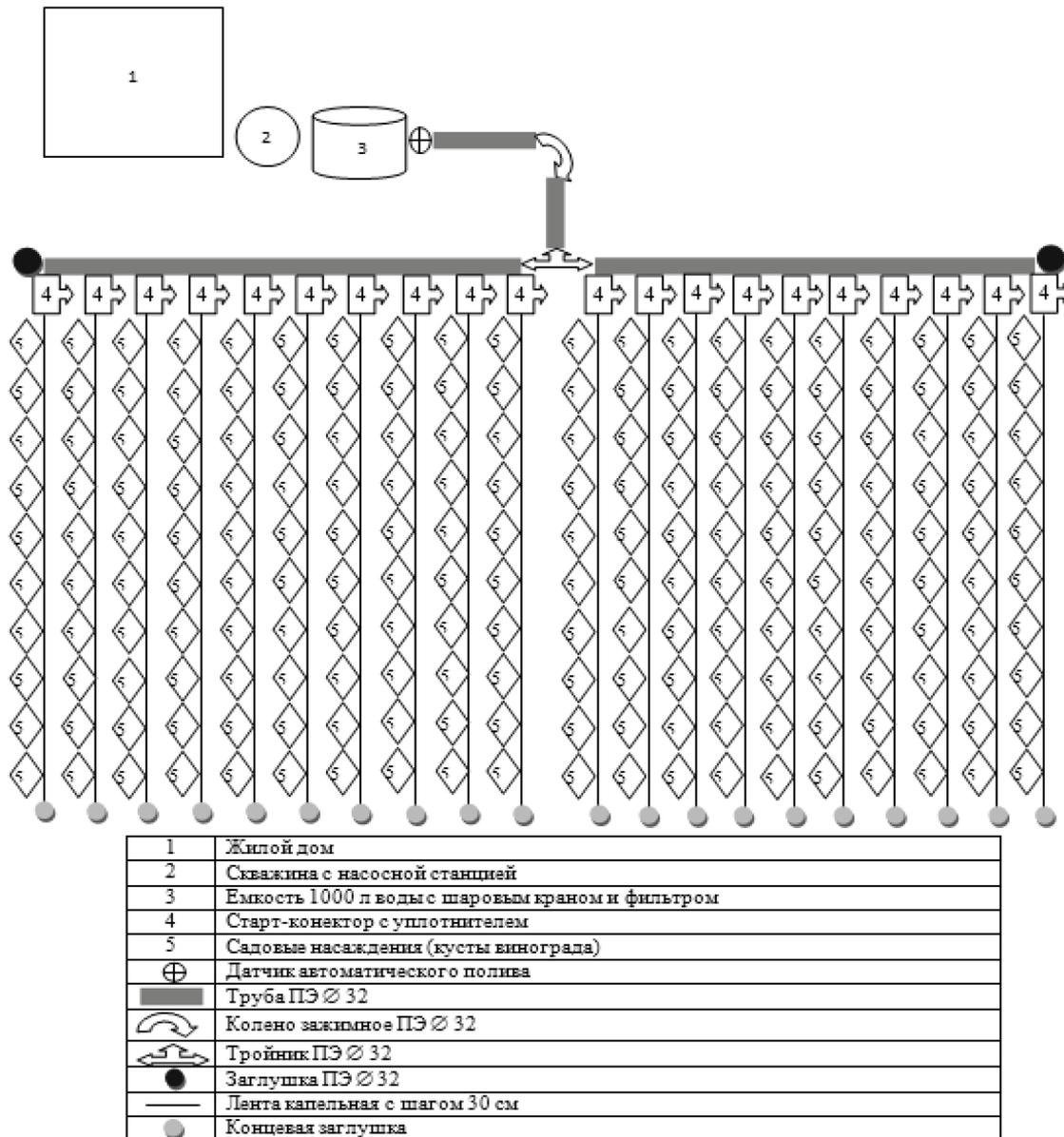


Рис. 1. Схема автоматической системы капельного орошения

анализирует данные о влажности, температуре, влажности, осуществляет полив [2]. Учитываются все нюансы полива: температура воды, влажность воздуха, влажность почвы, время выключения. Система полива запускается в четко указанное время дня или ночи. Следующий этап — установка магистральных труб, которыми могут послужить полиэтиленовые поливочные трубы. Их стыкуют к системе подачи воды разъемными соединениями. Используется полиэтиленовая труба диаметром 32 мм, которая с помощью фитингов прокладывается в нужных направлениях. В качестве фитингов используют соединительные муфты, углы, тройники. В трубе проделываются отверстия диаметром 14 мм, к которым прикрепляется капельная лента на расстоянии 1 метра друг от друга. Капельная лента соединяется с магистральной трубой с помощью старт-конектора благодаря уплотнительной прокладке с одной из сторон. Капельная лента используется с шагом 30 см и прокладывается отверстиями вверх, откуда

происходит орошение земли. На конце каждой ленты крепится заглушка, не позволяющая течь воде за пределы ленты. При необходимости можно заглушить магистральную трубу по такому же принципу [4,5]. В случае деформации капельной ленты используется ремонтный фитинг.

В рамках исследования был произведен расчет стоимости необходимых материалов и работ по установке системы автоматического полива (табл. 1) Выбор системы полива в данном проекте определяется тем, что способствует исчезновению болезней растений, которые распространяются за счет наличия большой влажности или ее отсутствия, что ведет к увеличению урожая в 1, 5 раза. Автоматизация процесса ведет к уменьшению времени работы на участке, данные с датчиков поступают на компьютер или смартфон, и в удаленном доступе можно оценивать ситуацию для принятия решения. Умные технологии обеспечивают комфортную жизнь человека после трудового дня [2].

Таблица 1. Стоимость проекта по капельному поливу

№	Наименование	Цена, руб.	Кол-во, шт.	Стоимость, руб.
1	Насосная станция 370W 2400 л/час	10919	1	10919
2	Емкость (бак) 1000 л	11000	1	11000
3	Шаровый кран Ø 32	170	1	170
4	Дисковый фильтр для капельного полива	499	1	499
5	Датчик автоматического полива	2812	1	2812
6	Труба ПЭ Ø 32	32	30	960
7	Колено ПЭ Ø 32	72	1	72
8	Тройник ПЭ Ø 32	100	1	100
9	Заглушка ПЭ Ø 32	40	2	80
10	Старт-конектор с уплотнителем	40	20	800
11	Заглушка капельной ленты	20	20	400
12	Лента капельная с шагом 30 см	8	200	1600
Итого: 29412				
Строительно-монтажные работы				
1	Первичный выезд специалиста на участок			1000
2	Проектирование системы автоматического полива			6000
3	Монтаж системы автоматического полива			20000
Итого: 27000				
Обслуживание системы				
1	Запуск системы автоматического полива перед сезоном			4000
ВСЕГО: 60412				

Литература:

- Green Technologies: The Basis for Integration and Clustering of Subjects at the Regional Level of Economy / V. V. Melikhov, A. A. Novikov, L. N. Medvedeva, O. P. Komarova // Integration and Clustering for Sustainable Economic Growth. — Springer International Publishing AG. 2017. P. 365–382.
- Конвергентная платформа smart agriculture и применение цифровых информационных технологий в агробизнесе / Гурина И. В., Медведев А. В., Медведева Л. Н., Рогачев А. Ф. // Учет и статистика. 2019. № 3 (55). С. 74–84.
- Худайев И., Фазлиев Ж., Шаропов Н. Капельное орошение — как водосберегающий способ орошения садов и виноградников / Школа науки. М.: Издательство «ИП Романов Александр Николаевич», 2019. № 4. С. 17–18.
- Капельное орошение: практика применения. [Электронный ресурс].— URL: <https://webferma.com/rasteniievodstvo/sistemi-orosheniya/kapelniy-poliv.html>.
- Руководство по проектированию и установке систем капельного полива. — [Электронный ресурс]. URL: <https://hunter.ua/images/Documentation/Proectirovanie/kapelni-poliv.pdf>.

Способы воздействия на эффективность работы трехфазного сепаратора

Золотов Михаил Александрович, студент магистратуры
Астраханский государственный технический университет

В статье систематизированы основные факторы, оказывающие воздействие на эффективность сепарации газожидкостной смеси. Даны рекомендации по повышению эффективности процесса сепарации.

Ключевые слова: трехфазный сепаратор, эффективность сепарации, газожидкостный поток, газ, смесь.

В газовой отрасли трёхфазные сепараторы (далее — ТФС) предназначены для разделения и очистки природной газожидкостной смеси, содержащей газ, жидкие углеводороды, воду и механические примеси.

Механизм работы большинства ТФС основан на гравитационном разделении газожидкостного потока на разные по плотности фракции: тяжелую (вода), среднюю (жидкий углеводородный конденсат) и легкую (углеводородные газы). Также

в комбинации зачастую применяются механизмы инерциального и циклонного действия для газовой фазы.

