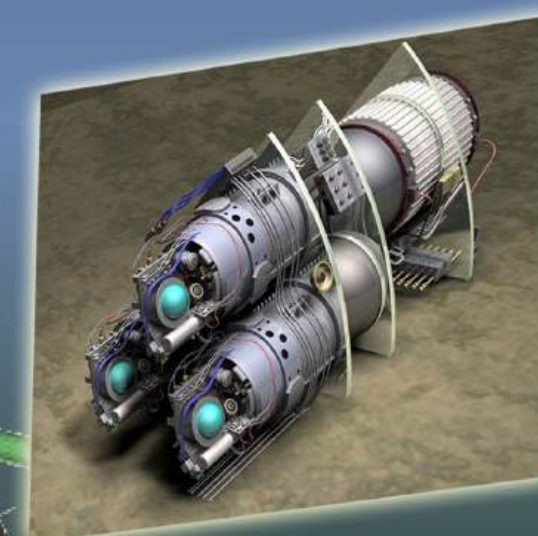
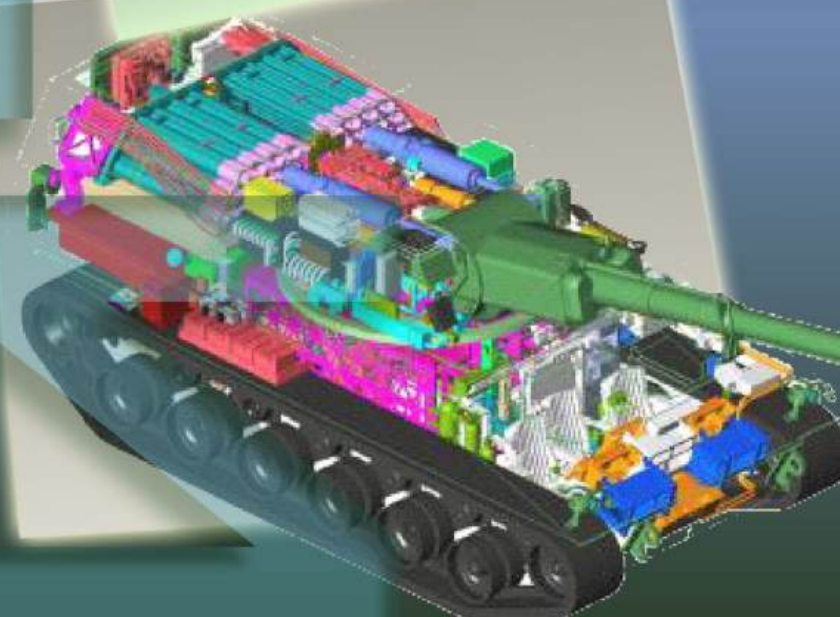
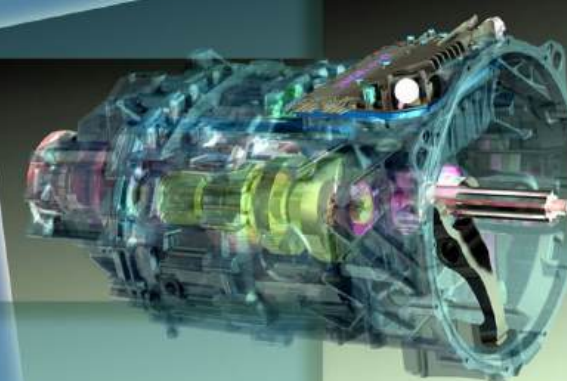
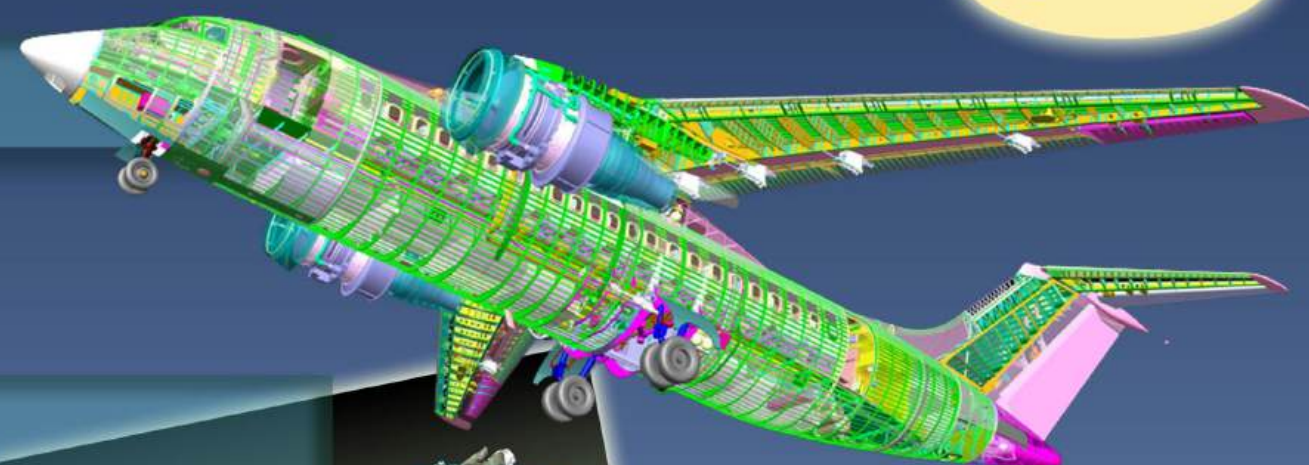


Индекс 79378

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

в проектировании и производстве

№ 1' 2023





Все новейшие достижения и современные разработки предприятий оборонного комплекса — в научно-технических журналах ФГУП «НТЦ оборонного комплекса «Компас»

ФГУП «Научно-технический центр оборонного комплекса «Компас» является издателем следующих научных журналов:



Межотраслевой научно-технический журнал
Оборонный комплекс — научно-техническому прогрессу России
(4 выпуска)
Подписной индекс **79379**
Издается с 1984 года



Межотраслевой научно-технический журнал
Конструкции из композиционных материалов
(4 выпуска)
Подписной индекс **80089**
Издается с 1981 года



Научно-технический журнал
Информационные технологии в проектировании и производстве
(4 выпуска)
Подписной индекс **79378**
Издается с 1976 года



Межотраслевой научно-практический журнал
Экология промышленного производства
(4 выпуска)
Подписной индекс **80090**
Издается с 1993 года



Научно-практический журнал
Вопросы защиты информации
(4 выпуска)
Подписной индекс **79187**
Издается с 1974 года

Все издания ФГУП «Научно-технический центр оборонного комплекса «Компас»:

✓ включены решением ВАК Министерства науки и высшего образования России в перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные результаты диссертаций на соискание ученой степени кандидата и доктора наук;

✓ метаданные выпусков включены в базу данных Российского индекса научного цитирования (РИНЦ).

Более подробную информацию об изданиях, подписке, дополнительных услугах можно получить по тел.: 8 (495) 491-43-17, 8 (495) 491-77-67, 8 (495) 491-77-20 (подписка);
факс: 8 (495) 491-44-80.
E-mail: izdanie@ntckompas.ru

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ПРОЕКТИРОВАНИИ И ПРОИЗВОДСТВЕ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ
УНИТАРНОЕ ПРЕДПРИЯТИЕ
«НАУЧНО-ТЕХНИЧЕСКИЙ ЦЕНТР
ОБОРОННОГО КОМПЛЕКСА
«КОМПАС»

№ 1 (189) 2023

НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Основан в 1976 г.

Главный редактор

И. А. Шеремет, акад. РАН, проф., заместитель
директора Российского фонда фундаменталь-
ных исследований по научной работе

Заместитель главного редактора

В. Н. Грдин, д-р техн. наук, проф.,
научный руководитель ЦИТИ РАН

Ответственный секретарь

Э. Ю. Есипова, канд. техн. наук,
ФГУП «НТЦ оборонного комплекса «Компас»

Редакционная коллегия:

В. В. Бордоже, д-р техн. наук, проф., Координа-
ционный совет по информационным тех-
нологиям предприятий ОПК РФ; А. И. Громов,
канд. хим. наук, НИУ «Высшая школа экономи-
ки»; В. В. Кондратьев, чл.-кор. РАН, д-р техн.
наук, проф., Нижегородский государственный
технический университет им. Р. А. Алексеева;
Ю. Н. Кофанов, д-р техн. наук, проф., МИЭМ
НИУ ВШЭ; С. А. Матвеев, канд. техн. наук,
ФГУП «ЦНИИмаш»; Е. И. Митрушкин, д-р техн.
наук, проф., АО «НИИАА»; С. В. Назаров, д-р
техн. наук, проф., ЗАО «Московский научно-
исследовательский телевизионный институт»;
М. А. Пирогова, канд. техн. наук, НИУ МЭИ;
С. В. Попов, д-р техн. наук, Холдинг «Швабе»;
А. Ф. Ронжин, д-р физ.-мат. наук, проф., Вычис-
лительный центр им. А. А. Дородницына;
П. О. Скобелев, д-р техн. наук, НАО «Группа
компаний «Генезис знаний», СамГТУ; В. В. Со-
ломин, канд. воен. наук, ФГУП «НТЦ оборонного
комплекса «Компас»; А. П. Солнцев, д-р. воен.
наук, проф., ФГУП «НТЦ оборонного комплекса
«Компас»; Е. В. Судов, д-р техн. наук, проф.,
НИЦ SALS-технологий «Прикладная логистика»;
А. В. Цыркун, д-р техн. наук, проф., ГКНПЦ
им. М. В. Хруничева; В. А. Шахнов, чл.-кор.
РАН, д-р техн. наук, проф., МГТУ им. Н. Э. Бау-
мана; Е. И. Шульгин, д-р техн. наук, НИЦ при
президиуме АИН им. А. М. Прохорова

☎ редакции (495) 491-43-17
E-mail: izdanie@ntckompas.ru
http://ntckompas.ru

Информационные технологии
в проектировании и производстве:
Науч.-техн. журн./ФГУП «НТЦ оборонного ком-
плекса «Компас», 2022.

№ 1 (189). С. 1—56.

Компьютерная верстка: Н. В. Ильина,
К. В. Трыкина

Подписано в печать 18.03.2023.

Формат 60×84 1/8. Бумага офсетная.
Усл. печ. л. 6,5. Уч.-изд. л. 6,7.

Заказ 2013. Тираж 250 экз.

Адрес редакции: 125424, Москва,
Волоколамское ш., 77, ФГУП «НТЦ оборон-
ного комплекса «Компас».
Отпечатано: 101000, Москва,
Лубянский проезд, д. 15, стр. 4, помещ. IX,
ком. 15, 16

ООО «Спиди-Принт.ру»
Индекс 79378. 7 статей.

Статьи рецензируются.

© Федеральное государственное унитарное предприя-
тие «Научно-технический центр оборонного комплек-
са «Компас», 2023

СОДЕРЖАНИЕ

Инновационные технологии в решении комплексных задач управления и проектирования, моделирования и производства

Пирогова М. А., Лешихина И. Е., Краюшкин В. А. Применение
технологий дополненной реальности и искусственного интел-
лекта для автоматизации процессов визуального контроля каче-
ства выполнения сборочных операций 3

Правильщиков П. А. Квантовый подход к решению задач
формальной верификации цифровых схем 10

Поляков Е. Ю., Кондусов Д. В., Кондусова В. Б. Способ повы-
шения эффективности автоматизированных систем технологиче-
ской подготовки производства 21

Информационные технологии в решении задач управления

Логинов И. В. Подход к расширению возможностей монито-
ринга функционального состояния автоматизированных систем в
условиях непрерывных изменений 26

Современные методы и средства решения задач проектирования и моделирования.

Объектно-ориентированное программирование

Соболь В. М. Автоматизированная станция проверки испол-
нения абонентского протокола специализированной коммуни-
кационной сети 34

Хвостов А. А., Ткач В. В., Журавлев А. А., Семенухин О. А.,
Татаренков Е. А. Проектирование оконных стеклопакетов на
основе математической модели тепломассопереноса с учетом
климатической зоны эксплуатации 42

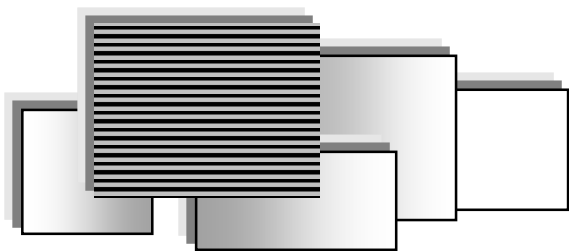
Прикладные вопросы и задачи применения информационных технологий

Бутко А. О., Каверная И. Н. Разработка системы управления
техническими данными на основе комплекса структурно-
параметрического моделирования 49

INFORMATION TECHNOLOGY OF CAD/CAM/CAE

THE SCIENTIFIC JOURNAL

Information on recent advances in CAD/CAM/CAE, industry & business & complex information technology, including methods and tools of computer graphing, geometrical and mathematical modeling, database control methods on estimation and provision of control and reliability.



ИННОВАЦИОННЫЕ ТЕХНОЛОГИИ РЕШЕНИЯ ЗАДАЧ ПРОЕКТИРОВАНИЯ, МОДЕЛИРОВАНИЯ И ПРОИЗВОДСТВА

УДК 338.364

DOI: 10.52190/2073-2597_2023_1_3

EDN: SKENUQ

ПРИМЕНЕНИЕ ТЕХНОЛОГИЙ ДОПОЛНЕННОЙ РЕАЛЬНОСТИ И ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ АВТОМАТИЗАЦИИ ПРОЦЕССОВ ВИЗУАЛЬНОГО КОНТРОЛЯ КАЧЕСТВА ВЫПОЛНЕНИЯ СБОРОЧНЫХ ОПЕРАЦИЙ

М. А. Пирогова, канд. техн. наук; И. Е. Лешихина, канд. техн. наук
ФГБОУ ВО НИУ «Московский энергетический институт», Москва, Россия

В. А. Краюшкин, канд. техн. наук
ООО «Продуктивные технологические системы», Москва, Россия

Рассмотрены возможности использования технологий дополненной реальности и искусственного интеллекта для повышения производительности и эффективности выполнения операций по контролю качества сборочных операций. Проанализированы особенности обеих технологий и специфика их практического применения.

Ключевые слова: дополненная реальность, искусственный интеллект, машинное зрение, цифровая трансформация производства.

Дорожные карты цифровой трансформации производства для отраслей дискретной промышленности предусматривают внедрение единых унифицированных технологических подходов на всех этапах жизненного цикла изделия и переходах между ними. В связи с этим все большую актуальность приобретает применение передовых информационных технологий "Индустрии 4.0" на "стыковых" операциях поточного производства, в том числе и для инспекции качества выполненных операций. Если в ходе проектно-конструкторских работ для такой инспекции возможно использование виртуального анализа, то непосредственно в процессе производства, на постах изготовления комплектующих единиц, узловой сборки изделий, монтажа агрегатов, проверка качества выполненных операций во многом зависит от опыта и умений сотрудников-инспекторов. Для повышения эффективности работы инспекторов посты контроля качества необходимо оснастить автоматизированными средствами, максимально использующими технологии "Индустрии 4.0".

Любая инспекция качества начинается с визуального осмотра изделия и выявления явных

его несоответствий ожидаемому результату. Для этого предлагается использовать технические решения, основанные на автоматизации процесса визуализации изображений, например технологии компьютерного зрения (машинного зрения), которые давно и прочно обосновались в промышленности среди технологий Интернета вещей [1]. Компьютерное зрение предлагается применять для оцифровки и трансляции изображения инспектируемого изделия в программное приложение, которое сопоставляет получаемый цифровой контент с заранее определенной цифровой моделью "идеального" изделия. Приложение, в котором можно получить результат такого сопоставления, должно работать в связке с программным обеспечением (ПО) для систем компьютерного зрения и передавать результат на устройство связи с инспектором. Результат представляется в виде цифровых значений параметров изделия, в форме звуковых сигналов, в виде цветовых сигналов или их комбинаций. Далее инспектор анализирует полученную информацию и принимает решение о том, является ли инспектируемый им образец качественным изделием, то есть наиболее полно соответству-

ющим "идеальному" образцу, или браком. В такой постановке задачи цифровой трансформации процессов визуального контроля качества выпускаемых изделий для ее решения требуется применение технологий, которые умеют работать с цифровым контентом "машинного зрения". Как следует из исследований авторов, к таким технологиям можно в первую очередь отнести метод одновременной локализации и построения карты *SLAM* (*simultaneous localization and mapping*) [2, 3] и тензорную классификацию на базе нейронных сетей глубокого обучения *DNN* (*deep neural networks*). Поскольку обе эти технологии не являются точными измерительными средствами, с помощью их применения автоматизируется только самый первый опыт, проверка "на первый взгляд", что позволяет оценить, похож ли рассматриваемый объект на то, что должно получиться в результате предыдущей операции (и он, следовательно, может быть продвинут далее по этапам жизненного цикла), или объект похож на дефектное изделие (то есть это брак исполнения предыдущей операции, и его нельзя передавать на следующий этап производства). Автоматизация первичной визуальной оценки качества изделия по его внешнему виду позволяет снизить количество неверно принимаемых инспектором решений, которые происходят или могут происходить при ускорении темпа работы, снижении внимания, повышении утомляемости, появлении эффекта замыливания глаз при длинных рабочих сменах и/или большом разнообразии ассортимента контролируемых изделий.

Применение технологии дополненной реальности для задачи визуального контроля

В рамках освоения систем и платформ дополненной реальности (ДР) в течение последних пяти лет авторы данной работы неоднократно получали запросы от проектных и производственных компаний из отраслей машиностроения и приборостроения на выполнение работ по автоматизации распознавания и трекинга изделий на основе их 3D-моделей. При этом как для 3D-мозаичных поверхностных моделей (3D-сканов образцового изделия) [2, 3], так и для параметрических твердотельных 3D-моделей сборок изделий в САПР [4] для распознавания и трекинга применяли метод *SLAM*, а также *VISLAM* (визуально-инерционный *SLAM*). Целью выполненных работ являлась не инспекция качества изделий, а только освоение применимости методов таргетирования в *SLAM*-технологии. Тем не менее, в ходе работ стало совершенно ясно, что для промышленного использования в рамках цифровой трансформации производства реально достижимое практическое распознавание (*SLAM*-таргетинг) по его 3D-представлению возможно лишь для 3D-моделей сборок изделий в САПР (САПР-моделей).

Именно эти модели можно считать образцом, применение которого в рамках "визуализационных" процессов ДР могло бы служить квалификатором качества исполнения реального объекта.

Существующие платформы ДР, как локальные, так и входящие в состав комплексов промышленного Интернета вещей, позволяют использовать результаты рабочего проектирования и структуру электронного состава изделия для этапов его жизненного цикла как образец для выполнения первичного визуального контроля. 3D-мозаичные поверхностные модели, особенно выполненные сканированием реальных физических образцов, не обладают в отличие от САПР-моделей достаточными характеристиками точности представления формы объекта и, следовательно, не могут быть достоверным источником качественного определения правильности исполнения реального изделия. Именно по причине более низкой точности 3D-сканов образцового изделия по сравнению с САПР-моделью для применения технологии дополненной реальности в процессе визуального контроля правильности сборки выбирается всегда такая платформа ДР, в которой САПР-представление для таргетирования первично.

Из многочисленных регулярно появляющихся в открытом доступе материалов по платформам ДР (см., например, [5, 6]), использующим для разработки приложений дополненной реальности напрямую 3D-данные в форматах промышленных САПР, т. е. без трансляции в тесселированные форматы игровых движков, можно в настоящее время выделить платформы *Vuforia* (включая *RE'FLEKT*) и *TeamViewer* (*Ubimax*), в которых САПР-модели используются для голограмм элементов контента ДР. Для *SLAM*-таргетирования 3D-модели в САПР используют только в *Vuforia Studio* и только в режиме объектного распознавания *Model Targeting*. САПР-модели для разработки приложений дополненной реальности в *TeamViewer xInspect* используют, в лучшем случае, для информационного сопровождения пошаговых инструкций операций рабочего места, технику же *SLAM* в *TeamViewer xInspect* для таргетирования не используют, а применяют таргетирование на основе меток.

К началу третьего десятилетия XXI в. всего лишь две платформы *Vuforia* — *Vuforia Engine* и *Vuforia Studio* могли работать с технологией *SLAM* напрямую. При этом *Vuforia Engine* в случае применения *SLAM* не использует форматы САПР и работает только с поверхностными представлениями на основе облака точек, полученных в результате сканирования, или с тесселированными представлениями САПР-моделей. Платформа *Vuforia Studio* работает напрямую с САПР-моделями как в режиме моделирования сценариев дополненной реальности, так и при использовании *SLAM*-таргетирования в режиме *Model Targeting*. Применение платформы

Vuforia Studio для разработки приложений дополненной реальности с исполнением режима *Model Targeting* в процедурах визуального контроля показало, что срабатывание таргета, т. е. выполнение достоверного распознавания реального объекта по САПР-модели, имеет место только в том случае, когда отличительный признак "правильности" выполненной реальной сборки (реального изделия) присутствует во фронтальной проекции подготовленного таргета.

Примеры *SLAM*-таргетирования для определения того, "видим" ли мы реально перед устройством ДР то изделие, или нет, в случае разработки приложений с использованием *Vuforia Studio* показали, что данного функционала распознавания по САПР-моделям вполне достаточно для выполнения поставленной выше задачи. При этом функциональность самой платформы, которая не ограничивается только таргетингом типа *Model Targeting*, а гораздо шире, оказалась избыточной для поставленной задачи визуальной инспекции качества. На самом деле, распознавание (в нашем случае *Model Targeting*) — это только начало, стартовый шаг в разработке приложений дополненной реальности в *Vuforia Studio*. Остальные функции, изначально "встроенные" в прототип, такие как использование многочисленных 3D- и 2D-виджетов для организации сцены ДР, возможности настройки и ветвления сценариев, анимация 3D-моделей в сценах и пр., не были задействованы в готовом приложении.

В середине 2021 г. вышел предварительный релиз новой разработки в линейке продуктов платформы *Vuforia* — *Vuforia Instruct*, которая предназначена только для задач инспекции готового изделия. *Vuforia Instruct* использует САПР-модель в качестве таргета. *Vuforia Instruct* — это приложение дополненной реальности, созданное в проприетарном редакторе *Vuforia Editor*, которое работает на устройстве пользователя (смартфон, планшет, очки дополненной реальности) только тогда, когда физический объект, например инспектируемая сборка, агрегат, узел или деталь, "совпадает" с 3D-таргетом. В качестве таргета в *Vuforia Instruct* может быть применена САПР-модель, подготовленная только в форматах промышленных систем *CATIA*, *NX*, *Creo* или нейтральных *STEP* и *Iges*. При разработке приложения в пространственной зоне *Vuforia Editor* размещаемая в этой зоне 3D-модель позиционируется таким образом, чтобы квалификационный объект, наличие или отсутствие правильного вида которого указывало бы на событие "распознан" или "не распознан", был расположен в наилучшем представлении с учетом перспективы и угла обзора. В *Vuforia Instruct* такой *SLAM*-таргет называется *Product Status*. Для удобства и с учетом особенностей процедур визуальной инспекции, их ограниченности по ДР-функционалу, на одно приложение *Vuforia Instruct* выделяется работа только с одним *Product Status*. Пример работы в редакторе *Vuforia Editor* при разработке приложения *Vuforia Instruct* представлен на рис. 1.

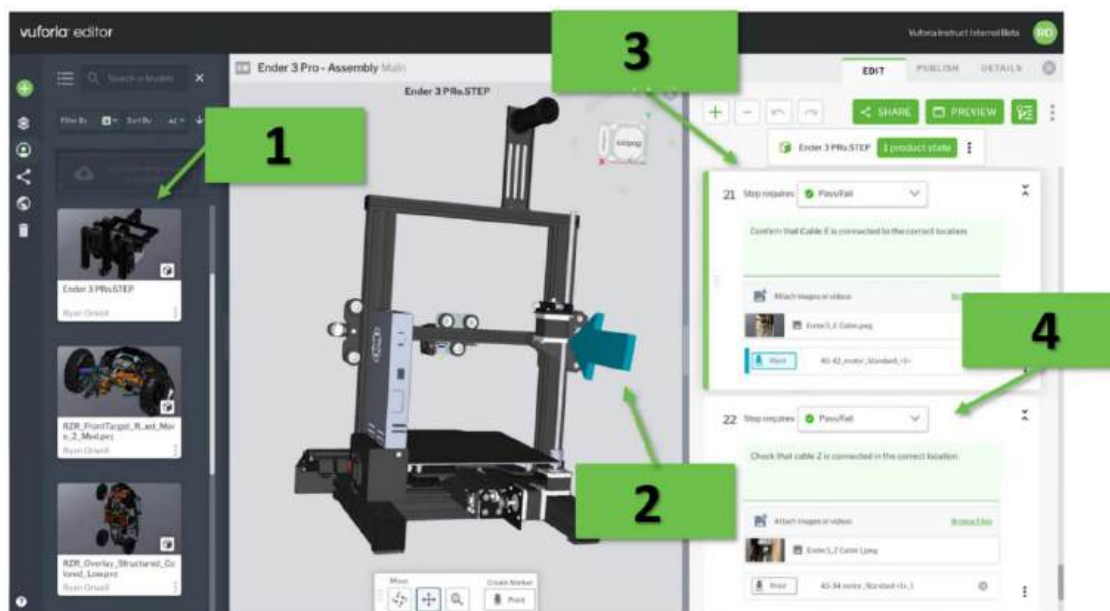


Рис. 1. Разработка в редакторе *Vuforia Editor* инструкции по визуальной инспекции изделия для применения в *Vuforia Instruct*:

- 1 — импорт САПР-данных; 2 — расстановка указателей элементов конструкции, на которые надо обратить внимание;
- 3 — описание действий по инспекции представленного на рассмотрение изделия ("памятка инспектора");
- 4 — интерактивная метка об исполнении инспектором соответствующего шага инспекции

Пример физического устройства, полученного для визуальной инспекции в приложения *Vuforia Instruct*, представлен на рис. 2. Инспектор в режиме реального времени на очередном шаге проверяет наличие конструктивного элемента — направляющей укладчика филамента — в составе инспектируемого изделия (белый контур на переднем плане — *SLAM*-таргет искомого узла).



Рис. 2. Пример работы приложения *Vuforia Instruct*

По мнению авторов данной работы, *Vuforia Instruct* — это наиболее сбалансированное и простое в освоении средство разработки приложений дополненной реальности для автоматизации промышленных процессов визуальной инспекции качества сборочных работ. Само приложение размещено на сервере в сети, аналогично приложениям дополненной реальности, выполненным в *Vuforia Studio*. Среда разработки *Vuforia Editor* также представляет собой облачное решение, доступное через *Web*-интерфейс. *Vuforia Editor* и репозиторий приложений для *Vuforia Instruct* — это "облачные" серверы с локализацией за пределами РФ, доступ к которым оформляется в виде подписки. Реализация работы с данными серверами осуществляется на основе облачной *SAAS*-структуры *Atlas*. На практике в условиях отечественного применения это означает, что с марта 2022 г. подписка на сервисы серверов *Vuforia Editor* и *Vuforia Instruct* недоступна [5, 6].

Применение технологии искусственного интеллекта для задачи визуального контроля

Визуальный контроль качества выполнения, например сборочных работ, сводится к тому, что необходимо классифицировать цифровой видеоконтент и определить похож или нет тот класс, который определен в результате обработки видеоконтента, на ожидаемый. Если рассматривать задачу визуальной инспекции входного видеопотока как задачу классификации получаемого контента, то можно принимать

решения с использованием технологии искусственного интеллекта. Авторами в работе [6] приведен пример освоения технологии искусственного интеллекта для решения промышленной задачи этапа сборки, когда оператор решал, правильный ли компонент он видит в зоне сборки. Вариант применения технологий работы с видеоконтентом от ООО "Мотив НТ", показанный в [6, рис. 3], подразумевал тензорный анализ видеозображения, распознавание и классификацию объекта, а также показ результата этого распознавания на экране. Очевидно, что была решена не только задача классификации, но и задача детектирования объектов на основе алгоритмов искусственного интеллекта.

В принципе, для задачи визуальной инспекции готового изделия — не для контроля пошаговой операции сборки, а только для финальной инспекции качества сборки — достаточно и классификации. Классификация видеоконтента в задачах машинного зрения выполняется в с использованием алгоритмов тензорного анализа и обучения нейронных сетей [7]. В отсутствие отечественного нейрочипа аппаратная составляющая для реализации такого рода алгоритмов основана на использовании видеопроцессоров и специального ПО для эффективного выполнения параллельных высоконагруженных вычислений. Лидерство в аппаратной реализации в течение нескольких последних лет уверенно удерживает компания *nVIDIA* с ее структурами центрального процессорного устройства (ЦПУ) и внедрением программных компонент *CUDA*. Отечественных аппаратных аналогов *nVIDIA* в настоящее время нет, но пока еще существуют реальные возможности получать работоспособные *nVIDIA* многоядерные ЦПУ-решения с полной поддержкой *CUDA* для отечественных разработок, в том числе и в области искусственного интеллекта.

В распоряжение авторов статьи был предоставлен модуль *nVIDIA Jetson Nano* — базовый компонент программно-аппаратного комплекса искусственного интеллекта для решения задачи визуальной инспекции качества сборки. Технические возможности работы с тяжело нагруженными вычислениями в *nVIDIA Jetson Nano* реализуются за счет многоядерных ЦПУ и графического процессора и оцениваются в 0.47 *TFLOPs*, что уже достаточно для выполнения работ по обработке видеопотока в реальном времени с параллельным распознаванием и классификацией. Для проверки возможностей использования алгоритмов компьютерного зрения на аппаратной платформе *nVIDIA Jetson Nano* сборка была доработана путем установки двух цифровых видеокамер *Raspberry Pi V2-8MP* непосредственно на макетную плату *Jetson Nano* по интерфейсу *MIPI CSI-2* и дополнительно еще одной *Web*-камеры, подключаемой к *Jetson Nano* по *USB*-интерфейсу. Спарка камер *Pi V2-8MP* рассматривалась в качестве устройства непосредственного захвата стереоизображе-

ния с возможностью анализа глубины получаемого изображения. Web-камера, в отличие от камер *Pi V2-8MP*, рассматривали не в качестве смонтированного на корпусе *Jetson Nano*, а вынесенного за пределы модуля присоединенного устройства захвата изображения инспектируемой области рабочего места.

На *nVIDIA Jetson Nano* была установлена операционная система *UBUNTU 18.04*, в состав которой входит полнофункциональная базовая типовая реализация языка программирования *Python* версий 2.7 и 3.6. Для реализации программ компьютерного зрения, использования алгоритмов искусственного интеллекта и для работы с нейронными сетями аппаратно-программный комплекс был дополнен библиотеками тензорного анализа *TensorRT* и репозитарием пре-тренированных сетей *DNN*, утилитами для классификации, детектирования и сегментации с поддержкой специализированных библиотек обработки изображений *Caffe*, *TensorFlow UFF* и свертков *ONNX*. Весь комплект ПО, включая операционную систему, *DNN*-репозитарий и утилиты, не подпадает под санкционные ограничения и доступен отечественным пользователям для скачивания и установки. Возможные разработки на основе рассматриваемого программно-аппаратного комплекса могут быть доведены до уровня коммерческого использования.

В отличие от проектов, где сочетаются машинное зрение и дополненная реальность, когда вся трудоемкость сводится к правильному ис-

пользованию заранее подготовленных и включенных в базовый состав платформы модулей-виджетов, проекты на основе искусственного интеллекта существенно более трудоёмки в разработке и требуют более высокого профессионального уровня разработчиков в области объектно-ориентированного программирования. В случае применения искусственного интеллекта вместо 3D-моделей конструкторским подразделениям необходимо по каждому из предполагаемых классов распознаваемых объектов готовить три группы наборов датасетов — тренировочный, тестовый и валидационный, причем готовить эти датасеты по месту применения самостоятельно, то есть без привлечения САПР-специалистов. Кроме того, поскольку расчеты выполняются в автоматическом режиме, задача тренировки на своих датасетах типовых *DNN*-сетей требует значительно большего машинного времени, даже с учетом высокой производительности многоядерных структур процессорной части *nVIDIA Jetson Nano*. Следовательно, разработчикам придется заняться оптимизацией ПО с учетом ограничений задействованного аппаратного и системного обеспечения и реальных условий промышленного применения.

На рис. 3 представлен разработанный авторами прототип рабочего места инспектора качества на базе комплекта *nVIDIA Jetson Nano*, который использовали для проверки применимости технологии искусственного интеллекта в промышленных задачах визуального контроля качества сборки.

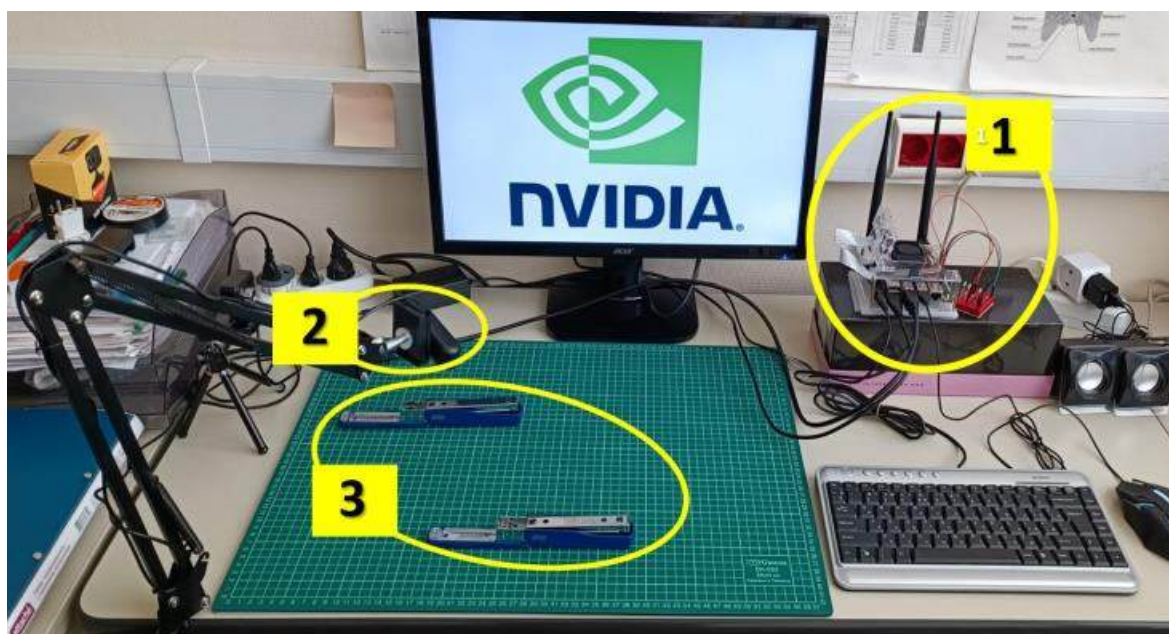


Рис. 3. Состав стенда на основе комплекта *nVIDIA Jetson Nano*

- 1 — *nVIDIA Jetson Nano* с двумя *Raspberry Pi V2-8MP* камерами, выходом на USB-аудиокарту и системой LED-индикации результатов инспекции, соединенной с *GPIO*-контактным полем;
- 2 — вынесенная Web-камера контроля образцов в поле визуальной инспекции;
- 3 — два образца некоторых механических устройств в поле визуальной инспекции: нижний — собранный без нарушений, верхний — бракованный

Основой процесса классификации на *DNN*-сети пользовательских датасетов стало базовое ПО — *Image Processing*. Программные доработки процедур *Image Processing* выполнялись с помощью модификации базового ПО *Tensor RT* (версия *Python 3.6*). Оригинальные дополнительные процедуры были разработаны в виде законченных программных фрагментов и блоков кода. Полученные решения были проинтегрированы в базовое ПО для решения задачи повышения достоверности в ходе распознавания с помощью компьютерного зрения и генерации сигналов для внешних процессов и процедур в ходе успешного выполнения классификации по сверточной сети с помощью искусственного интеллекта. Прототип рабочего места оператора визуальной инспекции качества выполнения сборочных операций в комплектации, представленной на рис. 3, проходил тестирование, результативность которого отмечена в ходе первичных оценочных демонстраций [8].

Заключение

Применение технологий дополненной реальности и искусственного интеллекта для визуальной инспекции результата выполнения сборочных операций на постах промышленного контроля качества позволяет ослабить влияние человеческого фактора и решать задачи автоматизации инспекции и контроля на уровне, достаточном для достижения полноценной цифровой трансформации промышленного дискретного производства. По мнению авторов, подкрепленному реальным опытом, решение задач визуальной постоперационной инспекции изделий, выполняемых с привлечением технологии *SLAM*-распознавания (дополненная реальность) или с использованием тренинга нейронных сетей глубокого обучения (искусственный интеллект), в настоящее время находящихся в стадии демонстрационных образцов, только начинает привлекать внимание индустрий. Пока что результаты, которые достигнуты и продемонстрированы с помощью этих технологий, еще только анализируют на предмет практической возможности внедрения и необходимости глубокой доработки под требования заказчиков. Однако выполненные авторами работы позволяют определить практическую ценность каждого из подходов, равно как и возможности их промышленной реализации в условиях существующих санкционных ограничений.

Применение технологии дополненной реальности для визуальной инспекции качества выполнения сборочных операций на постах промышленного контроля качества наиболее эффективно, просто и без программирования может быть реализовано с использованием платформы *Vuforia Instruct*. Процедуры визуальной инспекции в этом случае может разработать персонал, не сертифицированный в каче-

стве пользователей САПР и не имеющий квалификацию программистов. Для хранения разработок используются облачные сервисы, что отменяет расходы предприятия-заказчика на содержание и администрирование структуры хранения—доступа. Отмечается низкий порог входа в технологию, что привлекательно в случае промышленной реализации. Основным недостатком — санкции с марта 2022 г., полный запрет со стороны вендора на доступ к *Vuforia Editor-Vuforia Instruct* для предприятий и организаций, работающих на территории РФ и Белоруссии.