Принципиальным отличием от двухфазного сепаратора является образование нестабильной зоны — дисперсной жидкости, состоящей из диспергированных тяжелой и средней фаз (очень мелких перемешанных между собой капелек воды и жидкого концентрата либо нефти), которые впоследствии за счет процессов коалесценции (слияния однородных частиц) переходят в свои непрерывные фазы. При этом находящиеся в газожидкостном потоке пузырьки газа, изначально растворенные во всех фазах, поднимаются вверх, последовательно преодолевая сопротивление воды, дисперсной жидкости и жидких углеводородов.

Таким образом, основными направлениями повышения эффективности работы ТФС являются интенсификация процесса газоотделения и улавливание капелек жидких углеводородов, уносимых с газом. Вода как фаза с наибольшей плотностью легче поддается сепарации, и той ее частью, которая уносится

с газом и жидким углеводородным концентратом, можно пренебречь.

Лабораторным путем, а также посредством математического и компьютерного термодинамического моделирования выявлены факторы и процессы, воздействуя на которые можно влиять на эффективность работы сепаратора [1–4]:

1. Гидродинамические и термодинамические параметры потока:

а) структура потока, т.е. взаимное расположение и распределение в сечении сепаратора фаз с различной плотностью. Она влияет на взаимодействие фаз на границах раздела, а также на образование и форму внутренних течений и связана с проблемами расслоенного турбулентного движения смеси.

Как показывают исследования, наибольшая эффективность сепарации достигается при уменьшении величины объемного газосодержания (β) до значений, соответствующих первому режиму течения (см. рис. 1).

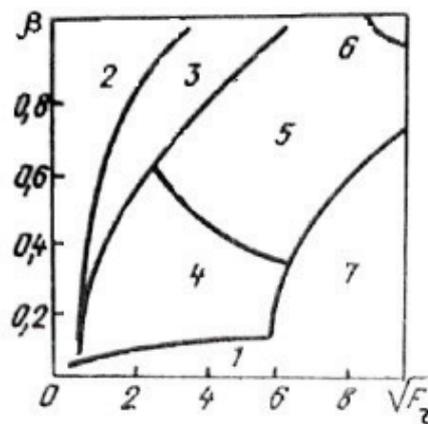


Рис. 1. Режимы течения газожидкостных смесей по Бакеру [1]
Примечание: β — объемное газосодержание; F_r — критерий Фруда.

б) состав смеси и физико-химические свойства ее компонентов (плотность, вязкость, поверхностное натяжение, газовый фактор, содержание воды, смол, парафинов, солей и др. веществ, молярные доли азота, метана и др. газов в газовой фазе и т.д.).

Воздействие на физико-химические свойства газожидкостного потока предусматривает снижение вязкости и поверхностного натяжения смеси. Чем ниже указанные параметры, тем эффективнее процесс сепарации.

Особо следует отметить поверхностное натяжение. Оно влияет на размер частиц жидкости в газе (обратная зависимость), а также на прочность жидкостных пленок (чем меньше поверхностное натяжение, тем легче потоку газа унести капельки жидкости из сепаратора). Поскольку поверхностное натяжение не поддается коррекции в промышленных условиях, его нельзя рассматривать как фактор повышения эффективности сепарации;

в) режим потока.

Зная режим потока (см. рис. 2), можно спрогнозировать объемное содержание жидкости, используя следующие формулы:

$$\lambda_L = \frac{q_L}{q_L + q_g}, \quad (1)$$

$$H_L = \frac{a \cdot \lambda_L^b}{F_r^c}, \quad (2)$$

где λ_L и H_L — объемы содержания жидкости соответственно при отсутствии или наличии проскальзывания фаз ($H_L \geq \lambda_L$);

q_L — объемный расход жидкой фазы;

q_g — объемный расход газовой фазы;

F_r^c — число Фруда для смеси;

a, b — эмпирические коэффициенты.

г) профиль скорости потока.

Увеличение средней скорости потока в свободном сечении сепаратора снижает эффективность сепарации, т.к. чем выше расход смеси, тем большее количество капелек жидкости не успевает отделиться и уносится с газом;

г) градиент давления и температурный режим.

Согласно законам Рауля и Дальтона, с увеличением давления системы уменьшается молярная концентрация компонента в газовой фазе при одновременном ее возрастании в жидкой. Согласно формуле Стокса, скорость всплытия пузырьков газа находится в прямой зависимости от квадрата диаметра этих пузырьков. Следовательно, чем выше давление в се-



Рис. 2. Горизонтальные режимы потока по Беггу и Брилли [1]

параторе, тем медленнее процесс отделения мелких и тяжелых пузырьков газа от жидкой фазы.

Температура влияет на процесс в обратном направлении. Увеличение температуры ускоряет сепарацию газа за счет роста константы фазового равновесия. Зависимость констант фазового равновесия углеводородов от температуры показана на рис. 3.

Кроме того, повышение температуры смеси приводит к снижению ее вязкости и, следовательно, к увеличению скорости всплытия пузырьков газа. Поскольку классический ТФС не является температурным сепаратором, в промышленных условиях для повышения температуры газожидкостной смеси осуществляют ее подогрев до ввода в сепаратор.

2. Интенсивность газоотделения:

- а) скорость всплытия пузырьков газа.
- б) время роста пузырьков газа;
- в) интенсивность кипения газа.

Как правило, скорость всплытия пузырьков газа не превышает 20 см/с, и зависит она от воздействия динамических сил (подъемной силы и силы гидродинамического сопротивления) и сил поверхностного натяжения на границах преодолеваемых фаз различной плотности. При движении газожидкостного потока происходит непрерывное зарождение новых и рост уже образовавшихся пузырьков газа. Время роста пузырьков до видимых размеров настолько мало ($\tau = 10^{-4}$ с), что его можно не учитывать в практических расчетах и полагать, что после попадания в сепаратор пузырьки газа всплывают примерно за 40–60 с [4].

На практике из сепаратора уносится некоторое количество растворенного газа даже при условии пребывания газожидкостной смеси в сепараторе вчетверо дольше указанного срока. Это можно объяснить «интенсивным перемешиванием газожидкостной смеси и дроблением пузырьков на входе в сепаратор, а также образованием внутри сепаратора завих-

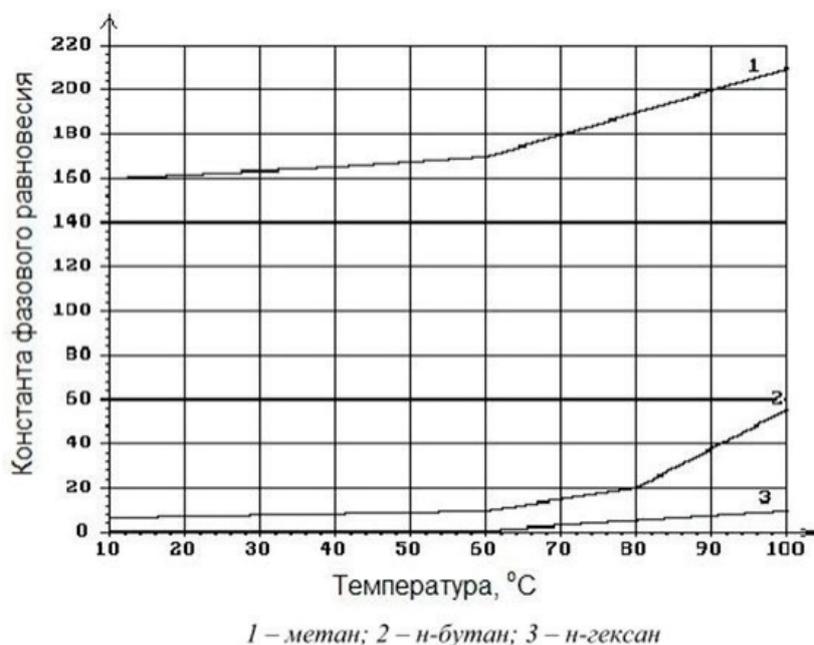


Рис. 3. Зависимость константы фазового равновесия компонентов смеси от температуры при постоянном давлении (P = 0,1 Мпа) [4]

рений и местных течений большой скорости, препятствующих всплытию пузырьков» [4].

Интенсивность кипения газа зависит от гидродинамических и термодинамических параметров потока, а также сил поверхностного натяжения. Чем больше число центров парообразования, тем пересыщеннее смесь и, как следствие, выше частота отрыва пузырьков и интенсивнее процесс дегазификации жидких фаз.

Таким образом, для повышения эффективности сепарации газа следует обеспечить более плавные ввод и движение смеси в сепараторе.

3. Время пребывания газожидкостной смеси в сепараторе.

Увеличение времени задержки смеси в сепараторе (в пределах технологического режима) повышает эффективность его работы.

4. Интенсивность перемешивания газожидкостной смеси в сепараторе.

Интенсивное перемешивание смеси усиливает турбулентность потока и ухудшает его структуру, что отрицательно воздействует на процессы коалесценции.

6. Конструктивные параметры сепаратора.