Применение технологии искусственного интеллекта для визуальной инспекции качества выполнения сборочных операций наиболее рационально может быть реализовано с использованием специализированных программно-аппаратных вычислительных комплексов *nVIDIA Jetson*, в частности *nVIDIA Jetson Nano*. Для применения технологии искусственного интеллекта требуется штат квалифицированных специалистов в области подготовки классификационных датасетов контролируемой продукции, программистов в области объектно-ориентированного ПО. Отмечаются существенные затраты машинного времени на стадии подготовки исходных данных и тренинга и ретренинга нейронных сетей. Однако в случае использования технологии искусственного интеллекта отсутствуют ограничения, связанные с применением промышленной платформы дополненной реальности — технические и программные компоненты для начала разработки приложений на территории РФ легально доступны. Таким образом, для отечественного пользователя при решении практических индустриальных задач, использующих "машинное зрение" и обработку полученного видеоконтента, использование технологии искусственного интеллекта предпочтительнее даже с учетом высокого порога входа в такую технологию.

Литература

1. Machine vision for Industry 4.0: application and case studies / edited by Roshani Raut, Salahddine Krit, Prasenjit Chatterjee // CRC Press — Taylor & Francis Group, LLC3. 2022.
2. SLAM (метод) [Электронный ресурс]. [https://ru.wikipedia.org/wiki/SLAM_\(%D0%BC%D0%B5%D1%82%D0%BE%D0%B4\)](https://ru.wikipedia.org/wiki/SLAM_(%D0%BC%D0%B5%D1%82%D0%BE%D0%B4)) (дата обращения: 14.02.2023).
3. Технология дополненной реальности / Лекция № 10. Промышленное применение ДР. Платформа от компании PTC Vuforia [Электронный ресурс]. http://a0601.narod.ru/AR_Lctn_10.pdf (дата обращения: 14.02.2023).
4. Лабораторная работа №3. Vuforia Studio: Создание приложения дополненной реальности (часть II из III) [Электронный ресурс]. http://a0601.narod.ru/LR03_ARforPot.pdf (дата обращения: 14.02.2023).
5. Краюшкин В. А., Лешихина И. Е., Пирогова М. А. Технология дополненной реальности как

эффективный инструмент применения пошаговых руководств и инструкций на рабочем месте // Информационные технологии в проектировании и производстве. 2022. №2. С. 23-28.

6. Краюшкин В. А., Лешихина И. Е., Пирогова М. А. Платформы дополненной реальности для промышленного применения: Россия 2022 // Информационные технологии в проектировании и производстве. 2022. № 4. С. 3—10.

7. Солем Ян Эрик. Программирование компьютерного зрения на языке Python. — М.: ДМК-пресс, 2016. — 312 с.

8. Проверка стенда-прототипа ПТС для выявления брака на конвейере. AI под Jetson Nano. Image processing [Электронный ресурс]. <https://www.youtube.com/watch?v=LxlaR19U2ro> (дата обращения: 14.02.2023).

APPLICATION OF AUGMENTED REALITY AND ARTIFICIAL INTELLIGENCE TECHNOLOGIES TO AUTOMATE THE VISUAL INSPECTION PROCESSES OF MANUFACTURING ASSEMBLY OPERATIONS

M. A. Pirogova, I. E. Leshikhina

National Research University "Moscow Power Engineering Institute", Moscow, Russia

V. A. Krayushkin

LLC "PTS", Moscow, Russia

The article discusses the possibilities of augmented reality (AR) and artificial intelligence (AI) technologies to improve the productivity and efficiency of performing operations of the visual inspections the quality of assembly operations. The features of both technologies and the specifics of practical application are analyzed.

Keywords: augmented reality, artificial intelligence, computer vision, digital transformation.

Пирогова Марина Аркадьевна, доцент.

E-mail: PirogovaMA@mpei.ru

Лешихина Ирина Евгеньевна, доцент.

E-mail: liy56@mail.ru

Краюшкин Владимир Анатольевич, руководитель проектов.

E-mail: vkray@pts-russia.com

Статья поступила в редакцию 15 февраля 2023 г.

УДК 004.421; 004.052.42

DOI: 10.52190/2073-2597_2023_1_10

EDN: SVNQNL

КВАНТОВЫЙ ПОДХОД К РЕШЕНИЮ ЗАДАЧ ФОРМАЛЬНОЙ ВЕРИФИКАЦИИ ЦИФРОВЫХ СХЕМ

П. А. Правильщиков, канд. техн. наук

ФГБУ "Институт проблем управления им. В. А. Трапезникова РАН", Москва, Россия

Описаны квантовые алгоритмы на примерах решения задачи верификации систем логического управления на основе жёсткой логики. Предполагается, что квантовые алгоритмы функционируют на платформе проблемно-ориентированного квантового компьютера с механизмом квантового параллелизма. Предложено решение задач формальной верификации на основе новой вычислительной модели квантового компьютера.

Ключевые слова: квантовые компьютеры, квантовые алгоритмы, формальная верификация, кубиты, кутриты, куквадриты, куниты.

В условиях роста кибератак наблюдается отказ разработчиков от программируемых логических контроллеров и возврат к использованию управляющих устройств на жёсткой логике [1—5], реализующей принцип построения систем логического управления на базе цифровых схем, таких, как комбинационные устройства (КУ), триггеры, счётчики, регистры. Аргументом в пользу жёсткой логики становится практически полная защищённость построенных на ее основе систем управления технологическими процессами от кибертерроризма, одним из примеров которого стало нападение на оборудование иранского завода по обогащению ядерного топлива в г. Нетанз [3]. Применение жёсткой логики практически полностью исключает возникновение как случайных ошибок программистов, так и злонамеренных аппаратных и программных "закладок", приводящих к сбоям и отказам.

С возвратом жёсткой логики вернулись и проблемы, связанные с формальной верификацией цифровых схем в целях поиска ошибок проектирования [6].

В случае программируемых логических контроллеров для верификации используют классические компьютеры и суперкомпьютеры. Чтобы решить задачу формальной верификации цифровых схем на жёсткой логике, необходимо разработать новые квантовые алгоритмы, функционирующие на платформе проблемно-ориентированного квантового компьютера (КК) с механизмом квантового параллелизма.

Первая работа по теме формальной верификации была представлена Дж. П. Ротом в 1977 г. [7]. Обзоры с описанием решения задач формальной верификации на платформе классических компьютеров можно найти в публикациях зарубежных авторов [8—10]. Для решения этих задач используется дорогостоящее оборудование — рабочие станции (их стоимость достигает 15000 тыс. долл. США), а также компактные настольные суперкомпьютеры (стоимостью примерно 150 тыс. долл. США). Иногда применяют

самые мощные суперкомпьютеры, которые стоят 100 млн долл. США и более. Это объясняется важностью задач верификации. Если ошибки обнаружены в программном обеспечении (ПО), то компания-разработчик выпустит исправленный вариант ПО для его достаточно простой замены, если же ошибка проектирования будет обнаружена в схемах приобретенного компьютера или управляющего устройства, то "мы навсегда остаёмся со своим горем" [9].

Формальная верификация — важный этап разработки, и ее решение сводится к решению булевых уравнений [6, с. 50, 77—79]. Ранее в работе [11] автор привёл краткий обзор существующих решений задач верификации. Там же приведён проект решения этих задач на платформе сопроцессора-ускорителя с механизмом классического гипермассового параллелизма, пригодного для использования параллельно-последовательных D -алгоритмов. В этом случае D -алгоритмы и механизм гипермассового параллелизма ускорителя являются, по сути, своеобразным вычислительным решето, отсеивающим на каждом шаге из множества потенциальных задач те, которые не удовлетворяют условиям решения*. Для этого в D -алгоритмах применяется операция "разность множеств", которая используется на каждом шаге D -алгоритма [12]. Дж. П. Рот использует последовательный D -алгоритм для формальной верификации [7]. Позднее был разработан параллельно-последовательный D -алгоритм для специализированного сопроцессора-ускорителя [11], позволяющий решать задачи верификации с полиномиальным временем $\Delta t_{\text{реш}}$, однако в некоторых случаях требуются экспоненциаль-

* В математике в качестве алгоритма поиска простых чисел используют решето Эратосфена. Известно также и решето числового поля, применяемое в качестве алгоритма факторизации целых чисел. Последовательные D -алгоритмы входят в класс алгоритмов с откатом. В параллельно-последовательных и квантовых D -алгоритмах откат отсутствует, так как он заменён расширением множества кандидатов на отсев на каждом шаге таких алгоритмов.

ные объёмы оперативной памяти (ОП) сопроцессора. Поэтому использование классических компьютеров и сопроцессоров для решения задач верификации приводит лишь к ограниченному успеху. Подчеркнём, что по мере развития КК объём памяти квантового регистра (объём $V_{\text{кр}}$) растёт экспоненциально. Именно поэтому проблемно-ориентированные КК — квантовые ускорители (КвУ) — предполагают новый подход к решению булевых уравнений в целях верификации.

В отличие от традиционных классических компьютеров, оперирующих битами, большинство КК оперирует квантовыми битами — кубитами. Кубит содержит 0 или 1, а в состоянии суперпозиции может одновременно содержать оба числа. Сегодня известны коммерчески доступные КК, а также прототипы новых КК, решающие определённые классы задач [13–15]. В связи с этим приведём последние достижения в международной квантово-компьютерной гонке за квантовым превосходством* [12, 16]. Так, выдающимся достижением является создание в Китае в 2020 г. проблемно-ориентированного КК *Jiuzhang*. Этот КК за 200 секунд решает сложную физическую задачу гауссовской выборки бозонов. По оценкам математиков, самому мощному суперкомпьютеру потребовалось бы для решения этой задачи около 2–2,5 млрд лет. Это экспериментально доказанный пример квантового превосходства. Больших успехов достигла и канадская компания *D-wave systems*, презентовавшая коммерчески доступный КК *Advantage*. Его регистр содержит 5000 кубитов ($L = 5000$) и может хранить 2^{5000} больших двоичных чисел. Его стоимость 15 млн долл. США. Заметим, что число 2^{5000} больше верхней оценки числа атомов во Вселенной 10^{81} . Этот КК уже приобрела для верификации программ и проектов аппаратуры компьютеров корпорация *Lockheed Martin*, входящая в военно-промышленный комплекс США. Такой же КК приобрела и компания *Google*, а также *NASA*, ЦРУ и ФБР.

В 2019 г. в корпорации *IBM* создан прототип КК с регистром в 53 запутанных кубита ($L = 53$), а также создан и коммерчески доступный КК *IBM Quantum System One* (его регистр содержит 27 запутанных кубитов). В 2021 г. *IBM* начала экспансию своих КК на зарубежные рынки, квантовые компьютеры *IBM Quantum System One* установлены в Германии, Японии и Южной Корее. К ним возможен облачный доступ. В ноябре 2021 г. *IBM* объявила, что ей удалось создать квантовый процессор *Eagle* на 127 кубитов. В 2022 г. компания уже выпустила процессор *Osprey* (регистр процессора

Osprey содержит 433 кубита). В 2023 г. *IBM* планирует выпустить процессор *Condor* с регистром, содержащим 1121 кубит [17]. Коммерчески доступен и КК американской компании *Honeywell* с 64 запутанными кубитами. Эти КК нуждаются в охлаждении почти до 0 °K. Однако компания *IonQ* в США представила процессор, который обещает простое производство и эксплуатацию КК без использования криогеники.

Особое внимание привлек к себе прототип квантового компьютера, разработанный группой ученых из *Perdue University* (США) [13]. Каждый разряд в регистре этого КК является кунитом (англ. эквивалент — *qudit*). Для каждого из кунитов количество различных чисел v , которые могут одновременно содержаться в одном куните, т. е. в одном разряде регистра КК, равно 32. Если число кунитов в регистре равно L , то ёмкость $V_{\text{кр}}$ равна $32^L = 2^{5L}$. Этот результат — начало нового направления в развитии КК, связанного не с увеличением числа L кубитов в регистре, а с увеличением числа v в каждом куните регистра. Увеличение числа L кунитов в регистре также приветствуется. Все приведённые КК являются проблемно-ориентированными в соответствии с теоремой о запрете на создание универсального программируемого КК [15]. Приведённые КК являются программируемыми, но не универсальными.

В создании квантовых компьютеров участвует и Россия. В 2015 г. на разработки КК был выделен 1 млрд руб. В октябре 2016 г. Президент РФ В. В. Путин распорядился выделить дополнительно около 1,5 млрд руб. на информационные технологии в части квантовых вычислений. В 2019 г. было решено предоставить Госкорпорации "Росатом" около 24 млрд руб. на создание прототипов КК. В 2022 г. стало известно, что объём финансирования разработки КК и квантовых информационных технологий составит 100 млрд рублей.

Постановка задачи

В случае формальной верификации цифровых схем, "наиболее простой и, как это обычно оказывается, весьма трудоёмкий способ решения булева уравнения с n переменными заключается в переборе всех 2^n наборов значений переменных и подстановке их в уравнение с целью выяснить, удовлетворяют ли они уравнению" [6]. Но полный перебор, трудоёмкий для классических компьютеров, можно быстро выполнить на КК. Это продемонстрировал квантовый алгоритм ** факторизации П. Шора [15]. Квантовый алгоритм, выполняющий полный перебор, представляет собой алгоритм, задающий последовательность унитарных операций, реализуемых квантовыми логическими элемен-

* Под квантовым превосходством понимается некоторый абстрактный порог в развитии, пройдя его КК будут способны решать очень сложные задачи, справиться с которыми за приемлемое время $\Delta t_{\text{пр}}$ недоступно ни одному из существующих суперкомпьютеров.

** Под квантовым алгоритмом понимается алгоритм, предназначенный для выполнения на КК.

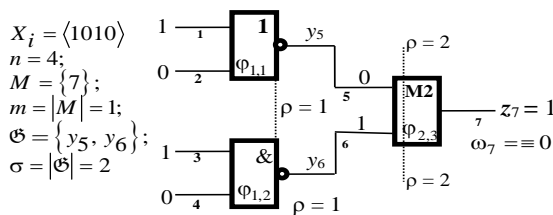
тами (квантовыми вентилями) с указанием, над какими именно кунитами их надо совершать*. Частными случаями кунитов являются кубиты ($v = 2$), кутриты ($v = 3$), куквадриты ($v = 4$). Часто под квантовым алгоритмом понимают некоторую квантовую схему, которая построена из квантовых вентилях, реализующих унитарные матрицы. Квантовые алгоритмы иногда меняют постановки известных задач. Это связано с практически неограниченными в потенциале ресурсами КК и такими их особенностями, как квантовое считывание из кунитов результатов решения задачи, т. е. с измерением кунитов. Например, для всякого кунита, содержащего v значений, в результате измерения можно получить только одно значение, но объём $V_{\text{кр}}$ регистра КК и квантовый параллелизм позволяют использовать в процессе верификации в качестве эталона громоздкую таблицу решений, которая задаётся в техническом задании (ТЗ) на разработку систем управления на жёсткой логике. На основе ТЗ выписывают булевы уравнения [7]. Примеры форм записи булевых уравнений имеют вид:

$$(x_6 \rightarrow x_1)(x_1 \rightarrow x_2 \vee x_3)(x_2 \rightarrow x_4 \vee x_1) \times (x_3 \vee x_5 \rightarrow x_6)(x_6 \rightarrow x_4 \vee x_5) = 1. \quad (1)$$

$$(x \vee y' \vee z)(x' \vee y \vee z) = 1. \quad (2)$$

$$(x_1 \vee x_2)' \oplus (x_3 \wedge x_4)' = 1. \quad (3)$$

В (1)–(3) символ \rightarrow обозначает операцию "импликация", символ \oplus — операцию "сумма по модулю 2". Уравнения используются для описания функционирования КУ в системах управления на основе жёсткой логики [6]. Подчеркнём, что в левой части (2) представлена булева формула в виде конъюнктивной нормальной формы. Более простое уравнение (3) можно представить в виде эквивалентного КУ (рисунк).



$$\mathfrak{M} = \mathfrak{M}_1 \cup \mathfrak{M}_2; \mathfrak{M}_1 \cap \mathfrak{M}_2 = \emptyset.$$

$$\mathfrak{M}_1 = \{\phi_{1,1}, \phi_{1,2}\}; \mathfrak{M}_2 = \{\phi_{2,3}\}; \rho = \overline{1, R}.$$

$$\mathfrak{N} = \{1, 2, 3, 4, 5, 6, 7\}, N = |\mathfrak{N}| = 7;$$

$$z_7 = f(x_1, x_2, x_3, x_4) = (x_1 \vee x_2)' \oplus (x_3 \wedge x_4)' = y_5 \oplus y_6.$$

Рис. Логическая сеть эквивалентного КУ

* Полный перебор часто называют "методом грубой силы" (англ. *brute-force search*, *brute-force method* или просто *brute-force*). Сложность полного перебора зависит от количества всех возможных решений задачи. Если пространство возможных решений очень велико, то при использовании классического компьютера или суперкомпьютера полный перебор может не дать результатов в течение нескольких лет или даже столетий.

На рисунке используют следующие обозначения: n — число входов КУ или аргументов уравнения (3); m — число выходов КУ; σ — число внутренних переменных y_5 и y_6 ; N — число дуг логической сети КУ.

Логическая сеть на рисунке разбита на 2 ранга ($R = 2$). К первому рангу ($\rho = 1$) относятся элементы $\phi_{1,1}$ и $\phi_{1,2}$, т. е. множество $M_1 = \{\phi_{1,1}, \phi_{1,2}\}$, где $\phi_{1,1}$ — элемент ИЛИ-НЕ, $\phi_{1,2}$ — элемент И-НЕ. Ко второму рангу ($\rho = 2$) относится элемент $\phi_{2,3}$ (M_2 — сумма по модулю 2), т. е. множество $M_2 = \{\phi_{2,3}\}$. С учетом внутренних переменных в КУ уравнение (3) можно записать в виде:

$$(x_1 \vee x_2)' \oplus (x_3 \wedge x_4)' = y_5 \oplus y_6 = z_7 = 1. \quad (4)$$

Для отражения близости задачи решения уравнения (3) к задаче решения построения тестов [18, 19], на рисунке показана константная неисправность $\omega_7 = 0$ в дуге 7 логической сети. Так как ω_7 расположена на выходе КУ, то для её обнаружения достаточно такого условия проявления, как наличие значения 1 в дуге 7. Это эквивалентно выполнению уравнения (3) или (4) на векторе $X_i (X_i = \langle 1010 \rangle)$, что показано на рисунке, где приведён вектор, обнаруживающий ω_7 . То есть вектор X_i является решением уравнений (3) и (4). Это необходимо отметить, так как решение булевых уравнений выполняется с использованием классических и квантовых D -алгоритмов. В 1966 г. последовательные D -алгоритмы и классическое исчисление кубических комплексов (ИКК) использовали для построения тестов для КУ [18, 19]. Здесь для решения булевых уравнений будут использованы квантовые D -алгоритмы (QD -алгоритмы) и матричное ИКК [20]. ИКК основано на использовании операций с многомерными кубами. Куб — это понятие, заимствованное Дж. П. Ротом из топологии для описания последовательного D -алгоритма [18]. В [18, 19] используют различные виды кубов. QD -алгоритмы применяют для решения булевых уравнений, и куб обозначен символом C_q . Куб C_q на каждом шаге q QD -алгоритма можно представить в виде многомерного вектора C_q . В координатах куба C_q отображены значения из алфавита $A = \{0, 1, \sim, \emptyset\}$, соответствующие значениям в дугах логической сети КУ. В A символ \sim обозначает неопределённое состояние, а символ \emptyset обозначает результат "пустого" пересечения в некоторой координате j . Постановка задачи верификации состоит в решении обратной задачи (задачи \bar{I}) для уравнений (3), (4) и проверки реализации таблицы решений, полученной из ТЗ. Обычно используют сокращённую таблицу решений [18, 19] (табл. 1).

Таблица 1

Сокращённая таблица решений для булевой формулы $z_7 = f(x_1, x_2, x_3, x_4)$ в уравнении (3) и (4)

$X_{p,i}$	$\langle x_1 x_2 x_3 x_4 \rangle$	z_7	$X_{p,i}$	$\langle x_1 x_2 x_3 x_4 \rangle$	z_7	$X_{p,i}$	$\langle x_1 x_2 x_3 x_4 \rangle$	z_7
$X_{p,1}$	$\langle 0 0 1 1 \rangle$	1	$X_{p,3}$	$\langle \sim 1 0 1 \rangle$	1	$X_{p,5}$	$\langle 1 0 1 0 \rangle$	1
$X_{p,2}$	$\langle \sim 1 \sim 0 \rangle$	1	$X_{p,4}$	$\langle 1 0 0 \sim \rangle$	1			

Повторим, что символ \sim в табл. 1 обозначает неопределённое (безразличное) значение в X_i . Так, вектор $X_{p,2} = \langle \sim 1 \sim 0 \rangle$ содержит множество $\beta_1 = \{\langle 0 1 0 0 \rangle, \langle 0 1 1 0 \rangle, \langle 1 1 0 0 \rangle, \langle 1 1 1 0 \rangle\}$, $|\beta_1| = 4$; вектор $X_3 = \langle \sim 1 0 1 \rangle$ — множество $\beta_2 = \{\langle 0 1 0 1 \rangle, \langle 1 1 0 1 \rangle\}$, $|\beta_2| = 2$; вектор $X_4 = \langle 1 0 0 \sim \rangle$ — множество $\beta_3 = \{\langle 1 0 0 0 \rangle, \langle 1 0 0 1 \rangle\}$, $|\beta_3| = 2$. Таблица решений читается так: если $X_{p,i} = X_{p,1} = \langle 0 0 1 1 \rangle$, то $z_7 = 1$. Если $X_{p,i} = X_{p,2} = \langle \sim 1 \sim 0 \rangle$, то $z_7 = 1$, или иначе: если $[(x_2 = 1) \& (x_4 = 0)]$, то $z_7 = 1$. По сути, в постановке задачи требуется ответить на вопрос: реализует ли построенное КУ (см. рисунок) заданное уравнение (4)? Ответом может быть да или нет (1 либо 0).

Матричное исчисление кубических комплексов

Матричное ИКК используют для формального описания QD-алгоритмов, которые также оперируют с сокращёнными таблицами истинности логических элементов [18, 19]. Сокращённые таблицы истинности позволяют существенно уменьшить перебор. Например, если для функции И-НЕ от четырех переменных x_1, x_2, x_3, x_4 полная таблица истинности содержит $2^4 = 16$ строк, то сокращённая таблица содержит всего 5 строк. Для элементов КУ далее приведены сокращённые таблицы истинности (табл. 2—5).

Таблица 2

ИЛИ-НЕ ($\varphi_{1,1}$)

№	$s_{p,k,l}$	1	2	5
1	$s_{1,1,1}$	0	0	1
2	$s_{1,1,2}$	\sim	1	0
3	$s_{1,1,3}$	1	\sim	0

Таблица 3

И-НЕ ($\varphi_{1,2}$)

№	$s_{p,k,l}$	3	4	6
1	$s_{1,2,1}$	0	\sim	1
2	$s_{1,2,2}$	\sim	0	1
3	$s_{1,2,3}$	1	1	0

Таблица 4

M2 ($\varphi_{2,3}$) — (1)

№	$s_{p,k,l}$	5	6	7
1	$s_{2,1,1}$	0	1	1
2	$s_{2,1,2}$	1	0	1

Таблица 5

M2 ($\varphi_{2,3}$) — (0)

№	5	6	7
1	0	0	0
2	1	1	0

Табл. 4 и 5, по сути, являются одной разделённой таблицей истинности элемента M2 ($\varphi_{2,3}$). Здесь используется табл. 4, в которой на выходе 7 элемента M2 ($\varphi_{2,3}$) на рисунке имеется 1, как и в уравнениях (3) и (4). В целях сокращения перебора табл. 5 не используется. Строки таблиц 2—5 могут быть объединены в конкатенации \mathcal{K}_1 и \mathcal{K}_2 в соответствии с рангами КУ (\mathcal{K}_1 для ранга $\rho = 1$ и \mathcal{K}_2 для ранга $\rho = 2$).

$$\mathcal{K}_1 = \mathcal{K}_{\text{или-не}} \parallel \mathcal{K}_{\text{и-не}} = s_{1,1,1} \parallel s_{1,1,2} \parallel s_{1,1,3} \parallel s_{1,2,1} \parallel s_{1,2,2} \parallel s_{1,2,3}. \quad (5)$$

$$\mathcal{K}_1 = \begin{matrix} 1 & 2 & 5 \\ 0 & 0 & 1 \end{matrix} \parallel \begin{matrix} 1 & 2 & 5 \\ \sim & 1 & 0 \end{matrix} \parallel \begin{matrix} 1 & 2 & 5 \\ 1 & \sim & 0 \end{matrix} \parallel \begin{matrix} 3 & 4 & 6 \\ 0 & \sim & 1 \end{matrix} \parallel \begin{matrix} 3 & 4 & 6 \\ \sim & 0 & 1 \end{matrix} \parallel \begin{matrix} 3 & 4 & 6 \\ 1 & 1 & 0 \end{matrix}. \quad (6)$$

$$\mathcal{K}_2 = \mathcal{K}_{M2} = s_{2,1,1} \parallel s_{2,1,2} = s_{2,1,1} \parallel s_{2,1,2} = \begin{matrix} 5 & 6 & 7 \\ 0 & 1 & 1 \end{matrix} \parallel \begin{matrix} 5 & 6 & 7 \\ 1 & 0 & 1 \end{matrix}. \quad (7)$$

Конкатенации \mathcal{K}_1 и \mathcal{K}_2 рассматриваются как кубы в целях эффективного использования квантового параллелизма, а значит и для ускорения решения задачи \check{I} . Подчеркнём, что при использовании КК производительность, а значит и время $\Delta t_{\text{реш}}$ определяется не тактовой частотой, а эффективным использованием квантового параллелизма. Квантовый параллелизм является частным случаем гипермассового параллелизма, и оценкой результата его использования служит время $\Delta t_{\text{реш}}$ для задачи \check{I} , которое обосновывается законом сохранения перебора (ЗСП) [21]. Математическое выражение для ЗСП достаточно просто:

$$Pr = Pr_{\text{пз}} = Pr_{\text{оз}} = \Delta t_{\text{пз}} = \Delta t_{\text{оз}} = \Delta t_{\text{реш}} = R_{\text{деп}}. \quad (8)$$

В (8) Pr — величина полного перебора, которая при использовании компьютеров определяется временем $\Delta t_{\text{реш}}$ решения задачи, $Pr_{\text{пз}}$ и $Pr_{\text{оз}}$ — величины перебора при решении прямой и обратной задачи, $\Delta t_{\text{пз}}$ и $\Delta t_{\text{оз}}$ — время решения прямой и обратной задачи, R — число рангов в эквивалентном КУ или в булевом уравне-

нии, деп — внесистемная дискретная единица перебора*. Одна деп равна времени выполнения одной элементарной операции α_j пересечения в координате j . Операция α_j является основной операцией в ИКК, а значит и во всяком D -алгоритме, включая и QD -алгоритмы. ЗСП показывает, как решать сложные задачи с полиномиальным временем $\Delta t_{\text{реш}}$. Табличное описание операции α_j в классическом ИКК приведено в табл. 6, где используется алфавит $A = \{0, 1, \sim, \emptyset\}$.

Напомним, что когда покоординатно пересекаются два многомерных куба $C_q \cap C_g$, то символ \emptyset из A обозначает пустое (противоречивое, неудачное) пересечение в координате j : $c_{j,q} \cap \theta_{j,g} = c_{j,(q+1)} = \emptyset$, где $c_{j,q}$ — значение в координате j куба C_q ; $\theta_{j,g}$ — значение в координате j куба C_g . После выполнения операции α_j в координате j получим результат пересечения — значение $c_{j,(q+1)}$ в координате j куба C_{q+1} . В качестве куба C_g может выступать конкатенация \mathcal{K} .

В матричном ИКК используем другой алфавит \tilde{A} : $\tilde{A} = \{0, 1, 2, 3\}$. В \tilde{A} символы 0 и 1 остаются теми же, что и в A , а символы \sim и \emptyset изменяются: $\sim \rightarrow 2$ и $\emptyset \rightarrow 3$. Их смысловое значение остаётся прежним. Важно, что $|A| = |\tilde{A}| = 4$. Это позволяет одновременно хранить все 4 символа \tilde{A} в одном куквадрите. Тогда табл. 6 преобразуется в табл. 7.

Объём памяти, доступный в КК или в КвУ, определяется объёмом гильбертова пространства \mathcal{H} квантового регистра. "В гильбертовом пространстве много места" [22]. В общем случае

регистр КвУ может быть построен из кунитов. С позиций квантовой механики число v может быть больше любого наперёд заданного числа \tilde{N} . В настоящее время практически достижимым считается число $v \leq 10^7$. В этом случае, как считает американский физик Д. Авшалом, физической основой кунита могут быть спины атомных ядер [23]. Но недавно ученые из венского университета достигли рекордного успеха. Им удалось ввести в состояние суперпозиции молекулу с формулой: $C_{707}H_{260}F_{908}N_{16}S_{53}Zn_4$. Удержание в состоянии суперпозиции такой большой молекулы, содержащей 1948 атомов и, следовательно, такое же количество атомных ядер, может быть использовано в квантовом компьютеростроении. Возможно такая молекула в состоянии суперпозиции станет физической основой кунита, и тогда максимальное количество v_{max} чисел, содержащихся в таком куните, может достигать $(10^7)^{1948} = 10^{13636}$. Конечно, достигнуть максимума непросто, и, может быть, полностью достигнуть величины v_{max} не удастся. Здесь уместно привести вектор состояния** кунита — вектор $|\psi\rangle_v$, а также вектор $V_{v,a}$ — вектор-столбец амплитуд вероятности или просто вектор амплитуд:

$$|\psi\rangle_v = a_1|0\rangle + a_2|1\rangle + \dots + a_\xi|\xi-1\rangle + \dots + a_v|(v-1)\rangle \quad (9)$$

$$\left| \sum_{\xi=1}^v |a_\xi|^2 \right| = 1.$$

$$V_{v,a} = [a_1, a_2, \dots, a_\xi, \dots, a_v]^T. \quad (10)$$

Таблица 6

Правила выполнения операции α_j в координате j : $c_{j,q} \cap \theta_{j,g} = c_{j,(q+1)}$ (используется алфавит A)

№	$c_{j,q} \cap \theta_{j,g}$	$c_{j,(q+1)}$
1	$0 \cap 0$	0
2	$0 \cap 1$	\emptyset
3	$0 \cap \sim$	0

№	$c_{j,q} \cap \theta_{j,g}$	$c_{j,(q+1)}$
4	$1 \cap 0$	\emptyset
5	$1 \cap 1$	1
6	$1 \cap \sim$	1

№	$c_{j,q} \cap \theta_{j,g}$	$c_{j,(q+1)}$
7	$\sim \cap 0$	0
8	$\sim \cap 1$	1
9	$\sim \cap \sim$	\sim

Таблица 7

Правила выполнения операции α_j в координате j : $c_{j,q} \cap \theta_{j,g} = c_{j,(q+1)}$ (используется алфавит \tilde{A})

№	$c_{j,q} \cap \theta_{j,g}$	$c_{j,(q+1)}$
1	$0 \cap 0$	0
2	$0 \cap 1$	3
3	$0 \cap 2$	0

№	$c_{j,q} \cap \theta_{j,g}$	$c_{j,(q+1)}$
4	$1 \cap 0$	3
5	$1 \cap 1$	1
6	$1 \cap 2$	1

№	$c_{j,q} \cap \theta_{j,g}$	$c_{j,(q+1)}$
7	$2 \cap 0$	0
8	$2 \cap 1$	1
9	$2 \cap 2$	2

* Внесистемные единицы не входят ни в одну из физических систем единиц. Внесистемные единицы можно разделить на независимые (определяемые без помощи других единиц, например градус Цельсия, полный угол (оборот), бит, байт и др.) и произвольно выбранные, но определяемые через другие единицы (например единица мощности лошадиная сила (735,499 Вт), единицы давления техническая атмосфера (1 кгс/см²), миллиметр ртутного столба (133,322 н/м²), бар (105 н/м²), бит в секунду, байт в секунду и др.).

** Иногда вектор состояния называют волновой функцией. Это связано с тем, что существуют два варианта квантовой механики (КМ): матричная КМ, идущая от работ В. Гейзенберга, и волновая КМ, идущая от работ Э. Шрёдингера.

Векторы $V_{v,a}$ используют для работы с унитарными матрицами U^\dagger . Векторы состояния для кубитов, кутритов и куквадритов приведены в (11)–(13).

$$|\psi\rangle_v = |\psi\rangle_2 = a_1|0\rangle + a_2|1\rangle \quad |a_1|^2 + |a_2|^2 = 1. \quad (11)$$

$$|\psi\rangle_v = |\psi\rangle_3 = a_1|0\rangle + a_2|1\rangle + a_3|2\rangle \quad |a_1|^2 + |a_2|^2 + |a_3|^2 = 1. \quad (12)$$

$$|\psi\rangle_v = |\psi\rangle_4 = a_1|0\rangle + a_2|1\rangle + a_3|2\rangle + a_4|3\rangle \quad |a_1|^2 + |a_2|^2 + |a_3|^2 + |a_4|^2 = 1. \quad (13)$$

Для примера приведём также вектор-столбец $V_{4,a}$ — вектор амплитуд для куквадрита:

$$V_{4,a} = [a_1, a_2, a_3, a_4]^T. \quad (14)$$

Подчеркнём, квантовые разряды и квантовые вентили рассматриваются здесь как абстрактные математические объекты с некоторыми заданными свойствами. Их физическая основа здесь не учитывается. Такой подход используется часто, например, в работе Нильсена М., Чанга И. [22]. Регистр QR КвУ длины L (где L — число кунитов в регистре QR) можно условно разбить на два регистра QR_1 и QR_2 . В данном случае регистры QR_1 и QR_2 построены из кунитов, и их длины L_1 и L_2 равны ($L_1=L_2=L/2$). Пространство $\mathcal{H}_{1,m}$ каждого кунита $k_{1,m}$ регистра QR_1

можно разбить на \tilde{v} гильбертовых подпространств $\mathcal{H}_{1,m}$ ($\mathcal{H}_{1,m,j} \triangleleft \mathcal{H}_{1,m}$), а пространство каждого кунита $k_{2,m}$ регистра QR_2 — на \tilde{v} подпространств $\mathcal{H}_{2,m}$ ($\mathcal{H}_{2,m,j} \triangleleft \mathcal{H}_{2,m}$) [24]. Каждое подпространство $\mathcal{H}_{1,m,j}$ и $\mathcal{H}_{2,m,j}$, по сути, является гильбертовым пространством куквадрита, которое может одновременно содержать 4 числа, а значит оно может содержать все числа алфавита \tilde{A} . В этом случае для каждого кунита $k_{1,m}$ или $k_{2,m}$ $v=4\tilde{v}$, где \tilde{v} — число подпространств куквадритов в одном куните. Подпространства $\mathcal{H}_{1,m,j}$ и $\mathcal{H}_{2,m,j}$, содержащие числа из \tilde{A} , необходимы для выполнения операции α_j , которая применяется для покоординатного пересечения кубов: $C_{q \cap C_g} = C_{q+1}$.

В случае КвУ операция α_j выполняется с помощью унитарной матрицы $U_{R_j}^\dagger$ — элементарной матрицы Рота. Операция α_j выполняется в соответствии с табл. 7 над числами в двух подпространствах $\mathcal{H}_{1,m,j}$ и $\mathcal{H}_{2,m,j}$ (по сути, в пространствах двух куквадритов). Один из вариантов матрицы Рота $U_{R_j}^\dagger$ имеет вид (15).

Матрица $U_{R_j}^\dagger$ (15) проверена на унитарность: произведение $U_{R_j}^\dagger \times U_{R_j}^{\dagger T} = I$, где $U_{R_j}^{\dagger T}$ — транспонированная матрица, I — единичная матрица. Исходный вектор $|\psi\rangle_{\text{исх}}$ — вектор состояния для двух подпространств $\mathcal{H}_{1,m,j}$ и $\mathcal{H}_{2,m,j}$ в целях реализации операции α_j с помощью матрицы $U_{R_j}^\dagger$ имеет вид (16).

$$U_{R_j}^\dagger = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 6 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 7 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 8 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 9 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 10 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 11 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 12 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 13 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 \end{bmatrix} \quad (15)$$

$$|\psi\rangle_{\text{исх}} = 1|00\rangle + 0|20\rangle + 0|11\rangle + 0|21\rangle + 0|02\rangle + 0|30\rangle + 0|22\rangle + 0|12\rangle + 0|31\rangle + 0|01\rangle + 0|03\rangle + 0|10\rangle + 0|13\rangle. \quad (16)$$

Для исходного вектора $|\psi\rangle_{\text{исх}}$ (т. е. до выполнения операции α_j , которая выполняется с помощью матрицы $U_{R_j}^\dagger$) вектор амплитуд имеет вид: $V_{\text{исх}} = [100000000000]$. В результате умножения матрицы $U_{R_j}^\dagger$ на вектор $V_{\text{исх}}$ получим: $V_{\text{рез}} = U_{R_j}^\dagger \times V_{\text{исх}} = [010000000000]$. В результате умножения вектор состояния $|\psi\rangle_{\text{рез}}$ будет, например, иметь вид:

$$|\psi\rangle_{\text{исх}} = 0|00\rangle + 1|20\rangle + 0|11\rangle + 0|21\rangle + 0|02\rangle + 0|30\rangle + 0|22\rangle + 0|12\rangle + 0|31\rangle + 0|01\rangle + 0|03\rangle + 0|10\rangle + 0|13\rangle. \quad (17)$$

Это означает, что во втором подпространстве $\mathcal{H}_{2,m,j}$ появится символ 0, что соответствует пересечению $0 \cap 0 = 0$ в табл. 7. В (16) и (17) подчеркнуты те 4 компоненты вектора $|\psi\rangle_{\text{исх}}$, которые невозможны, то есть не могут быть исходными по условиям задачи, так как в табл. 7 только 9 строк. Они добавлены в вектор $|\psi\rangle_{\text{исх}}$ для построения унитарной матрицы U_R^\dagger и квантовой реализации операции α_j в виде табл. 8.