Совершенствование конструкции сепаратора за счет введения в его секции дополнительных устройств способствует более эффективному:

а) выделению и отводу газа (вводятся отражательные устройства, щелевой дегазатор);

б) улавливанию капелек жидкого концентрата, уносимых газом (фильтры, конденсатоотводчики, отбойные устройства);

в) снижению турбулентности потока (пластины, щелевой дегазатор, цилиндрические и полуцилиндрические поверхности).

Мероприятия по повышению эффективности работы ТФС, основанные на воздействии выделенных выше факторов, представлены в таблице 1.

Таблица 1. Мероприятия по повышению эффективности работы трехфазного сепаратора (разработано автором)

Фактор	Характер воздействия на эффективность	Важность в управлении эффективностью	Мероприятия по повышению эффективности сепарации
1. Гидродинамические и термодинамические параметры потока: 1.1. Структура потока	Н	+	Снижение турбулентности потока, ввод реагентов-дезмульгаторов
1.2. Состав смеси и физико-химические свойства ее компонентов: — плотность пластовой смеси	0	+	Подогрев смеси, понижение давления
— динамическая вязкость смеси	0	+	Подогрев смеси
— поверхностное натяжение смеси	0	–	–
— содержание смол, парафинов, солей, механических примесей	Н	–	–
1.3. Режим потока	Н	+	Снижение турбулентности потока за счет конфигурации трубопровода перед входом в сепаратор, а также конструктивных параметров сепаратора
1.4. Среднерасходная скорость движения смеси	0	+	Контроль скорости ввода и движения смеси в сепараторе
1.5. Режим давления (на входе/ внутри/ на выходе из сепаратора)	0	+	Контроль давления (пониженное и равномерное в пределах технологического режима)
1.6. Температурный режим	П	+	Подогрев смеси перед вводом в сепаратор
2. Интенсивность газоотделения 2.1. Скорость всплытия пузырьков газа	П	+	Воздействие на термодинамические и гидродинамические параметры потока

Фактор	Характер воздействия на эффективность	Важность в управлении эффективностью	Мероприятия по повышению эффективности сепарации
2.2. Время роста пузырьков газа	0	–	–
2.3. Интенсивность кипения газа	П	+	Воздействие на термодинамические и гидродинамические параметры потока
3. Время задержки смеси в сепараторе	П	+	Обеспечение оптимального времени задержки в пределах технологического режима (находится расчетным путем)
4. Интенсивность перемешивания смеси	0	+	Более плавный ввод смеси в сепаратор, контроль скорости потока, контроль пульсации потока, снижение турбулентности, поддержание оптимального давления на выходе из сепаратора
5. Конструктивные параметры сепаратора	П	+	Введение дополнительных устройств (каплеуловительных, газоотводных, успокаивающих) в соответствующие секции сепаратора; предварительный отбор газа до ввода смеси в сепаратор, отдельный ввод фаз в сепаратор

Примечания: О/П — соответственно обратный/прямой характер воздействия на эффективность сепарации; Н — трудно установить зависимость между величиной фактора и эффективностью сепарации; «–» — величиной фактора можно пренебречь, либо она мало поддается управлению; «+» — величина фактора важна и поддается управлению.

По результатам исследования можно сделать вывод, что подходы к управлению эффективностью работы трехфазного се-

паратора основаны на воздействии на факторы прямого и обратного влияния, а также совершенствовании конструкции трубопровода до входа в сепаратор и конструктивных параметров самого сепаратора. Комплексное внедрение предлагаемых мероприятий позволит повысить сепарацию газа и уменьшить унос газом капелек жидких углеводородов в промышленных условиях.

Литература:

1. Борис А. А., Лягов А. В. Определение режима течения потока газожидкостной смеси в трубопроводах на установках путевого сброса воды Арланской группы месторождения ОАО «АНК»Башнефть» // Нефтегазовое дело.— 2012.— № 2 — с. 66–78. [Электронный ресурс]. Режим доступа: <http://ogbus.ru/article/view/opredelenie-rezhima-techeniya-potoka-gazozhidkostnoj-smesi-v-truboprovodax-na-ustanovkax-putevogo-sbrosa-vody-arlanskoj-gruppy-mestorozhdenij-oao-ank-bashneft/> (дата обращения 05.12.2020 г.).
2. Леонтьев С. А. Галикеев Р. М. Тарасов М. Ю. Технологический расчет и подбор стандартного оборудования для установок системы сбора и подготовки скважинной продукции: учебное пособие.— Тюмень: ТюмГНГУ, 2015.— 124 с.
3. Пятахин М. В. Гидродинамическая модель газожидкостного потока в скважине для импортозамещения коммерческого программного обеспечения // Территория «Нефтегаз».— 2016.— № 5 (май).— С. 34–42.
4. Ушева Н. В. Технологические основы и моделирование процессов промышленной подготовки нефти и газа: учебное пособие / Н. В. Ушева, Е. В. Бешагина, О. Е. Мойзес, Е. А. Кузьменко, А. А. Гавриков; Томский политехнический университет.— 2-е изд.— Томск: Изд-во Томского политехнического университета, 2013.— 128 с.

Вероятностный анализ отказов газопроводов и оборудования, влияющих на надежность функционирования многоступенчатых распределительных систем

Лёвина Евгения Владимировна, студент магистратуры

Научный руководитель: Стариков Альберт Николаевич, кандидат технических наук, доцент
Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых

В статье авторы анализируют причины отказов газопроводов и оборудования, которые влияют на надежную и безопасную эксплуатацию распределительных систем.

Ключевые слова: сети газораспределения, сквозное повреждение, утечка газа, внезапный отказ, участок газопровода.

Сети газораспределения состоят из таких элементов и узлов, как:

- газопроводы высокого, среднего и низкого давления;
- узлы отключающей арматуры (задвижки, краны с компенсаторами, устанавливаемые в колодцах);
- гидравлические затворы и сборники конденсата;
- сложные пересечения естественных и искусственных преград, komponуемые в качестве отдельных узлов системы, включают газопроводы, отключающие арматуры, контрольные трубки и др;
- газорегуляторные станции и газорегуляторные пункты [1].

Трубопроводы со всем оборудованием, находящиеся на территории потребителя, называют сетью газопотребления и не относятся к сетям газораспределения.

Отмеченные основные элементы и узлы распределительных систем характеризуются различными значениями показателей надежности, например различными значениями параметра потока отказов $\omega \frac{1}{год}$ или $\frac{1}{ч}$.

Изолирующие покрытия не всегда являются гарантией защиты от коррозии из-за возможности совпадения места повреждения изоляции с зонами коррозионно-активных грунтов. Сквозные повреждения трубопровода являются результатом комплекса случайных событий, поэтому относится к редким и случайным.

При нахождении сквозного повреждения малого размера его локализируют, не снижая давление. Восстановление производится в течение 2–4 дней, при этом выбирается время снижения давления с учетом минимального потребления.

Серьезное сквозное отверстие (больше 20 мм) приводит к тому, что участок газопровода немедленно отключают участка, что являются внезапным отказом.

Разрыв сварного шва является еще одним серьезным повреждением, которое происходит при совпадении пониженных сопротивлений шва, может вызываться дефектом сварки и увеличенными нагрузками на газопровод, это может происходить из-за некачественной проварки при монтаже [1].

Практическая независимость параметра потока отказов распределительных трубопроводов ω от их диаметра имеет большое значение при выборе структурного резерва кольцевых сетей. Надежность в данном случае определяется схемой сети и не зависит от диаметров участков.

Разрушение сварного соединения возникает кольцевыми трещинами разной длины, оно образуется по оси шва. В ходе ис-

следования было выяснено, что примерно 15 процентов от всех разрушений сварных швов происходит с полным разрывом стыков. Свищи и сквозные поры наблюдаются крайне редко.

При утечке газа из сварного шва аварийная служба устраняет вырезкой стыка и вваркой катушки или навариванием муфты. При маленькой трещине аварийная служба может временно устранить накладкой бандажа.

Открытый разрыв стыков или 50 процентов стыка приводит к внезапному отказу системы [1].

Существует два вида повреждения отключающей арматуры. В первом случае происходит нарушение плотности перекрытия при помощи задвижки. При этом необходимо отключить абонентов, чтобы произвести ремонт. Во втором случае повреждения газопровода приводит к утечке через арматуру, что очень опасно из-за большой вероятности загазованности зданий и сооружений, граничащих с таким участком газопровода.

К утечкам газа приводят повреждения арматуры: нарушение герметичности сальников, разъемных соединений и трещины в корпусах.

Неправильный выбор набивки сальника без правильного учета особенности работы и отсутствия своевременной подтяжки приводит к нарушению его плотности.

Герметичность разъемного соединения, состоящего из прокладки, фланцев и болтов, обеспечивается спрессованностью прокладки при накручивании болтов.

Разъемные соединения со временем утрачивают плотность из-за колебаний температур. Утечка газа при этом устраняется только при помощи замены прокладки с понижением давления на близлежащем участке [1,2].

Линзовый компенсатор выходит из строя из-за коррозии, нарушения герметичности во фланцевых соединениях, а также при разрушении сварных швов, которые соединяют части линз. Устраняются со снижением давления.