Таблица 8

Матричная реализация таблицы 7

1.	$ 00\rangle \rightarrow 20\rangle$	$V_{\text{исх}} = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$ $V_{\text{рез}} = [0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$
2.	$ 20\rangle \rightarrow 00\rangle$	$V_{\text{исх}} = [0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$ $V_{\text{рез}} = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$
3.	$ 11\rangle \rightarrow 21\rangle$	$V_{\text{исх}} = [0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$ $V_{\text{рез}} = [0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$
4.	$ 21\rangle \rightarrow 11\rangle$	$V_{\text{исх}} = [0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$ $V_{\text{рез}} = [0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$
5.	$ 02\rangle \rightarrow 30\rangle$	$V_{\text{исх}} = [0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$ $V_{\text{рез}} = [0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$
6.	$ 30\rangle \rightarrow 02\rangle$ (невозможен)	$V_{\text{исх}} = [0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$ $V_{\text{рез}} = [0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$
7.	$ 22\rangle \rightarrow 22\rangle$	$V_{\text{исх}} = [0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0]$ $V_{\text{рез}} = [0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0]$
8.	$ 12\rangle \rightarrow 31\rangle$	$V_{\text{исх}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0]$ $V_{\text{рез}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0]$
9.	$ 31\rangle \rightarrow 12\rangle$ (невозможен)	$V_{\text{исх}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0]$ $V_{\text{рез}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0]$
10.	$ 01\rangle \rightarrow 03\rangle$	$V_{\text{исх}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]$ $V_{\text{рез}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0]$
11.	$ 03\rangle \rightarrow 01\rangle$ (невозможен)	$V_{\text{исх}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0]$ $V_{\text{рез}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]$
12.	$ 10\rangle \rightarrow 13\rangle$	$V_{\text{исх}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0]$ $V_{\text{рез}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1]$
13.	$ 13\rangle \rightarrow 10\rangle$ (невозможен)	$V_{\text{исх}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1]$ $V_{\text{рез}} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0]$

Иными словами, табл. 8 представляет собой матричную реализацию операции α_j с помощью матрицы (15) табл. 7 — таблицы пересечений (результат пересечения появляется в подпространстве $\mathcal{H}_{2,m,j}$ пространства $\mathcal{H}_{2,m}$ второго кунита).

Число подпространств в пространстве $\mathcal{H}_{1,m}$, как и в пространстве $\mathcal{H}_{2,m}$ каждого кунита $k_{1,m}$ или $k_{2,m}$ равно \tilde{v} . Проверив матрицу (15) на унитарность и проверив реализацию операции α_j этой матрицей в табл. 8, построим унитарную блочно-диагональную матрицу $U_{R,BD}^\dagger$ для эффективного использования квантового параллелизма, когда на шаге q QD -алгоритма выполняется не одна операция α_j с помощью матрицы U_R^\dagger , а все возможные операции α_j , допустимые на этом шаге (описание QD -алгоритмов приведено ниже). В общем случае, блочно-диагональная матрица A имеет вид:

$$A = \begin{bmatrix} A_1 & 0 & \dots & 0 \\ 0 & A_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & A_{\tilde{v}} \end{bmatrix}. \quad (18)$$

В (18) $A_1, A_2, \dots, A_{\tilde{v}}$ являются блоками (подматрицами) в блочно-диагональной матрице A . Для построения унитарной блочно-диагональной матрицы $U_{R,BD}^\dagger$, следует блоки $A_1, A_2, \dots, A_{\tilde{v}}$ в (18) заменить на матрицы $U_{R_1}^\dagger, U_{R_2}^\dagger, \dots, U_{R_j}^\dagger, \dots, U_{R_{\tilde{v}}}^\dagger$ (16), которые станут блоками в матрице $U_{R,BD}^\dagger$.

$$U_{R,BD}^\dagger = \begin{bmatrix} U_{R_1} & 0 & \dots & 0 \\ 0 & U_{R_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & U_{R_{\tilde{v}}} \end{bmatrix}. \quad (19)$$

Иногда матрицу $U_{R,BD}^\dagger$ записывают в виде:

$$U_{R,BD}^\dagger = \text{diag}\{U_{R_1}^\dagger, U_{R_2}^\dagger, \dots, U_{R_j}^\dagger, \dots, U_{R_{\tilde{v}}}^\dagger\}. \quad (20)$$

Унитарную матрицу (19) или (20), где матрица $U_{R_j}^\dagger$ является блоком, теперь можно использовать для пересечения кубов при выполнении QD -алгоритмов. В начальный момент в соответствии с уравнением (4) и КУ на рис. 1 при использовании алфавита A куб C_0 имеет вид:

$$C_0 = \begin{matrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \sim & \sim & \sim & \sim & \sim & \sim & \sim & 1 \end{matrix} \quad (21)$$

При описании QD -алгоритмов используется алфавит \tilde{A} и куб C_0 следующего вида:

$$C_0 = \begin{matrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 1 \end{matrix} \quad (22)$$

По мере выполнения шагов QD -алгоритма, значения в координатах куба C_0 будут изменяться — постепенно неопределённые значения (" \sim " в алфавите A или " 2 " в \tilde{A}) будут заме-

няться на определённые значения 0 или 1. Исходный куб C_0 можно пересечь, например, с конкатенацией \mathcal{K}_1 : $C_0 \cap \mathcal{K}_1 = C_1$. На подготовительном этапе для выполнения пересечения $C_0 \cap \mathcal{K}_1$ и эффективного использования квантового параллелизма, куб C_0 должен быть представлен

в виде, удобном для пересечения с конкатенацией \mathcal{K}_1 :

$$C_0 = \begin{matrix} 1 & 2 & 5 & 1 & 2 & 5 & 1 & 2 & 5 & 3 & 4 & 6 & 3 & 4 & 6 & 3 & 4 & 6 & 7 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 1 \end{matrix} \quad (23)$$

Такое представление выполняет классический компьютер, который управляет КК или КвУ. После этого куб C_0 (23) может быть записан в кунит $k_{1,m}$. В кунит $k_{2,m}$ записывается \mathcal{K}_1 и покоординатно выполняется пересечение $C_0 \cap \mathcal{K}_1$ с использованием блоков $U_{R_j}^\dagger$ матрицы $U_{R,ВД}^\dagger$:

$$\begin{aligned} C_0 &= \begin{matrix} 1 & 2 & 5 & 1 & 2 & 5 & 1 & 2 & 5 & 3 & 4 & 6 & 3 & 4 & 6 & 3 & 4 & 6 & 7 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 1 \end{matrix} \\ \mathcal{K}_1 &= \begin{matrix} 0 & 0 & 1 & 2 & 1 & 0 & 1 & 2 & 0 & 0 & 2 & 1 & 2 & 0 & 1 & 1 & 1 & 0 & 2 \end{matrix} \\ \cap &= \text{-----} \\ C_1 &= \begin{matrix} 0 & 0 & 1 & 2 & 1 & 0 & 1 & 2 & 0 & 0 & 2 & 1 & 2 & 0 & 1 & 1 & 1 & 0 & 1 \end{matrix} \end{aligned} \quad (24)$$

Заметим, что пересечение $C_0 \cap \mathcal{K}_1$ выполняется на первом шаге QD-алгоритма в случае решения прямой задачи (задачи \check{D}). Так как старая классическая модель вычислений в виде классической машины Тьюринга подвергается жёсткой критике и для КК не работает [25, 26], необходима новая модель вычислений [17]. Квантовая машина Тьюринга и попытки её использовать также оказались непопулярными: в её основе лежат кубиты, а в настоящее время уже известны прототипы КК, у которых, как отмечалось выше, регистр построен на кутритах, куквадритах и кунитах.

Описание квантовых алгоритмов

Перед выполнением QD-алгоритмов исполняются подготовительные операции, например такие, как выписывание конкатенаций \mathcal{K}_1 и \mathcal{K}_2 , выписывание куба C_0 (см. выше (6), (7), (21), (22) и (23)). После пересечений $C_0 \cap \mathcal{K}_1 = C_1$ (см. (24)) на шаге q QD-алгоритма следует операция фильтрации, а после завершения всех квантовых вычислений следует процедура измерения. Процедура фильтрации необходима для удаления тех фрагментов полученных кубов, в координатах которых содержатся символы 3 (символу 3 в табл. 7 соответствует символ \emptyset в табл. 6). Фрагментами полученных кубов, удаляемых после выполнения пересечения $C_{k-1} \cap \mathcal{K}_k = C_k$, являются строки таблиц истинно-

сти, у которых хотя бы в одной координате содержится символ 3 (т. е. \emptyset) [12, 18–21]. Это так называемый квантовый мусор [22]. Пролиферация возникает в случае появления копий куба C_q на шаге q (см. далее). Процедуры фильтрации и пролиферации реализуется схемами фильтрации и пролиферации, которые построены на новых однородных и неоднородных квантовых логических вентилях, реализующих соответствующие блочно-диагональные матрицы [24, 27].

Теперь после проведения вспомогательных операций можно привести описание QD-алгоритмов по шагам q в целях решения задачи \check{I} . Исходным кубом C_0 является куб вида (22). Так как для решения задачи \check{I} движение QD-алгоритма начинается от выхода эквивалентного КУ, то сначала будет использовано покоординатное пересечение $C_0 \cap \mathcal{K}_2 = C_1$. Отметим, что исходный куб C_0 адаптирован под конкатенацию \mathcal{K}_2 , выражение которой приведено в (7), и которая связана с рангом $\rho = 2$ (см. рисунок):

	1	2	3	4	5	6	7		5	6	7		
$C_0 =$	2	2	2	2	2	2	1		2	2	1		(25)

Пример решения задачи \check{I} с помощью QD-алгоритма

Описание алгоритма по шагам.

1. Шаг 1 ($q=1$). На этом шаге выполняется покоординатное пересечение $C_0 \cap \mathcal{K}_2 = C_1$.

$$\begin{aligned} C_0 &= \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 5 & 6 & 7 \\ 2 & 2 & 2 & 2 & 2 & 2 & 1 & 2 & 2 & 1 \end{matrix} \\ \mathcal{K}_2 &= \begin{matrix} 2 & 2 & 2 & 2 & 0 & 1 & 1 & 1 & 0 & 1 \end{matrix} \\ \cap &= \text{-----} \\ C_1 &= \begin{matrix} 2 & 2 & 2 & 2 & 0 & 1 & 1 & 1 & 0 & 1 \end{matrix} \end{aligned} \quad (26)$$

После выполнения пересечения $C_0 \cap \mathcal{K}_2 = C_1$ в (26) выполняется процедура фильтрации, которая не находит пустых результатов в координатах куба C_1 в виде цифры 3 (символа \emptyset) в одной из координат. Используя процедуру пролиферации, выпишем результат пересечения $C_0 \cap \mathcal{K}_2$ в закодированном в алфавите виде \check{A} — в виде двух кубов $C_{1,1}$ и $C_{1,2}$:

$$C_{1,1} = \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 2 & 2 & 0 & 1 & 1; \end{matrix} \quad (27)$$

$$C_{1,2} = \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 2 & 2 & 1 & 0 & 1. \end{matrix} \quad (28)$$

Тогда на следующем шаге ($q=2$) необходимо одновременно выполнить два пересечения $C_{1,1} \cap \mathcal{K}_1$ и $C_{1,2} \cap \mathcal{K}_1$. Кубы $C_{1,1}$ и $C_{1,2}$ должны быть записаны в форме, позволяющей осуществить пересечения с конкатенацией \mathcal{K}_1 (см. (6)). Тогда кубы $C_{1,1}$ и $C_{1,2}$, адаптированные к пересечению с конкатенацией \mathcal{K}_1 , имеют вид (29) и (30):

$$C_{1,1} = \begin{matrix} & 1 & 2 & 5 & 1 & 2 & 5 & 1 & 2 & 5 & & 3 & 4 & 6 & 7 & 3 & 4 & 6 & 7 & 3 & 4 & 6 & 7 \\ 2 & 2 & 0 & 2 & 2 & 0 & 2 & 2 & 0 & & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 1 \end{matrix} \quad (29)$$

$$C_{1,2} = \begin{matrix} & 1 & 2 & 5 & 1 & 2 & 5 & 1 & 2 & 5 & & 3 & 4 & 6 & 7 & 3 & 4 & 6 & 7 & 3 & 4 & 6 & 7 \\ 2 & 2 & 1 & 2 & 2 & 1 & 2 & 2 & 1 & & 2 & 2 & 0 & 1 & 2 & 2 & 0 & 1 & 2 & 2 & 0 & 1 \end{matrix} \quad (30)$$

2. Шаг 2 ($q=2$). На этом шаге одновременно выполняют два покоординатных пересечения $C_{1,1} \cap \mathcal{K}_1 = C_{2,1}$ и $C_{1,2} \cap \mathcal{K}_1 = C_{2,2}$. Символ \emptyset в алфавите \tilde{A} обозначается цифрой 3.

$$\begin{matrix} & 1 & 2 & 5 & | & 1 & 2 & 5 & | & 1 & 2 & 5 & & 3 & 4 & 6 & 7 & 3 & 4 & 6 & 7 & 3 & 4 & 6 & 7 \\ C_{1,1} = & 2 & 2 & 0 & & 2 & 2 & 0 & & 2 & 2 & 0 & & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 1 \\ \mathcal{K}_1 = & 0 & 0 & 1 & & 2 & 1 & 0 & & 1 & 2 & 0 & & 0 & 2 & 1 & 2 & 2 & 0 & 1 & 2 & 1 & 1 & 0 & 2 \\ C_{2,1} = & 0 & 0 & 3 & & 2 & 1 & 0 & & 1 & 2 & 0 & & 0 & 2 & 1 & 1 & 2 & 0 & 1 & 1 & 1 & 1 & 3 & 1 \\ & 1 & 2 & 5 & & 1 & 2 & 5 & & 1 & 2 & 5 & & 3 & 4 & 6 & 7 & 3 & 4 & 6 & 7 & 3 & 4 & 6 & 7 \\ C_{1,2} = & 2 & 2 & 1 & & 2 & 2 & 1 & & 2 & 2 & 1 & & 2 & 2 & 0 & 1 & 2 & 2 & 0 & 1 & 2 & 2 & 0 & 1 \\ \mathcal{K}_1 = & 0 & 0 & 1 & & 2 & 1 & 0 & & 1 & 2 & 0 & & 0 & 2 & 1 & 2 & 2 & 0 & 1 & 2 & 1 & 1 & 0 & 2 \\ C_{2,2} = & 0 & 0 & 1 & & 2 & 1 & 3 & & 1 & 2 & 3 & & 0 & 2 & 3 & 1 & 2 & 0 & 3 & 1 & 1 & 1 & 0 & 1 \end{matrix} \quad (31)$$

Схемы фильтрации и пролиферации удаляют подстроки кубов $C_{2,1}$ и $C_{2,2}$, которые содержат символы \emptyset (цифру 3). Слово "удаляют" означает, что подстроки превращают в "квантовый мусор" и записывают его в свободный кунит [22]. После удаления "квантового мусора" схемы фильтрации и пролиферации выдают следующее множество решений уравнения (3) или (4) в виде множества векторов $\gamma = \gamma_{1,1} \cup \gamma_{1,2}$, где $\gamma_{1,1} = \{\langle 2102 \rangle, \langle 2120 \rangle, \langle 1202 \rangle, \langle 1220 \rangle\}$. Всякий вектор $X_i \in \gamma_{1,1}$ является неполностью определённым вектором, так как содержит цифру 2 (~). $\gamma_{2,2} = \{\langle 0011 \rangle\}$. Вектор $\langle 0011 \rangle$ — полностью определённый вектор. Тогда $\gamma = \{\langle 2102 \rangle, \langle 2120 \rangle, \langle 1202 \rangle, \langle 1220 \rangle, \langle 0011 \rangle\}$. Если пересечь каждый вектор $X_i \in \beta$ (см. табл. 1) со всеми векторами $X_i \in \gamma$, то получим следующее пересечение: $\beta_p = \{\langle 0011 \rangle, \langle 2120 \rangle, \langle 2101 \rangle, \langle 1002 \rangle, \langle 1010 \rangle\}$ в алфавите \tilde{A} (табл. 1).

$X_{p,i}$	$X_i = \langle 2102 \rangle$	$X_i = \langle 2120 \rangle$	$X_i = \langle 1202 \rangle$	$X_i = \langle 1220 \rangle$	$X_i = \langle 0011 \rangle$
	$\begin{matrix} 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 1 \\ 2 & 1 & 0 & 2 \\ - & - & - & - \end{matrix}$	$\begin{matrix} 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 1 \\ 2 & 1 & 2 & 0 \\ - & - & - & - \end{matrix}$	$\begin{matrix} 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 1 \\ 1 & 2 & 0 & 2 \\ - & - & - & - \end{matrix}$	$\begin{matrix} 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 1 \\ 1 & 2 & 2 & 0 \\ - & - & - & - \end{matrix}$	$\begin{matrix} 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ - & - & - & - \end{matrix}$
	$0 \emptyset \emptyset 1$	$0 \emptyset 1 \emptyset$	$\emptyset 0 \emptyset 1$	$\emptyset 0 1 \emptyset$	$0 0 1 1$
$\langle 2120 \rangle$	$2 1 2 0$	$2 1 2 0$	$2 1 2 0$	$2 1 2 0$	$2 1 2 0$
\cap	$2 1 0 2$	$2 1 2 0$	$1 2 0 2$	$1 2 2 0$	$0 0 1 1$
	$2 1 0 0$	$2 1 2 0$	$1 1 0 0$	$1 1 2 0$	$0 \emptyset 1 \emptyset$
$\langle 2101 \rangle$	$2 1 0 1$	$2 1 0 1$	$2 1 0 1$	$2 1 0 1$	$2 1 0 1$
\cap	$2 1 0 2$	$2 1 2 0$	$1 2 0 2$	$1 2 2 0$	$0 0 1 1$
	$2 1 0 1$	$2 1 0 \emptyset$	$1 1 0 1$	$1 1 0 \emptyset$	$0 \emptyset \emptyset 1$
$\langle 1002 \rangle$	$1 0 0 2$	$1 0 0 2$	$1 0 0 2$	$1 0 0 2$	$1 0 0 2$
\cap	$2 1 0 2$	$2 1 2 0$	$1 2 0 2$	$1 2 2 0$	$0 0 1 1$
	$1 \emptyset 0 2$	$1 \emptyset 0 9$	$1 0 0 2$	$1 0 0 0$	$\emptyset 0 \emptyset 1$
$\langle 1010 \rangle$	$1 0 1 0$	$1 0 1 0$	$1 0 1 0$	$1 0 1 0$	$1 0 1 0$
\cap	$2 1 0 2$	$2 1 2 0$	$1 2 0 2$	$1 2 2 0$	$0 0 1 1$
	$1 \emptyset \emptyset 0$	$1 \emptyset 1 0$	$1 0 \emptyset 0$	$1 0 1 0$	$\emptyset 0 1 \emptyset$

Схемы фильтрации и пролиферации выделяют десять векторов X_i , которые образуют множество $\beta = \{\langle 0011 \rangle, \langle 2100 \rangle, \langle 2120 \rangle, \langle 1100 \rangle, \langle 1120 \rangle, \langle 2102 \rangle, \langle 1101 \rangle, \langle 1002 \rangle, \langle 1000 \rangle, \langle 1010 \rangle\}$, которое записано в ТЗ. В результате получили те же десять векторов, на которых выполняются уравнения (3) и (4), т. е. у построенного КУ (см. рисунок) на выходе появляется 1, когда на его входе появляется вектор X_i ($\forall X_i \in \beta^* f(x_1, x_2, x_3, x_4) = 1$). Таким образом, задача верификации решена, и в сигнальном разряде (это может быть кунит, а может быть и кубит) будет записана 1, что после измерения укажет на получение ответа на вопрос, поставленный в задаче и, значит, построенное КУ (рисунок) верно реализует ТЗ. В общем случае можно получить некоторое число векторов $X_i \in \beta^*$, но не все векторы [28]. То есть нельзя получить в явном виде все векторы $X_i \in \beta^*$. Такова специфика квантовых вычислений, и нам после многих вычислительных экспериментов, выполненных на КК, придётся научиться доверять результатам квантовых вычислений после измерения сигнального кунита или кубита. Мы уже привыкли доверять автопилотам, управляющим самолётами в полёте (кроме взлёта и посадки, выполняемых вручную). Мы привыкаем доверять беспилотным автомобилям, движущимся в среде автомобильного движения. Мы привыкнем доверять и измерению сигнального кунита, говорящему о результатах формальной верификации.

Теперь нетрудно подсчитать необходимые числа v и L . В процессе выполнения QD -алгоритма наибольшее число \tilde{v} пересечений приведено в (32), где $\tilde{v} = 25$. Тогда $v = 4 \times \tilde{v} = 100$. Учитывая, что КвУ должен решать и другие задачи, необходимое число $v = 108$, а $L = 10$, включая и сигнальный кунит. Для решения данной задачи чисел $v = 108$, а $L = 10$ вполне достаточно. Время $\Delta t_{\text{реш}} = R = 2\text{деп}$ (см. (8)). Но здесь при решении задачи верификации выполняется ещё одно пересечение (32). Следовательно, $\Delta t_{\text{реш}} = 3\text{деп}$. Число μ квантовых вентилях, каждый из которых реализует матрицу $U_{\text{Р,БД}}^\dagger$, содержащую \tilde{v} блоков (подматриц) (см. (19) и (20)), равно 5. По аналогии точно также можно решить задачу верификации для КУ, реализующих уравнения (1) и (2).

Заключение

В условиях развития систем искусственного интеллекта, например создания беспилотных автомобилей, и появления сложных систем логического управления на жёсткой логике, к которым предъявляются повышенные требования по надёжности, возникают новые проблемы в области их проектирования. Так, на важном этапе формальной верификации цифровых схем и микросхем, которые иногда производят в

миллионах экземпляров, особенно сложной является проблема обнаружения ошибок проектирования. Использование классических компьютеров и суперкомпьютеров для решения этой проблемы приводит лишь к ограниченному успеху. Но появление КК и перспективы их развития, связанные с квантовым превосходством существующих КК, а также с их быстрым развитием, вселяют оптимизм. Однако не известен ни один квантовый алгоритм, способный решить проблему их формальной верификации. В данной работе сделана попытка исправить этот пробел, так как высокая производительность уже существующих квантовых компьютеров и КК будущего основана не на тактовой частоте квантовых процессоров, а на эффективном использовании механизма квантового параллелизма.

Ускорение решения той или иной вычислительной задачи с помощью квантового алгоритма является нетривиальной проблемой. Как утверждает профессор МГУ Ю. И. Ожигов: *"...с абстрактной точки зрения достижение квантового ускорения — это своеобразное произведение искусства, которое встречается не так часто"*. Именно поэтому некоторые квантовые вычисления крайне трудно применять эффективно, и никакие программы, реализующие классические алгоритмы для традиционного компьютера, на КК работать не могут. Для эффективного использования механизма квантового параллелизма необходимо либо создать совершенно новые квантовые алгоритмы, адаптированные для использования этого механизма в целях решения сложных задач, либо, что чаще, приходится прибегать к модификации известных классических алгоритмов в целях адаптации их к исполнению на КК. В данной работе на основе закона сохранения перебора и матричной модификации классического исчисления кубических комплексов были созданы квантовые D -алгоритмы, имеющие существенные различия в сравнении с классическими D -алгоритмами Дж. П. Рота. Эффективность использования предложенных алгоритмов определяется простой оценкой их вычислительной сложности на основе закона сохранения перебора, так что в случае решения задач формальной верификации цифровых схем квантовые D -алгоритмы можно отнести к классу эффективных алгоритмов с полиномиальной временной сложностью. При использовании для этих же целей классических алгоритмов, временная сложность в некоторых случаях является экспоненциальной, что, как правило, является неприемлемым.

Литература

1. Манушакян К. Г. Технические средства телематики. Курс лекций по микропроцессорной технике. Часть 1. Утверждено в качестве учебного пособия редсоветом МАДИ (ГТУ). — М.: Московский автомобильно-дорожный институт (ГТУ), 2007 [Электрон-

ный ресурс]. URL: <http://lib.madi.ru/fel/fel1/fel08E014.pdf> (дата обращения: 16.06.2022).

2. Конспект лекций по дисциплине информатика на тему "Управляющие устройства с жёсткой логикой" [Электронный ресурс]. URL: <https://infourok.ru/konspekt-lekciy-po-discipline-informatika-na-temu-uu-s-zhestkoy-logikoy-2923014.html> (дата обращения: 16.06.2022).

3. Норин Е. Вирус страшнее бомбы. Как хакеры уничтожили ядерный завод в Иране [Электронный ресурс]. URL: <https://life.ru/p/1047800> (дата обращения: 16.06.2022).

4. "Физприбор" объявил вторую волну конкурса для хакеров, способных взломать уникальную систему защиты [Электронный ресурс]. URL: <https://ren.tv/news/v-rossii/253187-fizpribor-obiavit-vtoruiu-volnu-konkursa-dlia-khakerov-sposobnykh-vzломat-unikalnuu-sistemu-zashchity> (дата обращения: 16.06.2022).

5. Жёсткая логика [Электронный ресурс]. URL: <https://kartaslov.ru/карта-знаний/Жёсткая+логика> (дата обращения: 16.06.2022).

6. Закревский А. Д. Параллельные алгоритмы логического управления. Изд. 3-е. — М.: Едиториал УРСС, 2012.

7. Roth J. P. Hardware verification // IEEE Trans. Computers. 1977. V. C-26. № 12. P. 1292—1291.

8. Kumar J. Prototyping the M68060 for Concurrent Verification // IEEE Design and Test of Computer. 1997. V. 14. № 1. P. 34—41.

9. Maliniak D. Systems Verifies, Emulates Monsters ICs in A Flash // Electronic Design. August 20. 2001. V. 49. № 17. P. 52—54.

10. Jones R. B., O'Leary J. W., Seager C. J., Aagaard M. D., Meiham T. F. Practical Formal Verification in Microprocessor Design // IEEE Design & Test of Computers. 2001. V. 18. № 4. P. 16—25.

11. Правильщиков П. А. Проблема верификации БИС и её решение путём использования "вычислительного решета" на основе закона сохранения переноса // Оборонный комплекс — научно-техническому прогрессу России. 2002. № 4. С. 45—61.

12. Правильщиков П. А. Теоретико-множественные основания новой модели вычислений — квантового генератора тестов // Информационные технологии в проектировании и производстве. 2017. № 3. С. 20—28.

13. Imany P., Jaramillo-Villegas J. A., Alshaykh M. S., Lukens J. M. et al. High-dimensional optical quantum logic in large operational spaces // Quantum Information. 2019. V. 5. P. 4.

14. Перри Р. Элементарное введение в квантовые вычисления: Учеб. пособие / Пер. с англ. — Долгопрудный: Изд. Дом "Интеллект", 2015.

15. Shor P. Algorithms for Quantum Computation: Discrete Logarithms and Factoring // Foundations of Computer Science, 1994 Proceedings, 35th Annual Symposium on — IEEE. 1994. P. 124—134.

16. Правильщиков П. А. Квантовая революция в вычислительной технике и новые модели вычислений // Информационные технологии в проектировании и производстве. 2020. № 1. С. 27—35.

17. IEEE Spectrum, IBM [Электронный ресурс]. URL: <https://spectrum.ieee.org/ibm-quantum-computer-ospreyfalse> (дата обращения: 14.11.2022)

18. Roth J. P. Diagnosis of automata failures: a calculus and method // IBM J. Research and Development. 1966. № 7. P. 18—32.

19. Карабский В. В., Пархоменко П. П., Согомо-нян Е. С., Халчев В. Ф. Основы технической диагностики. Кн. 1. — М.: "Энергия", 1976.

20. Правильщиков П. А. Новая квантовая математика: матричное исчисление кубических комплексов и квантовые D-алгоритмы // Информационные технологии в проектировании и производстве. 2017. № 2. С. 21—32.

21. Правильщиков П. А. Закон сохранения переноса и естественный параллелизм D-алгоритмов для построения тестов и моделирования в технической диагностике // Автоматика и телемеханика. 2004. № 7. С. 156—199.

22. Нильсен М., Чанг И. Квантовые вычисления и квантовая информация / Пер. с англ. — М.: Мир, 2006.

23. Антес Г. Перспективы субатомных ИТ // Computerworld. 2006. № 37. С. 43—47.

24. Бортаковский А. С., Пантелеев А. В. Линейная алгебра в примерах и задачах. — М.: Высш. шк., 2010.

25. Дойч Д. Структура реальности. (The Fabric of Reality) — РХД — Москва—Ижевск, 2001.

26. Прескилл Дж. Квантовая информация и квантовые вычисления. — Москва—Ижевск: ИКИ (НИИ Регулярная и хаотическая динамика), 2008. Т. 1.

27. Правильщиков П. А. Новая квантовая логика: новые однородные и неоднородные квантовые логические элементы // Информационные технологии в проектировании и производстве. 2019. № 1. С. 42—51.

28. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. — М.: Мир, 1982. — 416 с.

A QUANTUM APPROACH TO SOLVING PROBLEMS FORMAL VERIFICATION OF DIGITAL CIRCUITS

P. A. Pravilshchikov

V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences (RAS), Moscow, Russia

The reasons for the return to logical control systems based on hard logic are outlined. A description of quantum algorithms is given on examples of solving the problem of verification for logical control systems based on rigid logic. It is assumed that quantum algorithms operate on the platform of a problem-oriented quantum computer with a quantum parallelism mechanism. To avoid unnecessary details, instead of a quantum computer, we consider here the solution of verification problems based on its new computational model.

Keywords: quantum computers, quantum algorithms, formal verification, qubits, qutrits, ququadrts, qunits.

Правильщиков Павел Алексеевич, ведущий научный сотрудник.
E-mail: pravilshchikov@mail.ru

Статья поступила в редакцию 24 ноября 2022 г.

УДК 004.896

DOI: 10.52190/2073-2597_2023_1_21

EDN: SYITWI

СПОСОБ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ АВТОМАТИЗИРОВАННЫХ СИСТЕМ ТЕХНОЛОГИЧЕСКОЙ ПОДГОТОВКИ ПРОИЗВОДСТВА

Е. Ю. Поляков; Д. В. Кондусов, канд. техн. наук; В. Б. Кондусова, д-р техн. наук
ФГБОУ ВО «Оренбургский государственный университет», Оренбург, Россия

Рассмотрены основные стадии технологической подготовки производства с учётом реальных производственных условий. Представлен сравнительный анализ решений, позволяющих многократно использовать ранее накопленные знания. Предложен вариант онтологии предметной области "Конструкторско-технологическая подготовка производства".

Ключевые слова: автоматизированная система технологической подготовки производства, онтологическая база знаний, технологическая документация, технологический процесс, САПР.

В текущих геополитических условиях, и в соответствии с Указом Президента Российской Федерации от 30 марта 2022 г. № 166 «О мерах по обеспечению технологической независимости и безопасности критической информационной инфраструктуры Российской Федерации» особое внимание привлечено к решениям, направленным на интеллектуализацию программных средств. На машиностроительных предприятиях на протяжении жизненного цикла изделия, применяются соответствующие системы автоматизации, повышение эффективности которых является актуальной задачей. Применение баз знаний, сформированных из конструкторско-технологической документации на основе онтологии представляется перспективным способом решения.

Анализ вариантов развития процессов автоматизации технологической подготовки производства

Важным этапом жизненного цикла изделий является технологическая подготовка производства (ТПП). Качество конечного продукта, сроки его выхода на рынок и конкурентоспособность предприятия в целом определяется уровнем автоматизации ТПП [1].

Рассмотрим четыре основные стадии ТПП в реальных производственных условиях:

1. *Проработка технологической документации (ТД).* Перед тем как приступить к разработке технологического процесса (ТП) документация проходит предварительную проработку, в которой задействовано большинство служб предприятия. Чертежи и их анализ на технологичность проходят технический контроль. Определяются объёмы работ, сроки, экономическая эффективность, обеспеченность материальными, трудовыми и финансовыми ресурсами. Предварительная разработка производится без учёта типизации ТД и на основе опыта сотрудников задействованных бюро.

2. *Разработка межцеховых технологических маршрутов.* ТПП начинается с определения межцеховых технологических маршрутов, в которых указана последовательность обработки заготовок, деталей и сборочных единиц по производственным подразделениям. До сих пор в большинстве случаев вышеперечисленные действия на предприятиях выполняются на бумажном носителе в виде планов и графиков, которые регламентируют прохождение изделия по подразделениям и никак не отражают маршрутное или маршрутно-операционное описание технологии изготовления изделия посредством сквозного ТП. Каждое подразделение описывает свою часть изготовления в отдельном технологическом процессе, что усложняет задачу использования типовых решений, накопленных за долгие годы работы предприятия.

3. *Разработка операционных процессов.* Схема и порядок разработки ТП установлен на базе типовых решений и процессов с использованием индивидуальных технологических решений.

4. *Отработка и запуск в производство технологического процесса.* Для запуска в производство технологического процесса необходимо в соответствии с ГОСТ 3.1109-82 создать и отработать весь комплект документов ТП. В настоящее время технологические процессы на предприятиях разрабатываются преимущественно с использованием специализированных САПР, таких как «Спрут-ТП», «Вертикаль», ADEM CAPP, T-FLEX Технология и др.

В процессе проектной деятельности на предприятиях накапливаются знания в виде конструкторской и технологической документации. Использование имеющихся данных из архивных документов, учёт специфики производства и выполнение бизнес-процессов ТД на предприятии осуществляется как во время проработки, так и во время запуска изделия в производство. В связи с этим, актуальной является интеллектуализация САПР для автоматизации отдельных стадий технологической подготовки произ-

водства. Однако процесс автоматизации ТПП имеет ряд трудностей, связанных с потребностью поддержания должного уровня качества выпускаемой продукции. Анализ возможных проблем, возникающих при увеличении объёмов производства и оказывающих влияние на автоматизацию ТПП, представлен в таблице.

Согласно таблице автоматизация любого типа подготовки производства первое время будет препятствовать производственному процессу. При увеличении объёма производства на предприятии любое принятое решение может привести к замедлению производственного процесса. Для минимизации возникновения осложнений требуется повышение эффективности подготовки производства с использованием решений, позволяющих многократно применять ранее накопленные знания.

Обзор публикаций

Рассмотрим наиболее значимые с точки зрения авторов данной статьи результаты работ российских исследователей, опубликованные за последние двадцать лет и направленные на изучение возможностей использования ранее

накопленных знаний в автоматизированных системах производства.

Куликов Д. Д. [2] подробно описывает концепцию развития автоматизированных систем ТПП (АСТПП) и предлагает использовать единое информационное пространство (ЕИП) с использованием трёх уровней иерархии знаний в онтологии АСТПП для организации эффективного взаимодействия между подсистемами ТПП с последующей поставкой информации в автоматизированную систему управления предприятием.

Овдей О. М. и Проскудина Г. Ю. [3] рассматривают применение PDM-системы для создания онтологии ТПП на примере выбора инструментальных средств путём организации систем управления знаниями и использованием редакторов онтологии, позволяющих редактировать и анализировать онтологии различных уровней знаний.

Авторы работ [2, 4] уделяют внимание одной из главных задач при разработке и внедрении АСТПП — организации ЕИП с использованием графа знаний, обеспечивающего совместную согласованную работу конструкторов, технологов и других инженерных специалистов, участвующих в процессах подготовки производства.

Перечень вариантов развития и возможных последствий при увеличении объёмов производства

№	Вариант развития	Используемый метод	Возможные недостатки	Возможные последствия
1	Увеличение штата сотрудников	Наставничество, сопровождение новых сотрудников квалифицированными специалистами с опытом работы на предприятии	Отсутствие понимания специфики работ. Затраты времени на обучение. Допущенные ошибки, обусловленные отсутствием наставничества при высокой загрузке производства.	Замедление производственного процесса ввиду высокой загрузки квалифицированных специалистов. Снижение качества выпускаемой продукции, брак. Текущность персонала.
2	Увеличение объёмов ТПП	Автоматизация ТПП. Увеличение штата сотрудников.	Отсутствие специализированных автоматизированных систем для ТПП. Отсутствие должного количества квалифицированных специалистов.	Замедление производственного процесса из-за отвлечения трудовых и временных затрат на автоматизацию ТПП. Возможные последствия из п.1.
3	Автоматизация ТПП	Использование универсальных CAD/ CAE/ CAPP/ CAM систем, интегрированных между собой и с ERP-системой. Своевременное пополнение и поддержание в актуальном состоянии баз данных. Увеличение штата сотрудников.	Отсутствие у руководства понимания конечной цели автоматизации. Увеличение трудовых и временных затрат на автоматизацию ТПП. Использование САПР, не учитывающих реальные производственные условия. Отсутствие квалифицированного персонала. Повышенный объём работ на начальных стадиях внедрения автоматизации.	Большие денежные затраты на закупку программного обеспечения, лицензий, технической поддержки и оплату внедренческой деятельности. Возможные последствия из п.1.