Конденсаторосборники повреждаются в месте стояков и кранов. Стойки часто подвергаются коррозионным повреждениям, это объясняется низким качеством изолирующего покрытия. При ремонте стояков давление снижать не нужно.

Каждое мелкое повреждение ликвидируется без снижения давления в газопроводе, а, следовательно, и не отражается на потребителях. Но при отключении элемента от системы происходит отказ этого элемента. Отказ элемента на нерезервированной системе приводит к отказу всей системы. При резервированной сети такой отказ может и не способствовать отказу системы [3].

Можно сделать вывод, что повреждение элементов сети подразделяются на группы:

1) повреждение, при котором происходит отказ элемента и это требует отключение сети для ремонтных работ (трещины и разрывы в сварных швах; повреждения, вызванные коррозией труб (более 5 мм); трещины в корпусах задвижек и кранах; фланцевые повреждения; разрывы на сварных швах и повреждения линзовых компенсаторов и корпусов конденсатосборников; механическое повреждение трубопроводов и оборудования);

2) мелкое повреждение, устранимые без снижения давления газа и отключения участка (несквозные коррозионные повреждения; сквозные повреждения до 5 мм; свищи в сварных швах; утечки сальниковых уплотнений задвижек и кранов; утечки из кранов трубок конденсатосборников).

Существует два вида отказа: внезапные и постепенные. К внезапным относятся повреждения требуют немедленно отключить участок газопровода с целью его ремонта [4]. Таблица 1 отражает долю внезапных повреждений на газопроводах и оборудовании.

Таблица 1. Классификация повреждений и отказов элементов сетей газораспределения на среднем и высоком давлении

Вид повреждения и поврежденный элемент	Отказы элемента и%		Повреждения, не приводящие к отказу
	1-я гр — внезапный отказ	2-я гр — постепенный отказ	
Повреждения газопроводов			
Коррозия газопроводов	Сквозное повреждение (20 мм и более), сквозное повреждение, расположенное вблизи зданий 10...13%	Сквозное повреждение менее 20 мм 45...55%	Каверны, сквозные повреждения менее 5 мм 30...40%
Трещины в сварных стыках и их разрывы	Полный разрыв или более половины окружности. Разрыв стыка вблизи зданий 15...20%	Трещины в сварных стыках, неполный разрыв стыка 85...75%	Коррозионные свищи в сварном шве 5...10%
Механические повреждения, носящие случайный характер	50%	50%	¾
Повреждения оборудования газопроводов			
Задвижки (чугунные)	Разрывы и трещины в корпусе, отрывы фланцев, пробой прокладок выпадение клиньев 15%	Трещины в клиньях, неполное перекрытие 15%	Утечка в сальниковом уплотнителе 70%
Пробковые краны (типа КС)	Отрыв фланца, необходимость замены прокладки 1...1,5%	¾	Утечка через нажимной болт и сальник 98,5...99%
Линзовые компенсаторы	Разрыв сварного стыка 15%	Разрыв сварного стыка 85%	¾
Конденсатосборники	Разрывы сварных швов горшка 3...4%	Разрывы сварных швов горшка и трубки, поломки сальникового крана 4...5%	Коррозия трубки: утечка через сальник крана, во фланцах и резьбовых соединениях головок 91...93%

Чтобы выявить значения показателей надежности был выполнен анализ повреждений сетей газораспределения в условиях города. С помощью анализа получены значения параметров внезапных отказов, являющиеся основой для расчета надежности систем, 1/(км×год): коррозионные повреждения, повреждения сварных швов, механические повреждения.

Так как причины коррозионных повреждений, разрывов сварных стыков, механические повреждения, проблемы

с линзовыми компенсаторами и конденсатосборниками независимы друг от друга, расчетные значения параметров потока отказов равно сумме ω для всех вышеперечисленных видов [1].

С каждым годом показатель надежности повышается благодаря новым технологиям и материалам, используемым для реконструкции и строительства новых сетей.

Литература:

1. Горелов, С.А Сооружение и реконструкция распределительных систем газоснабжения: учеб. пособие РГУ нефти и газа им. И. М. Губкина / Горелов С. А., Горяинов Ю. А., Васильев Г. Г., Чугунов Л. С., Земенков Ю. Д.— М.: Недра, 2002.
2. 2. Яковлев, Е. И. Газовые сети и газохранилища: учебник для вузов, 2-е изд., перераб.идоп. / Е. И. Яковлев — М.: Недра, 1991.
3. 3. Багдасаров, В. А. Обслуживание и ремонт городских газопроводов. / В. А. Багдасаров Л.: Недра, 1985.

4. Шурайц, А. Л. Газопроводы из полимерных материалов: Пособие по проектированию, строительству и эксплуатации / А. Л. Шурайц, В. Ю. Каргин, Ю. Н. Вольнов — Саратов: Журнал «Волга — XXI век», 2007.

Пути разработки системы управления для беспилотного транспортного вертолета

Малясов Александр Александрович, студент магистратуры

Высшая школа системного инжиниринга Московского физико-технического института (г. Долгопрудный)

Научный руководитель: Пальченко Никита Вячеславович, главный конструктор по научно-техническому заделу

АО «Национальный центр вертолетостроения имени М. Л. Миля и Н. И. Камова» (г. Москва)

Изложена концепция реализации системы управления для беспилотного транспортного вертолета. Определены требования к системе необходимые для создания аппарата указанного класса, а также оценка снижения стоимости создания ЛА

Ключевые слова: беспилотный вертолет (БВ), опционально-пилотируемый вертолет (ОПВ), бортовой комплекс управления (БКУ).

Сегодня в большинстве случаев становится понятно, что беспилотная техника со временем потеснит «классические» пилотируемые летательные аппараты, начиная от вертолетов заканчивая самолетами.

Высокотехнологичная область находится на первом этапе создания целого семейства от аппаратов до пилотажных комплексов различной глубины автоматизации систем. Среди разработчиков существуют два основных подхода по проектированию беспилотных систем: первый идет по направлению развития беспилотников в чистом виде, где человеческий фактор полностью исключен из управления летающей техникой; второй идет по пути сохранения живого пилотирования, но обеспечив максимальное упрощение управление летательным аппаратом. С последующим развитием технологий, достигнув приемлемого уровня готовности и безопасности, оба пути соединяться в едином направлении, которое позволит опционально выбирать разные уровни контроля, что в настоящее время уже реализуется во многих перспективных проектах.

Примером второго пути развития беспилотных систем являются проекты компаний Sikorsky и Boeing. Проекты заключаются в модернизации серийных вертолетов с применением современных автономных технологий. Примером технологии служит демонстратор, построенный на базе вертолета модели S-92B и УН-60. Ключевым моментом реализации технологии заключается в обеспечении сокращения эксплуатационных расходов и повышении конкурентоспособности по сравнению с другими винтокрылыми аппаратами. Новые вычислительные системы беспилотных технологий позволят повысить уровень автономности, что обеспечит снижение требований к пилотированию и повысит востребованность воздушного судна на коммерческом рынке.

На сегодняшний день сегмент вертолетов гражданского назначения имеет устойчивую тенденцию развития, что обеспечивает за счет наличия у них конструктивных особенностей с перечнем уникальных полетных характеристик, которые являются незаменимыми во многих областях хозяйственной деятельности. По данным International Bureau of Aviation (Рисунок

1) наблюдается устойчивая мировая тенденция роста флота вертолетов гражданского назначения [1].

В тоже время в вертолетной отрасли наблюдается дефицит пилотов, где средний возраст пилота вертолета в Российской Федерации составляет 53 года [2]. Проблема подготовки пилотов связана с тенденцией по уменьшению количества АУЦ (Рисунок 2) [3]. В настоящее время подготовкой пилотов вертолетов гражданского назначения занимаются только два АУЦ (Омский летно-технический колледж и ЗАО «Русские вертолетные системы»), что обеспечивает около 180 пилотов в год. Ежегодно из профессии уходит около 200 пилотов вертолетов, что говорит о будущих проблемах обеспечения рынка авиаперевозок в Российской Федерации [2].

Решение проблемы нехватки авиационных кадров и удовлетворения устойчивой потребности авиаперевозок возможно за счёт применения беспилотных ЛА. Но в настоящее время существует большое количество ограничений связанных с применением беспилотных ЛА в различных отраслях, а именно:

— Ограничение применения при выполнении работ требующих точного пилотирования или пилотирования в ограниченном пространстве (городская застройка или наличие плотного авиационного трафика);

— Ограничение по перевозке людей;

— Ограничения полетов над населенными пунктами.

Третье ограничение связано с проблемой отсутствия нормативной базы и процесса сертификации беспилотных ЛА. Второе ограничение связано с обеспечением достаточного уровня надежности и безопасности эксплуатации, что может быть отражено путем применения ОПВ [4].

Первое ограничение может быть решено путем автоматизации маневров ЛА с применением и отработкой на ОПВ комплекса управления.

Путь построения от пилотируемого вертолета в полностью беспилотный вертолет возможно посредством модернизации серийного пилотируемого вертолета в два этапа. На первом этапе, используются имеющиеся серийные изделия и оборудование, с минимальной переделкой для обеспечения создания

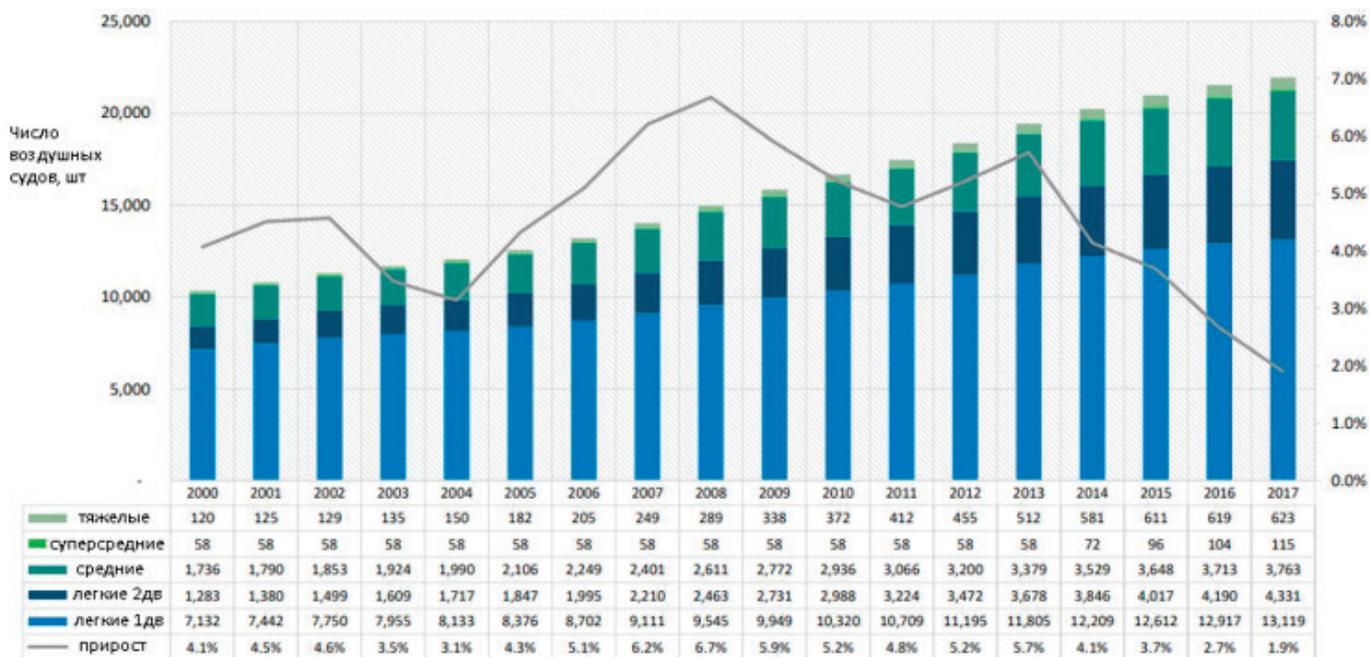


Рис. 1. Статистика по динамике мирового флота вертолетов гражданского назначения

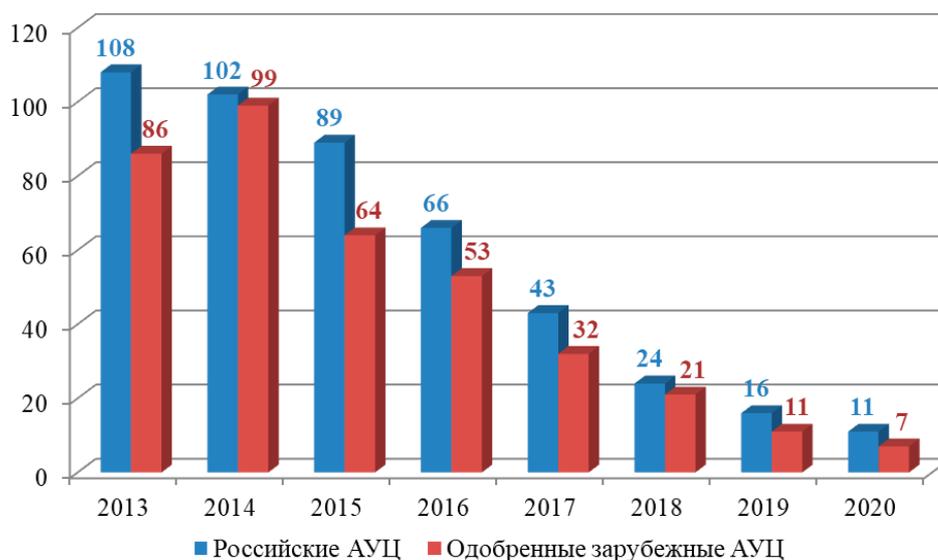


Рис. 2. Динамика количества АУЦ в Российской Федерации

варианта вертолета, предназначенного для полетов, как с летчиком, так и в беспилотном режиме — опционально-пилотируемый вариант. Это позволит отработать беспилотный вариант вертолета, проверить правильность выполнения логики работы под контролем летчика-испытателя. На втором этапе, при установке необходимого обновленного оборудования, создать вариант полностью беспилотного варианта вертолета и подтвердить полученные при отработке первого этапа результаты.

Для реализации первого этапа вертолет должен оснащаться бортовым комплексом управления (БКУ), обеспечивающим возможность выполнения вертолетом всех беспилотных

режимов, а также сохраняет возможность пилотирования летчиком как минимум по правилам визуального полета. Управление беспилотным полетом осуществляется как непосредственно с терминала на борту вертолета, так и по радиоканалу с наземного пункта управления.

На втором этапе создается полностью беспилотный вертолет путем модернизации программно-аппаратного обеспечения. Управление и контроль беспилотного вертолета осуществляется также с наземного пункта управления (НПУ). Для облегчения конструкции и увеличения длительности полета с вертолета демонтируются все более не нужные органы управ-

ления и средства индикации летчика, элементы механической проводки управления.

При выборе состава и структуры БКУ должны обеспечиваться следующие основные требования:

Функциональные:

- обеспечение достаточной точности навигационного решения (величина порядка единиц метров) для осуществления требуемой точности управления (величина по скорости до единицы м/с и по углам до градуса) при выполнении маловысотного полета, в том числе на режимах взлет/посадка, с обеспечением обновления информации не менее 50 Гц [5];

- БКУ должен иметь открытую архитектуру, как в части внутренней структуры, так и в части интерфейса с целевым оборудованием, а также возможность осуществления коррекционных действий с удаленного пункта управления (НПУ или пилотируемый летательный аппарат (ЛА) в случае отсутствия связи с НПУ);

- обеспечение достаточной скорости принятия решений при реализации автоматического или автоматизированного полета;

- обеспечение алгоритмов выполнения маневров в соответствии с правилами полета по приборам и обеспечения требования норм ИКАО и ФАП, в том числе внештатных;

Безопасности и надежности:

- обеспечение требований по показателям надежности работы комплекса;

- обеспечение отказоустойчивости комплекса, то есть сохранение его работоспособности без ухудшения технических характеристик при возникновении одного отказа или нескольких разнородных отказов аппаратуры из состава БКУ;

- способность комплекса к аппаратно-алгоритмической реконфигурации при обнаружении и распознавании отказа аппаратуры для обеспечения его работоспособности;

- обеспечение защиты информации КРЛ от несанкционированного доступа и управления;

- реализация процедур опознавания и идентификации ЛА;

Эксплуатационные:

- обеспечение оперативного (в полете) изменения полетного задания по информации, полученной от пункта управления по командной радиолинии;

- обеспечение снижения уровня оперативного контроля состояния и работоспособности систем ЛА, с сохранением достаточного уровня информирования оператора для принятия решений;

- наличие развитых средств обеспечения контролепригодности комплекса на всех этапах его применения, в том числе при реализации предполетного контроля, для обнаружения и распознавания (идентификации) отказа аппаратуры.

Возможны два варианта исполнения БКУ: с использованием электродистанционной системой управления (ЭДСУ) на ЛА или на базе имеющейся механической проводки управления со штатным вычислителем управления [6].

Оснащение БКУ ЭДСУ вместо механической проводки позволит:

- уменьшить массу вертолета за счет исключения механической части проводки управления;

- повысить управляемость вертолета до I уровня (по стандарту ADS-33) в различных условиях его эксплуатации;

- парировать выход параметров полета за эксплуатационные ограничения;

- уменьшить психофизиологическую нагрузку на экипаж с повышением его ситуационной осведомленности.

Альтернативный вариант создания БКУ выполняется с использованием штатной механической СУ с бортовым вычислителем управления. Переоборудование в опционально-пилотируемый ЛА осуществляется посредством параллельного подключения комплекса управления к штатной проводке управления аппаратом, что позволяет:

- сократить экономические и технические затраты на доработку вертолета;

- сократить объем летных испытаний ЛА в пилотируемом режиме, поскольку используются отработанные органы управления;

- упростить поэтапное проведение испытаний отдельных режимов автоматического режима управления;

- обеспечить возможность поэтапного наращивания программно-аппаратных средств БКУ, что дает возможность вести параллельно как летные испытания БКУ и вертолета в целом, так и собственно разработку (наращивание) программно-аппаратных средств БКУ.