Кондусов Д. В., Кондусова В. Б., Сергеев А. И., Сердюк А. И. [5—8] описали влияние использования накопленных инженерных знаний на стадиях жизненного цикла изделия и предложили применять конструкторские знания для поиска 3D-моделей в имеющихся на предприятиях архивах с последующей их индексацией или с последующей индексацией знаний.

Гаврилова Т. А., Тузовский А. Ф., Чириков С. В., Ямпольский В. З. [9, 10] описывают онтологический подход к проектированию информационно-коммуникационных систем, в которых ранее используемые знания становятся доступными для большинства сотрудников.

Наместниковым А. М. [11] разработан онтологический подход, модели, методы и средства которого представляют собой теоретическую основу для анализа слабоструктурированных ресурсов проектной организации на начальных этапах проектирования сложных автоматизированных систем, использующих нечеткие логические формализмы при формировании контекстно-ориентированных запросов к архивам технических документов.

Гуськов Г. Ю., Бениаминов Е. М. [12, 13] посвятили свою работу описанию знаний с последующим их применением в области онтологического проектирования с использованием онтологии хранения знаний. Авторы [12, 13] предложили онтологическую модель шаблона проектирования, построенную на основе объектных свойств и свойств типов данных как для серверных, так и клиентских частей автоматизированных систем проектирования программно-аппаратных комплексов.

Работы [12, 14] описывают создание интеллектуальных систем для решения проблем поиска форм представления данных с помощью формата описания нечеткой онтологии [15]. Взяв за основу увеличение онтологической визуализации знаний, авторы данных работ рассматривают возможность решения инженерных задач с использованием аппарата нечеткой логики.

Колошкина И. Е. [16] описывает комплектность ТД согласно ЕСТД, приводит примеры автоматизации маршрутно-операционного описания ТП с использованием типовых технологических решений.

Куликов Д. Д., Падун Б. С., Яблочников Е. И. [17] отмечают сложность подготовки, согласования, нормоконтроля и утверждения большого комплекса документов с часто возникающей необходимостью возврата на предыду-

щие этапы проекта для исправления неверных или неэффективных решений и затрагивают проблему автоматизации бизнес-процессов технологических проектов.

Пример схемы онтологии технологических знаний АСТПП

Анализ вышеперечисленных исследований позволяет сделать вывод о том, что в качестве одного из способов повышения эффективности ТПП можно рассматривать применение АСТПП на основе онтологической базы технологических знаний.

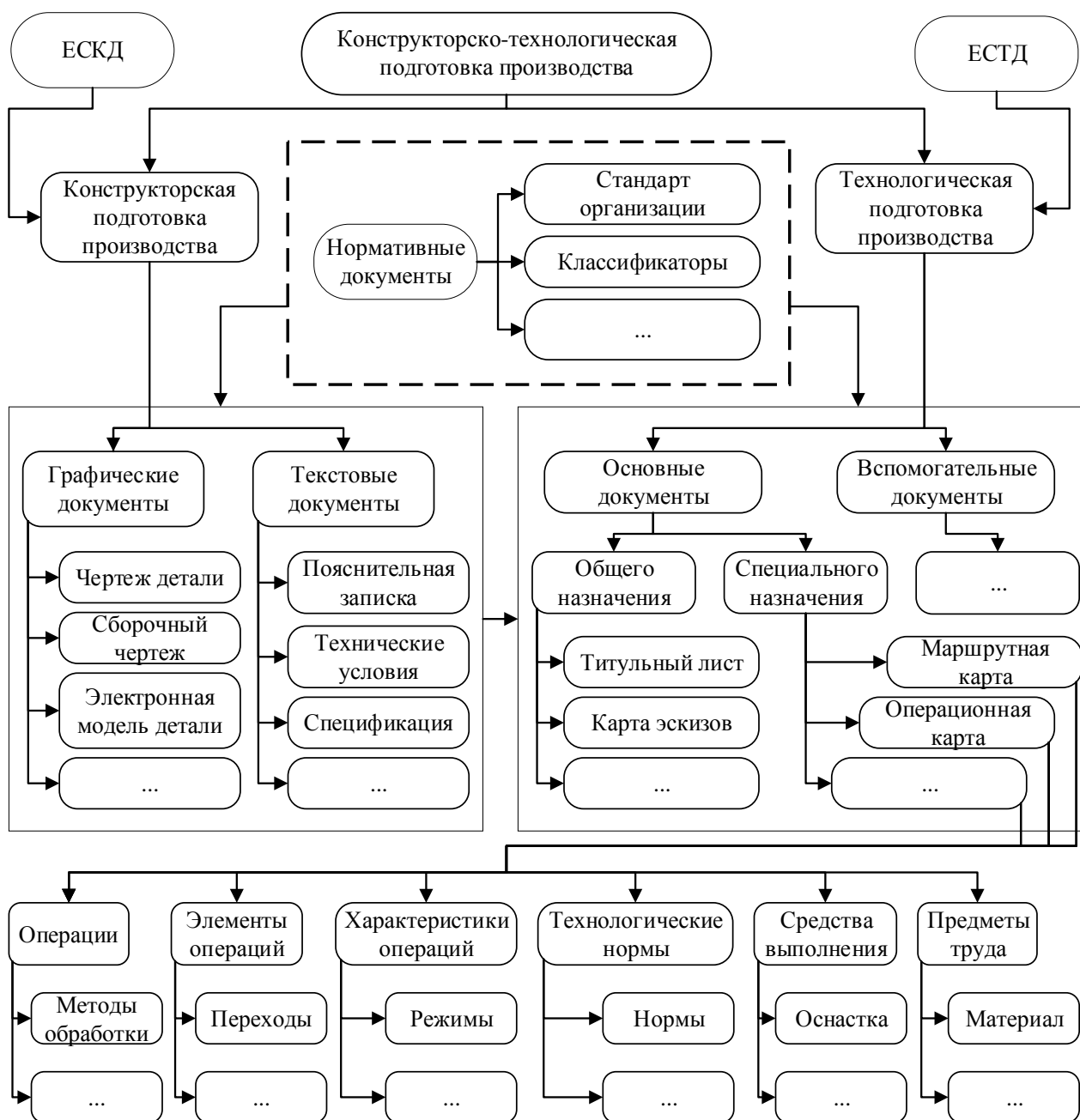
Разрабатываемая АСТПП должна:

- включать в себя онтологии различного уровня;
- содержать конструкторско-технологическую и архивную документацию;
- быть интегрирована в ЕИП предприятия.

В онтологии АСТПП выделяют три уровня иерархии знаний [18]. Верхний уровень – онтология науки, включающая научную деятельность и научные знания. Средний уровень – онтология предметной области, включающая классы понятий и отношений. На нижнем уровне находится онтология предметных знаний, которая включает экземпляры классов понятий и отношений.

В АСТПП используются множество словарей разных типов и структур. Словарь (онтология языка документов) – это система языковых средств выражения онтологии предметной области. Лингвистическая информация представлена в словаре функциональными группами лексических единиц, выраженных классов понятий и набора дополнительных атрибутов, отражающих специфику выражений [2]. Для включения ТПП в ЕИП необходимо привести структуру словарей к единому виду.

С учётом вышеперечисленных требований авторами разработана онтология предметной области «Конструкторско-технологическая подготовка производства», использующая онтологию среднего, нижнего уровня и основные концепты, применяемые в ТПП, которая укрупненно приведена на рис.1. По данной онтологии предметной области прослеживается зависимость разработки технологической документации не только от ЕСТД, но и от ЕСКД, конструкторской подготовки производства и нормативных документов с последующим переходом на онтологию предметных знаний в части основных технологических документов.



Обобщенная схема уровней онтологий базы технологических знаний

Предложенная онтология предоставляет возможность структурированно извлекать знания и их взаимосвязи из технологической документации в целях формирования базы технологических знаний. Интеграция такой базы в АСТПП для последующего использования на любой стадии ТПП позволит повысить эффективность АСТПП.

Заключение

Применение в АСТПП базы знаний, сформированной из конструкторско-технологической документации и построенной на онтологии предметной области «Конструкторско-технологическая подготовка производства» поз-

волит решить проблемы автоматизации основных этапов ТПП, минимизирует негативное влияние на текущий производственный процесс, предоставит возможности сотрудникам предприятия (в том числе вновь принятым, с отсутствием понимания специфики работ) использовать реально существующие предметные знания ТПП, что в совокупности способствует повышению эффективности АСТПП.

Литература

1. Митрофанов С. П., Куликов Д. Д., Милеев О. Н., Падун Б. С. Технологическая подготовка гибких производственных систем. – Л.: Машиностроение, 1987.

2. Куликов Д. Д. Управление знаниями в автоматизированной системе технологической подготовки производства // Известия высших учебных заведений - Приборостроение. 2014. С. 41-45.
3. Овдей О. М., Проскудина Г. Ю. Обзор инструментов инженерии онтологий // Электронные библиотеки. – М.: Институт развития информационного общества. 2004. Т. 7. Вып. 4.
4. Яблочников Е. И. Методологические основы построения АСПИ. – СПб: СПбГУ ИТМО, 2005.
5. Кондусов Д. В. Методика автоматизированного проектирования изделий машиностроения на основе многократного использования конструкторских знаний: специальность 05.13.12 «Системы автоматизации проектирования (машиностроение)»: Автореферат диссертации на соискание учёной степени кандидата технических наук. – Оренбург: Оренбургский государственный университет. 2020.
6. Кондусов Д. В., Кондусова В. Б., Сергеев А. И., Сердюк А. И. Влияние накопленных инженерных знаний на значение коэффициента готовности наукоемких изделий // Автоматизация в промышленности. 2020. № 2. С. 53–56.
7. Кондусов Д. В., Кондусова В. Б. Интеллектуализация автоматизированного проектирования с использованием накопленных инженерных знаний // Вестник машиностроения. 2019. № 5. С. 61–63.
8. Кондусова В. Б. Автоматизация разработки технологической документации // Информационные технологии в проектировании и производстве. 2020. №2. С. 16–20.
9. Гаврилова Т. А. Онтологический подход к управлению знаниями при разработке корпоративных систем автоматизации // Новости искусственного интеллекта. 2003. №2.
10. Тузовский А. Ф., Чириков С. В., Ямпольский В. З. Системы управления знаниями (методы и технологии). – Томск: НТЛ, 2005.
11. Наместников А. М. Интеллектуальные репозитории технической документации в проектировании автоматизированных систем: специальность 05.13.12 «Системы автоматизации проектирования (промышленность)»: Автореферат на соискание доктора технических наук. – Ульяновск: Ульяновский государственный технический университет, 2018.
12. Гуськов Г. Ю. Методы и средства формирования предметных онтологий в автоматизированном проектировании программно-аппаратных комплексов: специальность 05.13.12 «Системы автоматизации проектирования (промышленность)»: Автореферат диссертации на соискание ученой степени кандидата технических наук. – Ульяновск: Ульяновский государственный технический университет, 2018.
13. Бениаминов Е. М. Некоторые проблемы широкого внедрения онтологий в ИТ и направление их развития // Труды Симпозиума "Онтологическое моделирование". – М.: ИПИ РАН 2008. С. 71–82.
14. Голенков В. В. Онтологическое проектирование интеллектуальных систем // Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2017): материалы VII Междунар. научн. техн. конф. – Минск: БГУИР, 2017. С 37–56.
15. Bobillo F., Straccia U. Fuzzy Ontology Representation using OWL 2 // International Journal of Approximate Reasoning. Vol. 52. No. 7. 2010. P. 1073–1094.
16. Клошкина И. Е. Автоматизация разработки технологической документации // Информационные технологии в проектировании и производстве. 2019. № 1. С. 56–61.
17. Куликов Д. Д., Падун Б. С., Яблочников Е. И. Перспективы автоматизации технологической подготовки производства // Известия высших учебных заведений - Приборостроение. 2014. С. 7–11.
18. Боровикова О. И., Булгаков С. В., Сидорова Е. А. Система знаний информационного интернет-портала по научной тематике // Молодая информатика: Сб. тр. аспирантов и молодых ученых. – Новосибирск, 2005. С. 11–19.

A WAY TO INCREASE THE EFFICIENCY OF AUTOMATED SYSTEMS TECHNOLOGICAL PREPARATION OF PRODUCTION

E. Yu. Polyakov, D. V. Kondusov, V. B. Kondusova

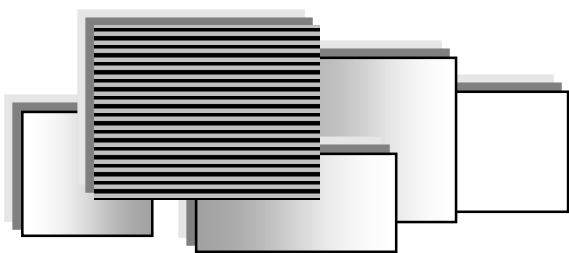
Federal State Budgetary Educational Institution of Higher Education "Orenburg State University",
Orenburg, Russia

The main stages of technological preparation of production are studied taking into account real production conditions. A comparative analysis of the repeated use of engineering knowledge is made. The ways of increasing the efficiency of automated systems of technological preparation of production.

Keywords: automated system of technological preparation of production, ontological knowledge base, technological documentation, technological process, CAPP- systems.

Поляков Евгений Юрьевич, старший преподаватель.
E-mail: P_Jenka@mail.ru
Кондусов Дмитрий Викторович, доцент.
E-mail: kdimka@list.ru
Кондусова Валентина Борисовна, доцент.
E-mail: valyosha@list.ru

Статья поступила в редакцию 15 марта 2023 г.



ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В РЕШЕНИИ ЗАДАЧ УПРАВЛЕНИЯ

УДК 681.5:004.75

DOI: 10.52190/2073-2597_2023_1_26

EDN: TGM1WD

ПОДХОД К РАСШИРЕНИЮ ВОЗМОЖНОСТЕЙ МОНИТОРИНГА ФУНКЦИОНАЛЬНОГО СОСТОЯНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ В УСЛОВИЯХ НЕПРЕРЫВНЫХ ИЗМЕНЕНИЙ

И. В. Логинов, канд. техн. наук
Академия ФСО России, г. Орёл, Россия

Сформулирована проблема мониторинга функциональных возможностей непрерывно изменяющихся автоматизированных систем. Решение проблемы предполагает пересмотр границ объекта управления и дополнительный учет параметров источников требований и ресурсов в процессе мониторинга функциональной пригодности. Снижение уровня неопределенности информации достигается за счет реконфигурации системы мониторинга на основе минимума ресурсоемкости и ввода дополнительных данных из журналов событий и текстов нормативных документов с использованием положений теории нечетких множеств.

Ключевые слова: автоматизированная система, мониторинг, изменение назначения, функциональные возможности, реконфигурация, неопределенность.

Усложнение состава функций автоматизированных систем (АС) приводит к значимому увеличению их структурно-функциональной сложности. Современные автоматизированные системы управления (АСУ) — например, АСУ предоставлением услуг связи на основе модели *TMN*, интегрированные АСУ предприятиями — *MES*, *ERP*, *PLM*-системы, АСУ безопасностью типа "умный город", — представляют собой распределенные многофункциональные (МФ) АС, имеющие десятки взаимодействующих функциональных компонентов (ФК).

Непрерывное изменение состава и содержания функций МФ АС связано с их модернизацией и реконфигурацией. Наблюдающееся увеличение интенсивности изменения внешних условий и параметров функционирования АС из-за влияния рынка и регуляторов требует развертывания системы мониторинга, которая позволяет предоставлять информацию для решения задач реконфигурации и эволюции [1]. Цель такой системы мониторинга — определение изменения требований назначения, задач пользователей, условий применения, а также определение соответствия функций АС изменившимся требованиям. Технической основой

системы мониторинга являются средства выявления изменения в требованиях для формирования новой системы требований и модель функциональной пригодности АС для оценивания соответствия возможностей АС изменившимся требованиям. Таким образом, усложнение функций автоматизированных систем, стохастический поток заявок на реконфигурацию и модернизацию средств автоматизации, неопределенность изменения внешних условий и процессов функционирования обостряет актуальность решения задачи совершенствования функциональности мониторинга.

Обзор подходов к мониторингу автоматизированных систем

Как уже было отмечено, современные автоматизированные системы развиваются в направлении расширения своей функциональности. Например, АСУ предприятием связи, построенная на основе модели *TMN*, автоматизирует около 300 технологических и управленческих процессов. Примерами автоматизированных процессов являются распределение и

поставка ресурсов, конфигурирование и активация ресурсов, их тестирование. Подсистему, автоматизирующую отдельный технологический процесс, рассматривают как функциональный компонент АС. Совокупность действий АС, направленная на достижение выполнения технологического процесса, представляет собой функцию АС.

Управление МФ АС требует проведения мониторинга функционирования в режиме реального времени, обработки ошибок, обнаружения и устранения неисправностей и функциональных деградаций системы [2]. Из-за большого масштаба и сложности АС, а также появления новых требований к функциям, возникновение неисправностей и сбоев в работе АС является обычным явлением. Это обуславливает наличие потребности в механизмах для обнаружения аномальных состояний. Известные подходы к мониторингу функциональной пригодности включают *расширенный мониторинг требований назначения и состояния МФ АС, проактивный и прогностический мониторинг*. Их применение позволяет перейти к устойчивому управлению развитием. Стохастический характер функционирования МФ АС [3] показывает необходимость применения механизмов наблюдения за состоянием объекта в условиях неопределенности. Отмечается тесная связь результативности мониторинга и эффективности реализации управления как функциональной пригодностью АС, так и технической исправностью [4, 5]. Основными требованиями к процессу мониторинга МФ АС являются: своевременность, эффективность, высокая степень устранения неопределенности данных, удовлетворение заказчика, соответствие процесса требованиям [6].

Управление реконфигурацией в АС реализуется на основе *моделей управляемого объекта*. Развитие моделей в системах управления связано с конфигурационным управлением оборудованием. Реализация такого управления требует совершенствования моделей управляемого объекта. В инфокоммуникационной подсистеме АС широко используют *SNMP*-подобные технологии, которые позволяют собирать первичные данные об объектах управления и аккумулировать их в *MIB* базе [7]. Усложнение объекта управления обуславливает переход к *многоуровневым перестраиваемым моделям*, например *автоматным* [8], требующим интеграции механизмов синтеза, сопровождения и поддержки сложных систем в процессе использования. Снижение нечеткости мониторинга достигается за счет использования подходов на основе положений теории нечетких множеств. Например, в [9] представлены механизмы мониторинга за состояниями сетевых элементов телекоммуникационной составляющей АС с использованием нечетких множеств в рамках концепции *TMN* при неполноте знаний об объектах мониторинга.

ГОСТ Р ИСО 19439-2022 вводит понятие "представление модели предприятия" [10]. В соответствии с данным ГОСТом, для описания систем (предприятий) используют функциональное, информационное, ресурсное и организационное представления. Мониторинг АС может быть выполнен по этим представлениям, и в его процессе должны быть получены данные по изменению функционального назначения АС, характеристик выполняемых автоматизированных процессов, параметров процессов обработки и передачи информации (информационных процессов), характеристик потоков и запасов ресурсов, параметров состояния технических средств автоматизации.