Помимо системы управления маневрированием вертолет оснащен основными и вспомогательными системами, управление которыми необходимо осуществлять как вручную летчиком при выполнении пилотируемого полета, так и в автоматическом режиме, при беспилотном полете. Внедрение бортовой интегрированной системы контроля и управления позволит выполнять анализ состояния общевертолетных систем и аппаратно-алгоритмическую реконфигурацию в случае возникновения отказов, а также осуществлять проверку работоспособности комплекса пред/после полетной подготовки.

При реализации предложенной концепции модернизации пилотируемого вертолета в опционального-пилотируемый вертолет позволит обеспечить сокращение следующего объема работ:

- уменьшение объема разработок 60% систем на этапах эскизно-технического проекта (Этап 1) и выпуска РКД (Этап 2) (объем разработок остается в части модернизации ПНК, системы управления ЛА и общевертолетными системами);

- объем доработок по результатам испытаний (Этап 3) останется на том же уровне затрат;

- уменьшение объема доработки опытного производства (Этап 4) на 90%;

Работы на данном этапе касаются доработки линии сборки ЛА в части, линии монтирования и проверки оборудования.

- уменьшение стоимости изготовления опытных вертолетов, как для наземных (Этап 5), так и для летных испытаний (Этап 6) на 60% (ввиду возможности доработки серийных вертолетов);

- уменьшение объема стендовых испытаний (Этап 7) на 60% (ввиду проведения повторного цикла испытаний только для отработки оборудования и агрегатов системы управления);

- увеличение летных и сертификационных полетов (Этап 8) в 1,5 раза с 350 до 500 полетов.

Увеличение число полетов связано с отсутствием в настоящий момент сертификационного базиса для ОПВ.

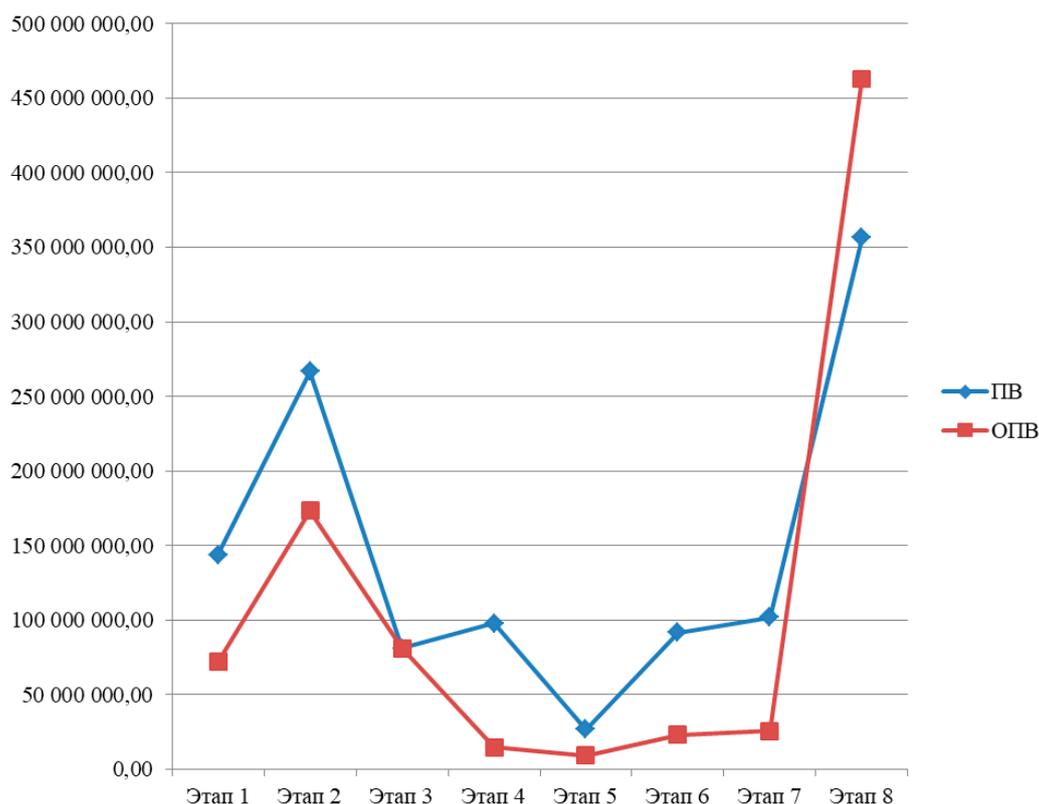


Рис. 3. Структура стоимости разработки ОПВ и ПВ

На рисунке 3 представлен график сравнения стоимости разработки ОПВ и пилотируемого вертолета (ПВ) по этапам разработки изделия.

Реализация концепции создания системы опционально-пилотируемого вертолета путем модернизации пилотируемого вертолета позволяет в течение двух-трех лет создать ОПВ с по-

следующей модернизацией в БЛА, отвечающий всем современным требованиям к ЛА по необходимому функционалу, надежности и отказоустойчивости, обеспечит решение проблемы роста авиаперевозок при существующей нехватке авиационных кадров, а также отработки нормативной базы и процесса сертификации беспилотных ЛА.

Литература:

1. Интернет ресурс <https://russia.trud.com>
2. Интернет ресурс <http://www.ato.ru>
3. Интернет портал Росавиации <https://favt.gov.ru>
4. Bento M. Unmanned aerial vehicles: an overview // Inside GNSS.— 2008.— № 1.
5. Forecast International Expects UAV Market to Rise Strongly through the Next Decade
6. UAVglobal. Unmanned systems and manufacturers: информационный портал, <http://www.uavglobal.com>

Опыт применения технологии бурения с контролем давления совместно с телесистемой с гидравлическим каналом связи при бурении горизонтальных скважин на Куюмбинском нефтегазовом месторождении

Невыпрегайло Игорь Николаевич, студент магистратуры
Тюменский индустриальный университет

Куюмбинское НГМ характеризуется сложным геологическим разрезом, в котором бурение сопровождается катастрофическими поглощениями в условиях высокой трещиноватости породы и как следствие потерями больших объёмов промывочной жидкости. В статье рассматривается опыт бурения горизонтальных участков скважины с применением технологии «бурение с контролем

давления» БКД, совместно с телесистемой с гидравлическим каналом связи. Рассматриваются возникшие при этом проблемы и ряд технологических решений для повышения эффективности бурения в осложненных условиях.

Ключевые слова: поглощение, осложнение, БКД, ГНВП, MWD, РУО, наращивание, промывочная жидкость, телесистема, промывка, азотирование, сигнал.

На Куюмбинском месторождении при бурении горизонтального участка скважины наряду с применением бурения на растворах на углеводородной основе применяется технология бурения с контролем давления (БКД).

В этой технологии в качестве промывочной жидкости применяется сырая нефть с нагнетанием азота в бурильную колонну, для исключения или снижения потерь бурового раствора, сохранения продуктивности пласта и вскрытия более протяжённых горизонтальных секций.

При бурении используется компоновка с забойным двигателем. Для корректировки профиля скважины применяется режим направленного бурения без вращения бурильной колонны, и роторное бурение за счет вращения всей бурильной колонны верхним силовым приводом. Таким образом бурение осуществляется комбинированным способом

Сопровождение бурения для контроля параметров проектного профиля и записи гамма-каротажа в реальном времени осуществлялось с помощью телесистемы с гидравлическим каналом связи MWD.

Нормальным условием для работы телесистемы с гидравлическим каналом связи, является принцип однородности и не сжимаемости промывочной жидкости в трубном и затрубном пространстве. Поскольку это принцип нарушался при азотировании скважины, сигнал от телесистемы отсутствовал и в режиме роторного и направленного бурения.

Для перезаписи гамма-каротажа и бурения в направленном режиме для корректировки профиля скважина переводилась на нефть. Причем требовались длительные ожидания для восстановления гидравлического потока и сигнала от телесистемы с постоянным расхаживанием бурильной колонны.

Бурение в направленном режиме всегда сопровождалось поглощением (на разных скважинах разные объемы поглощения), неустойчивым сигналом от телесистемы и низкой достоверностью передаваемых данных из-за неполного выхода пачек азотированной нефти.

Проблема снятия статичного замера от телесистемы заключалась в том, что, если в затрубном пространстве оставались азотные пачки, происходил переток промывочной жидкости через телесистему при отключенных буровых насосах. Наличие перетока воспринимался телесистемой как режим промывки, статический замер в таком случае отсутствовал, приходилось проводить повторные замеры.

Компромиссный вариант для поддержания скважины на балансе давлений и наличия сигнала от телесистемы для записи

каротажа в процессе бурения роторным способом, достигался путем снижения объема закачивания азота, но несмотря на это, со временем скважина теряла баланс давлений, приходилось останавливать бурение и ждать восстановления скважины на приток за счет закачки больших объемов азота.

Помимо этого, наблюдались частичная разгрузка веса бурильной колонны при прохождении муфты бурильного замка через роторный устьевого герметизатор при направленном бурении. Для исключения этого эффекта бурилось 3 м ротором для прохождения муфты б.к. ниже этого герметизатор.

Протяжённость горизонтальных участков по проекту строительства скважины составляла в среднем от 700 до 1100 м. Основная проблема при бурении после 600 метров горизонтального участка, заключалась в доведении нагрузки на долото. Зависания и срывы компоновки на забой, затяжки при отрыве от забоя, скачки давления, отключение насосов, повторные ориентирования приводили к существенным потерям времени и ресурсов.

По мере углубления наступал такой момент, когда отсутствие веса на крюке при бурении в направленном режиме не позволяло продолжать бурение и производить корректировку профиля, а профилактические мероприятия в виде прокачки вязкоупругих составов, и подъемы до башмака обсадной колонны с обратной проработкой не улучшало ситуацию. В таких случаях бурение прекращалось по технологическим причинам с не до буренными горизонтальными участками.

Вся совокупность факторов бурения в таких осложнённых условиях показала не совместимость двух технологий: БКД и гидравлический канал связи, и ситуация требовала другого подхода. Такой подход был найден.

Он заключался в использовании телеметрической системы с электромагнитным каналом связи. Электромагнитный канал связи использует электромагнитные волны, образующиеся между изолированным участком колонны бурильных труб и протекающие по горной породе, и менее зависим от качества промывочной жидкости.

Проблема доведения нагрузки до забоя решилась включением в КНБК осциллятора, устройства для создания малоамплитудных осевых колебаний в буровой колонне способствующих снижению сил трения, что способствует и улучшению передачи нагрузки на долото

В дальнейшем опыт бурения показал эффективность такого подхода. Бурение скважины выполнялось до проектной глубины с сокращением временных затрат.

Литература:

1. Харитонов А. А., Квеско Н. Г. «Бурение геологоразведочных и эксплуатационных скважин на Куюмбинском Л. У». Международный научно-исследовательский журнал № 5–699–101. 2018.
2. А. И. Булатов, Е. Ю. Проселков, Ю. М. Проселков. Бурение горизонтальных скважин / — Краснодар: Совет. Кубань, 2008. — 424 с:

Причины пожаров на предприятиях угледобывающей промышленности

Радченко Владимир Евгеньевич, студент магистратуры

Научный руководитель: Акименко Наталья Юрьевна, кандидат технических наук, доцент

Тихоокеанский государственный университет (г. Хабаровск)

В статье рассмотрены вопросы обеспечения пожарной безопасности на предприятиях угледобывающей промышленности, проведен анализ причин возникновения аварий, а также дан ряд рекомендаций, способствующих повышению уровня пожарной безопасности и недопущения травматизма и летальных исходов.

Ключевые слова: пожарная безопасность, угледобывающая промышленность.

Угольная промышленность [1] — одна из ведущих отраслей ТЭК. Уголь используют как технологическое сырье (в виде кокса) в черной металлургии и химической промышленности (коксовые газы) для производства минеральных удобрений и пластмасс, а также уголь используют как энергетическое сырье для производства электроэнергии на ТЭС, для отопления жилищ. Общие геологические запасы угля в России оцениваются свыше 4 трлн тонн. В России сосредоточено 12% мировых запасов угля. Средний объем добычи угля в России в год составляет 475,0 млн тонн.

Для того, чтобы рядовой уголь имел большее содержание полезного компонента, строятся горные предприятия для первичной переработки твердых полезных ископаемых с целью получения технически ценных продуктов, пригодных для промышленного использования.

С точки зрения экономики Российской Федерации угольная промышленность является сегодня одной из системообразующих, при этом полностью является рыночной отраслью, ни одного государственного предприятия в отрасли нет. Угольные предприятия являются градообразующими для 31 моногорода общей численностью 1,5 млн человек. В угольной отрасли трудится 148 тыс. человек, плюс 500 тыс. рабочих мест в смежных отраслях.

Но если мы посмотрим с другой стороны, то видно, что угледобывающая промышленность является взрывопожароопасной и несет в себе большую угрозу для жизни тысяч человек. Причины пожара могут быть разные: взрыв угольной пыли или метано-воздушной смеси, бездействие или ошибка в действиях в нештатной ситуации; несоблюдение правил пожарной безопасности (техники безопасности); нарушение (повреждение), отключение систем взрывозащищенности оборудования, систем автоматики и безопасности электрооборудования; эксплуатация оборудования при параметрах, выходящих за пределы технических условий; проведение постоянных или временных огневых работ без специального разрешения.

Ежегодно в угледобывающей отрасли происходят пожары, причиняющие вред здоровью человека. Начиная с 2010 года на предприятиях угольной промышленности в следствии аварий произошло несколько крупных пожаров [2]:

1) 9 мая 2010 года в г. Междуреченске Кемеровской области на угледобывающем предприятии (шахта) «Распадская», в результате двух взрывов метано-воздушной смеси с интервалом в 4 часа, произошел пожар, унёсший жизни 91 человека, еще 229 человек получили травмы различной степени тяжести.

2) 20 января 2013 года в г. Кеселевске Кемеровской области на шахте № 7 ОАО «СУЭК-Кузбасс» по причине обру-

шения породы бортовой выработки произошел пожар, что привело к взрыву взрывов метано-воздушной смеси. В результате аварии погибло 4 человека.

3) 11 февраля 2013 года в г. Воркуте Республики Коми на угледобывающем предприятии «Воркутинская» в результате нарушения взрывозащищенности оборудования произошел взрыв угольной пыли и пожар, унесший жизни 19 человек, еще 3 работников получили травмы различной степени тяжести.

4) 25 февраля 2016 года в г. Воркуте Республики Коми на угледобывающем предприятии (шахта) «Северная» в результате взрыва метано-воздушной смеси произошло обрушение породы и возник пожар, унесший жизни 36 человек, 5 человек получили травмы различной степени тяжести.

5) 23 июня 2017 года в п. Южный Республики Коми на угледобывающем предприятии (шахта) «Интинская» в результате обрушения металлических конструкций части кровли произошел пожар, в результате которого погиб 1 человек.

6) 10 мая 2018 года в г. Междуреченске Кемеровской области на угледобывающем предприятии (шахта) «Распадская» в результате возгорания угля в лаве вспыхнул пожар. Из шахты были эвакуированы 195 человек, пострадавших нет.

7) 15 августа 2020 года в п. Воргашор Республики Коми на угледобывающем предприятии (шахта) «Воргашорская» в результате обрушения галереи (перехода), соединяющего здания погрузки и сортировки погибли 4 человека.

8) 14 сентября 2020 года в г. Белово Кемеровской области на угледобывающем предприятии (шахта) «Чертинская-Коксовая» в результате обрушения кровли в штреке произошел пожар, в результате которого 1 человек погиб, 1 пропал без вести.

По приведенным выше данным видно, что из-за недостаточного количества воздуха для создания безопасной концентрации, основными причинами возникновения пожаров являются взрыв метано-воздушной смеси и возгорание угля, что приводит к разным тяжелым последствиям, не редко человеческим жертвам и наносит большой материальный ущерб.

Для предотвращения данных чрезвычайных ситуаций в настоящее время принимаются следующие мероприятия:

1. Совершенствование методики прогнозирования локальных участков пластов угля с высокой природной метаносимостью. Существующие методы определения природной метаносимости и интенсивности выделения метана не учитывают неравномерность природной газоносности в пласте, которая изменяется в широких пределах, как по простиранию, так и по слоям пласта.

2. Разработка методических рекомендаций расчёта расхода воздуха, критерием эффективности которого должно быть активное контролируемое снижение концентрации метано-воздушной смеси у поверхности забоя выработки.

3. Увеличение расходов воздуха по всем выработкам шахты.

Также если обратить особое внимание на основные технологически процессы, применяемые на угледобывающих пред-

приятиях, а это использование пожароопасных материалов, образование угольной пыли во время процесса обработки угля, буровзрывные и очистные работы, концевые операции в лаве и т.д., можно сделать вывод, что контрольно-надзорная деятельность, проводимая для соблюдения правил пожарной безопасности, должна проводиться с учетом технологических процессов, проводимых на угледобывающих предприятиях.

Литература:

1. А. В. Лунев. Угольная промышленность России в мировой энергообеспечении, 2006. — 276 с.
2. Шахтерская энциклопедия. Аварии на шахтах России. URL: <http://miningwiki.ru/wiki>, обращение к источнику 08.12.2020.
3. Старостено А., Михайлов Л. и др. Безопасность в ЧС техногенного характера. Взрывы и промышленная безопасность// ОБЖ. — 2004. — № 5. — с. 35–40.
4. ГОСТ Р 22.0.08–96. Безопасность в чрезвычайных ситуациях. Техногенные чрезвычайные ситуации. Взрывы.

Концепция BIM-технологии при проектировании автомобильных дорог

Углова Евгения Владимировна, доктор технических наук, профессор;
 Николенко Максим Александрович, кандидат технических наук, доцент;
 Чирва Дмитрий Владимирович, кандидат технических наук, доцент;
 Саркисян Давид Артурович, студент магистратуры;
 Керимов Закир Алимйраевич, студент магистратуры;
 Оганян Юрий Кароевич; студент магистратуры
 Донской государственный технический университет, (г. Ростов-на-Дону)

В статье авторами проводится анализ технологии информационного моделирования и оценка эффективности использования.

Ключевые слова: автомобильные дороги, BIM-технологии, информационные модели, информационное моделирование, жизненный цикл, среда общих данных, жизненный цикл, ЦММ.

BIM (англ. Building Information Model или Modeling) — это метод проектирования, в ходе использования которого применяются все параметры, связанные с жизненным циклом объекта строительства, начиная от финансовых затрат, необходимых на само строительство, заканчивая последующими ежемесячными расходами на электроэнергию.

Все эти данные формируют так называемую информационную модель, в которой изменение одного параметра приводит к автоматическому перерасчету всех остальных. В этом принципиальное отличие BIM от обычной 3D-визуализации. Одним из главных плюсов является то, что работать с единой информационной моделью могут одновременно несколько групп специалистов.

BIM позволяет выполнять большое количество функций, необходимых для реализации объекта, основными из которых являются:

1. Создание чертежей;
2. Разработка отчетов;
3. Построение (моделирование) графиков выполнения работ;
4. Выполнение полного анализа проекта;
5. Управление состоянием объекта на стадии эксплуатации;
6. Визуализация объектов.

Именно поэтому, у проектировщиков и строителей имеются безграничные возможности для принятия наилучшего решения, опираясь при этом на все имеющиеся данные.

Концепция BIM представляет собой особый подход к жизненному циклу объекта, при котором информационная модель содержит в себе конструкторскую, технологическую, экономическую и другую информацию о составляющих его взаимосвязанных элементах.

Жизненный цикл автомобильной дороги — это промежуток времени, за который объект проходит все процессы своего существования, начиная от стадии концептуального планирования, заканчивая ликвидацией (демонтажом).

Его делят на крупные стадии, которые в свою очередь подразделяются на отдельные этапы и процессы. Основными этапами жизненного цикла автомобильной дороги являются:

1. Планирование. На данном этапе производится разработка несколько различных вариантов проложения трассы, их технико-экономическое сравнение и дальнейший выбор оптимального варианта.

2. Проектирование. Крайне трудоемкий процесс, в ходе которого осуществляют: геометрическое моделирование местности; создание детальной геометрической модели автомобильной дороги, пересечений, развязок, искусственных сооружений и т.д.

3. Строительство, ремонт или реконструкция. Это так называемая этап реализации проекта, в ходе которого производятся строительные работы и подготовка объекта к сдаче в эксплуатацию.

4. Эксплуатация. На протяжении данного этапа автомобильная дорога эксплуатируется и регулярно обслуживается, в связи с чем проводится: периодическая диагностика; процесс выявления дефектов; планирование мероприятий по текущему и капитальному ремонту как самой автомобильной дороги, так и расположенных на ней сооружений.

Для определения эффективности BIM-технологий рассмотрим весь процесс проектирования линейного объекта посредством программного комплекса компании Autodesk. Для понимания стоит охарактеризовать основной функционал используемых программных продуктов:

Autodesk InfraWorks 360 — программа, предназначенная для концептуального проектирования, быстрого 3D-моделирования существующей инфраструктуры на основе данных различных САПР и ГИС, растровых материалов, а также данных из открытых источников, оперативной разработки, анализа и визуализации нескольких вариантов проектов инфраструктурных объектов (дорог, мостов, инженерных коммуникаций, площадных объектов, участков застройки и т.д.)

AutoCAD Civil 3D — программа предназначенная для проектирования объектов инфраструктуры, позволяет оптимизировать выполнение задач, поддерживать согласованность данных и быстро реагировать на изменения.

Главным плюсом, и причиной использования, AutoCAD Civil 3D является то, что данный продукт позволяет создавать любые требуемые элементы поперечных сечений дорожной одежды, достаточно быстро анализирует проектные решения, достигая оптимума, за короткий промежуток времени.

Autodesk Navisworks — программа предназначенная для комплексного управления BIM-проектами, объединения моделей, поиска пересечений, работы с 4D (время/деньги) и 5D (время + деньги) информацией. Данная программа применяется для проверки моделей на наличие ошибок на стадии проектирования, мониторинга строительства, демонстрации объектов и многих других задачах.

Стоит отметить, что для создания информационной модели возможно использование различных программных комплексов, а не только продуктов Autodesk.

Весь процесс состоит из 4-х основных этапов:

I. Общее представление проектируемого линейного объекта в Autodesk InfraWorks 360.

Используя данные открытых источников и функционал Autodesk InfraWorks 360 создается модель, содержащая информацию о застройке территории, названиях улиц и объектов, рельефе, гидрографии. Проектируется план, продольный и поперечный профиль трассы, после чего данную модель экспортируют в AutoCAD Civil 3D.

II. Детальная проработка модели в AutoCAD Civil 3D.

На основе созданной модели и с помощью функционала AutoCAD Civil 3D производится: изменение конструкции поперечного сечения, создание поверхности по верху участка дороги на основе коридоров, моделирование и увязка пересечений и примыканий, устройство искусственных сооружений.

По итогу работы с AutoCAD Civil 3D создается трехмерная динамическая модель, все элементы которой логически взаимосвязаны.

III. Экспорт данных в Autodesk InfraWorks 360 с дальнейшей визуализацией.

После проработки необходимо экспортировать уже отректированную модель трассы из AutoCAD Civil 3D в Autodesk InfraWorks 360. Для лучшей наглядности полученного проекта настраиваются текстуры и подгружаются элементы обустройства дороги. На данном этапе проект можно считать завершённым.

IV. Итоговый сбор модели в Autodesk Navisworks.

Готовый проект, со всеми прилагающимися файлами, подгружают в Autodesk Navisworks. При использовании функционала данной программы получают: отчет о стоимости и объемах используемых строительных материалов, почасовой график использования механизмов, календарный график строительства. Autodesk Navisworks, путем визуализации, позволяет продемонстрировать каждый этап, при этом учитывая введенные строки строительства как самого трассы, так и других элементов (путепроводы, трубы и т.д.).

Главным плюсом BIM технологии является в первую очередь то, что создаются взаимосвязанные участки автомобильной дороги и развязки. Так, меняя общий профиль трассы, было видно, как это отражается на развязках, на объемах работ и т.д. А без своевременного и автоматического внесения результатов корректировки проекта в общую трёхмерную модель невозможно было бы выполнить в более короткий срок без использования BIM технологии.

Литература:

1. ОДМ 218.3.105–2018 — «Методические рекомендации по организации взаимодействия участников разработки проектной и рабочей документации на пилотных проектах строительства, капитального ремонта и реконструкции автомобильных дорог с применением BIM-технологии»;
2. ГОСТ Р 57311–2016 — «Моделирование информационное в строительстве. Требования к эксплуатационной документации объектов завершённого строительства»;
3. СП 301.1325800.2017 — «Информационное моделирование в строительстве. Правила организации работ производственными техническими отделами»;
4. Райкова Л. С., Анисимов С. С., Петренко Д. А. 3D-визуализация как современная технология повышения качества проектных решений // САПР и ГИС автомобильных дорог. 2014 № 1(2). С. 20–24.
5. Сарычев Д. С., Скворцов А. В. Элементы моделей автомобильных дорог и уровни проработки как основа требований к информационным моделям // САПР и ГИС автомобильных дорог. 2015 № 1(4). С. 30–36. DOI: 10.17273/CADGIS.2015.1.4.
6. Open Geospatial Consortium // Официальный сайт международного консорциума Open Geospatial Consortium. URL: <http://www.opengeospatial.org/> (дата обращения: 10.12.2020).

Молодой ученый

Международный научный журнал
№ 50 (340) / 2020

Выпускающий редактор Г. А. Кайнова
Ответственные редакторы Е. И. Осянина, О. А. Шульга, З. А. Огурцова
Художник Е. А. Шишков
Подготовка оригинал-макета П. Я. Бурьянов, М. В. Голубцов, О. В. Майер

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.

Журнал размещается и индексируется на портале eLIBRARY.RU, на момент выхода номера в свет журнал не входит в РИНЦ.

Свидетельство о регистрации СМИ ПИ № ФС77-38059 от 11 ноября 2009 г., выдано Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзор).

ISSN-L 2072-0297

ISSN 2077-8295 (Online)

Учредитель и издатель: ООО «Издательство Молодой ученый»

Номер подписан в печать 23.12.2020. Дата выхода в свет: 30.12.2020.

Формат 60×90/8. Тираж 500 экз. Цена свободная.

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; <https://moluch.ru/>

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