Отдельно следует отметить важность извлечения информации об изменении функциональных требований. Основными факторами динамики функциональных требований являются: изменение внешних условий (требований рынка и регуляторов), изменение автоматизированного процесса или предприятия в целом (организационно-штатные, нормативные изменения), переосмысление процессов автоматизации (изменение технологических процессов, границ многофункциональной системы), уточнение требований (переосмысление требований и их уточнение, приведение в соответствие с реальным положением дел), изменение возможности и ограничения по автоматизации деятельности [11–13].

Проблема мониторинга функциональной пригодности многофункциональных автоматизированных систем

В качестве объекта мониторинга рассмотрим многофункциональную автоматизированную систему S , состоящую из множества функциональных компонентов автоматизации s_i . Незначительное влияние автоматизации одних технологических процессов на другие позволяет пренебречь взаимовлиянием ФК и рассмотреть МФ АС в виде множества $S = \{s_i\}$. Работа МФ АС опишем моделью \hat{S} .

Изменения внешней среды Y и потребностей в автоматизации $N \subset Y$ приводят к несоответствию модели \hat{S} объекту S , где относительная ошибка оценивания объекта ε_n может быть определена как:

$$\varepsilon_n = \frac{\hat{S} - S}{S} = \sum_{i=1}^N \alpha_i \frac{|\hat{s}_i - s_i|}{s_i}, \quad (1)$$

где N — число функциональных компонентов АС;

s_i, \hat{s}_i — i -й ФК и его модель соответственно;

α_i — показатель значимости ФК.

В соответствии с [14] и при допущении, что аппроксимация распределения характеристик наблюдаемой МФ АС является экспоненциальной, а характеристики МФ АС при периодическом мониторинге независимы, вероятность наличия актуальной информации $P_{\text{акт}}$ об объекте S может быть определена выражением:

$$P_{\text{акт}} = k_{\text{н}} \frac{\xi^2}{q(\xi + t_{\text{обр}})} \left[1 - e^{-\frac{q}{\xi}} \right], \quad (2)$$

где ξ — среднее время между значимыми изменениями функций МФ АС;
 q — период мониторинга МФ АС;
 $t_{\text{обр}}$ — среднее время обработки информации, полученной в результате мониторинга;
 $k_{\text{н}} = 1 - \varepsilon_{\text{н}}$ — коэффициент, характеризующий изменение вероятности наличия актуальной информации из-за ошибки оценивания.

При применении известных подходов к мониторингу в рамках внешнего проектирования систем возникает проблема, связанная с тем, что увеличение интенсивности изменений МФ АС и требований к ней приводит к "снижению адекватности" модели \hat{S} (то есть к уменьшению вероятности наличия актуальной информации об объекте S) из-за сокращения ξ и увеличения ошибки оценивания $\varepsilon_{\text{н}}$ (рис. 1).

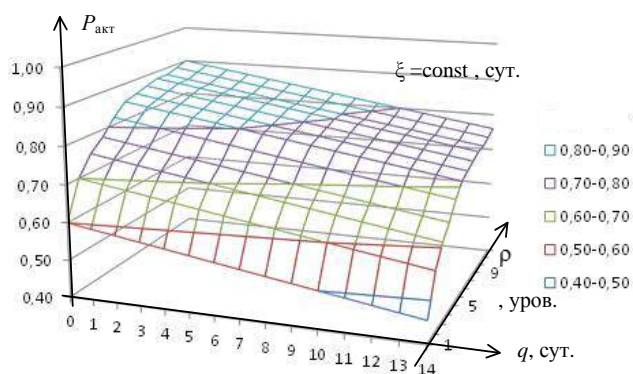


Рис. 1. График зависимости вероятности наличия актуальной информации о МФ АС от периода и глубины ее мониторинга

На величину $\varepsilon_{\text{н}}$ оказывает влияние глубина мониторинга ρ — число функциональных компонентов АС, которые подвергаются мониторингу. Источником проблемы несоответствия возможностей системы мониторинга требованиям по управлению функциональными возможностями непрерывно изменяющихся автоматизированных систем является разнесение контуров выявления изменения требований (в рамках внешнего проектирования системы) и мониторинга состояния функций (в рамках оперативного

управления) МФ АС. Решение проблемы может быть достигнуто за счет совершенствования механизмов мониторинга изменения функций МФ АС [2].

Увеличение уровня неопределенности состояния многофункциональной автоматизированной системы при ее усложнении

Усложнение МФ АС, вызванное ростом количества ее функций, и увеличение доли требований к АС, изменившихся в единицу времени, приводят к повышению уровня неопределенности состояния автоматизированной системы. Если мониторинг технических компонентов и процессов их функционирования реализуется средствами измерения, то характеристики процессов автоматизации и требования к их функциональным возможностям определяют на основе экспертных оценок. Неопределенность может быть лингвистической (неопределенность смысла фраз и нечеткость оценок) и физической (неточность вследствие несовершенства измерительных средств и случайности наблюдаемых параметров из-за стохастической природы процессов функционирования) [15]. На практике в принятии решений зачастую используется недостоверная (включая преднамеренное искажение), неточная, неполная (полученная в результате частичного наблюдения) и противоречивая информация [16].

При рассмотрении модели АС в виде показателей значимости и функциональной пригодности составляющих ее функциональных компонентов $s_i = \langle k_i, \alpha_i \rangle$ наблюдаются шумы, вызванные неопределенностью значимости ФК ε_{α} , неопределенностью пригодности ФК ε_k и неполнотой критериев управления ε_r (рис. 2).

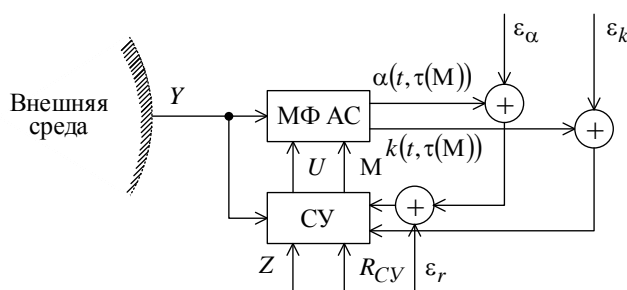


Рис. 2. Формальная схема мониторинга функциональной пригодности при неопределённости исходных данных о состоянии МФ АС

Снижение уровня неопределенности в известных исследованиях достигается посредством:

- выявления неявно заданных причинно-следственных связей на основе теории нечетких множеств и теории планирования экспериментов [16];

ции". Изменение процесса — операций и их последовательности — определяется путем восстановления процесса и расчета отклонений от известной структуры:

$$L \rightarrow \{\sigma_i\} \rightarrow \{g_i\} \rightarrow pr_L \subset FS = \{fs\}, \quad (5)$$

где $\{\sigma_i\}$ — последовательность записей в журнале;
 $\{g_i\}$ — вектор первичных признаков операций;
 pr_L — модель процесса;
 $FS = \{fs\}$ — функциональные характеристики МФ АС.

$$\text{Вектор } g_i = (v_i^1, \dots, v_i^n, e_{i1}, \dots, e_{im}, h_{i1}^1, \dots, h_{im}^r),$$

где v_i^1, \dots, v_i^n — статические признаки;
 e_{i1}, \dots, e_{im} — класс операции в позиции;
 $h_{i1}^1, \dots, h_{im}^r$ — динамический признак данной операции.

Автоматическое построение структуры процесса по журналам событий позволяет делать оценки хода его выполнения при наличии предварительно полученных оценок реализаций $w \in W$. Для этого применима предсказательная модель на основе обучающей выборки. Для решения задачи может использоваться метод оценки на основе леса решающих деревьев [18].

Извлечение данных о новых требованиях из вводимых в действие нормативных и регулирующих документов ND может быть реализовано на основе подхода *Legal Profile for Goal-oriented Requirements Language* [19]. В его рамках выполняется представление документа в форме семантической сети функциональных требований:

$$X \rightarrow \{ND\} \rightarrow \langle SemND \rangle \rightarrow QR = \langle \{qr\}, E \rangle : RS_{i-1} \xrightarrow{QR} RS_i, \quad (6)$$

где $SemND = \langle CNCP, E_{CNCP} \rangle$ — семантическая сеть взаимосвязанных E_{CNCP} концептов $CNCP$ нормативных требований QR ; $qr = \langle \langle Cnd, Sbj, Act, Obj \rangle, \Phi \rangle$, где $Cnd \in CNCP$ — условия (множество) реализации требования; $Sbj \in CNCP$ — субъект; $Act \in CNCP$ — действие с объектом; $Obj \in CNCP$ — объект, Φ — проверяемый критерий.

Применение нечетких оценок для снижения уровня неопределенности

Нечеткий характер измеряемых и оцениваемых явлений определяет использование положений теории нечетких множеств для описаний значений наблюдаемых параметров.

При непрерывном характере изменения случайных величин таких как, например, время выполнения проекта реконфигурации, предлагается использовать функции плотности вероятности $X = \phi(X)$, полученные на основе результатов наблюдений. Числовые показатели можно оценивать с помощью *методов интервальных оценок* $X = [\underline{X}, \bar{X}]$, *треугольных чисел* $X = \langle x_{\min}, x_{\text{mod}}, x_{\max} \rangle$ и *нечетких функций принадлежности* $X = \mu(x)$. Методы интервальных оценок могут применяться, например, для оценки трудоемкости операций, метод треугольных чисел — для определения ресурсоемкости проекта модернизации, а метод нечетких функций принадлежности — для оценки количества задач, выполняемых функциональным компонентом в плановом периоде.

Для учета неопределенности значений нечисловых показателей можно использовать *модели нечетких запросов* $X = \mu(H)$ и *колеблющиеся во времени нечеткие наборы лингвистических оценок* $X = H_s$: $H_s = \langle \{x_i, \dots, x_j\}, T \rangle$, где $x_i \in B = (H, T(H), U, G, M)$ — лингвистическая переменная, $T = \{1, 2, \dots, N\}$ — множество, определяющее время [20]. Колеблющиеся во времени нечеткие наборы лингвистических оценок отличаются возможностью задания изменяемых во времени оценок. В качестве примера таких оценок можно привести выражение: "Значение показателя надежности будет не хуже, чем *высоким* до 2025 года, а затем станет не хуже, чем *средним*". Здесь "высокий", "средний" — элементы шкалы оценивания (термы), "не хуже чем" — пример отношения (унарного или бинарного).

Перестраивание системы мониторинга в ходе эволюции автоматизированной системы

Как отмечалось выше, изменение многофункциональной автоматизированной системы требует изменения системы мониторинга. Однако наряду с тем, что в настоящее время потенциально доступны эффективные программно-аппаратные измерительные датчики для систем мониторинга, совокупная стоимость их приобретения и использования довольно высока.

Для обеспечения необходимой актуальности информации о функциональной пригодности АС с учетом ограничений на ресурсы мониторинга может быть применен алгоритм оптимизации сети мониторинга MN. Данный алгоритм определяет состав и порядок реконфигурации сенсорной сети программно-аппаратных датчиков $top \in MON$ для сбора информации об измене-

ниях S , внешней среды $y \in Y$ и потребностях в аналитической информации $n \in N$ на основе критерия минимума ресурсоемкости реконфигурации R при заданных ограничениях актуальности информации ξ :

$$MN = \arg \min \sum R(S, y, z); \xi \leq \xi^{\text{доп}}, \quad (7)$$

где $R(S, y, n)$ — ресурсы на изменение системы мониторинга MON одного ФК автоматизации. В составе ресурсов учитываются ресурсы на добавление новых датчиков, а также их функционирование в активном (с генерацией тестового сигнала) и/или пассивном режимах с функциями затрат $r^{<a>}$ и $r^{<p>}$ соответственно;

$$R(S, y, n) = \sum_{i=1}^n r_{c,i} + \sum_{i=1}^n \mu_i r_{\Phi,i} + k r_p, \quad (8)$$

где $r_{c,i}$ — стоимость приобретения датчика;
 μ_i — интенсивность использования;
 $r_{\Phi,i}$ — удельная ресурсоемкость использования $r_{\Phi} = \{r^{<a>}, r^{<p>}\}$;
 k — количество реконфигураций сети мониторинга;
 r_p — ресурсоемкость реконфигурации.

Каждый датчик при этом описывается кортежем вида $mon = \langle \{r_c, r_{\Phi}\}, \{k_{\text{акт}}, k_{\text{сн}}\} \rangle$, где $k_{\text{акт}}$ — коэффициент увеличения актуальности информации состояния МФ АС, $k_{\text{сн}}$ — коэффициент снижения уровня неопределенности.

Влияние совершенствования мониторинга на эффективность работы АС

Управление функциональным развитием осуществляется в условиях ограниченных ресурсов — реализовать все заявки на новые функции невозможно. Рассмотрим пример модернизации автоматизируемой системы со следующими характеристиками: $t_{\text{функ}} \in [50, 120]$, $n \in [30, 50]$. Неуправляемые характеристики системы управления: $\mu_N = 1/3$, $t_{\text{восс}} = 10$, $k^{\text{восс}} = 0,99$, $k^{\text{мин}} = 0,8$. Показатели времени заданы в условных единицах, в качестве которых могут выступать сутки, недели, месяцы.

Вероятность наличия актуальной информации $P_{\text{акт}}$ зависит от уровня неопределенности в измерении характеристик АС (рис. 4). При возможности выбора для реализации одного функционального компонента из двух средняя величина коэффициента функциональной пригодности K изменяется от $2/3$ до $1/2$ с уменьшением $P_{\text{акт}}$.

Пример изменения коэффициента функциональной пригодности K от количества функциональных компонентов и уровня неопределенности информации приведен на рис. 5.

Из рис. 5 видно, что при увеличении уровня неопределенности информации, измеряемой ошибкой оценивания ε_n , значительно снижается коэффициент пригодности. В условиях недостаточности ресурсов снижение уровня неопределенности информации за счет реконфигурации системы мониторинга и ввода дополнительных данных позволяет повысить коэффициент функциональной пригодности на 3—5 %.

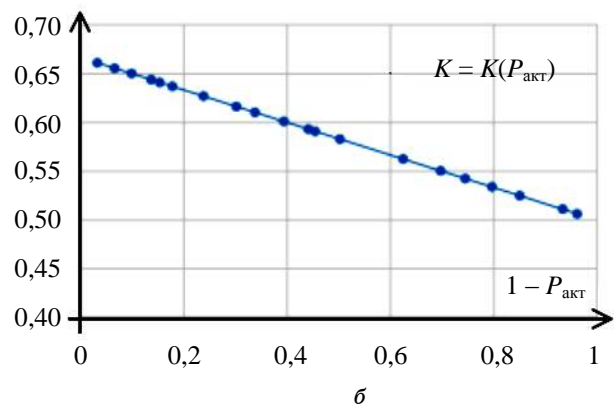
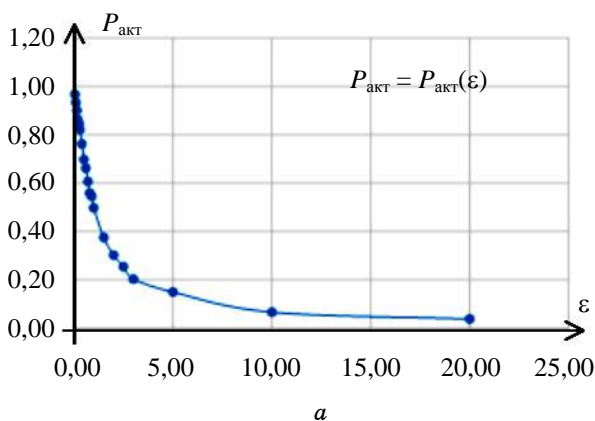


Рис. 4. График зависимости:

а — вероятности актуальности информации от величины ошибки;
 б — коэффициента функциональной пригодности от вероятности актуальности информации

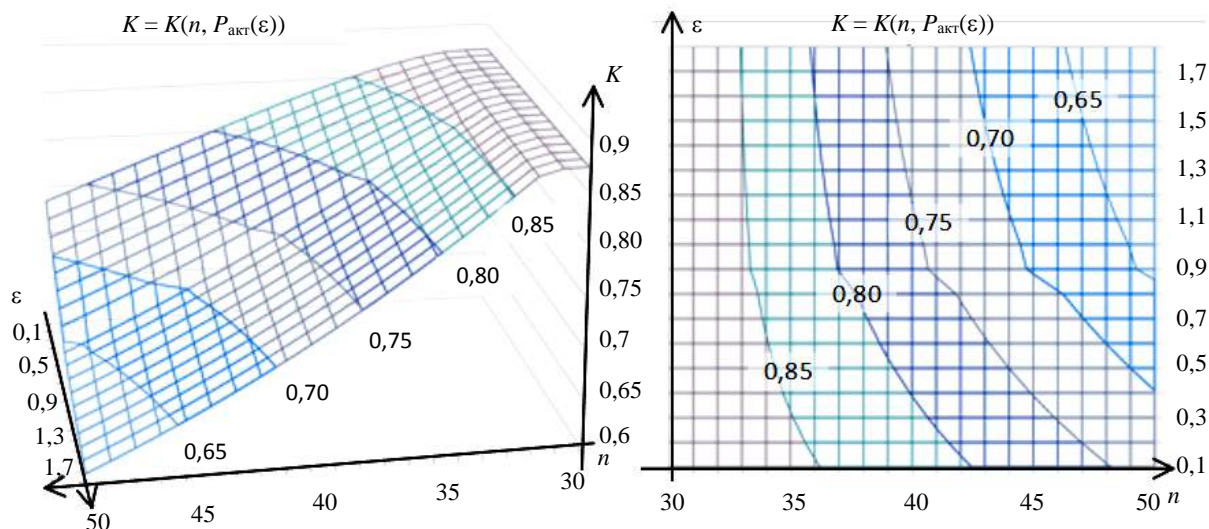


Рис. 5. Изменение коэффициента функциональной пригодности от количества функциональных компонентов и ошибки оценивания

Заключение

Рассмотрена проблема совершенствования механизмов мониторинга функциональной пригодности многофункциональных автоматизированных систем в условиях интенсивного изменения требований назначения АС. Решение задачи мониторинга АС в интересах управления функциональными возможностями требует пересмотра границ объекта управления и учет характеристик источников требований и ресурсов при мониторинге. Снижение уровня неопределенности информации и соответственно увеличения вероятности наличия актуальной информации об управляемой автоматизированной системе достигается за счет реконфигурации системы мониторинга на основе показателя минимума ресурсоемкости, ввода дополнительных данных на основе автоматизированной обработки журналов событий и текстов нормативных документов с помощью технологией *process-mining*, а также учетом неопределенности наблюдаемых параметров с использованием положений теории нечетких множеств. Предложенное решение позволило повысить эффективность работы многофункциональной автоматизированной системы в условиях недостаточных ресурсов.

Литература

1. Гришаков В. Г., Логинов И. В. Управление динамической реконфигурацией ИТ-инфраструктуры в меняющихся условиях // Информационные системы и технологии. 2016. № 3(95). С. 13—22.
2. Core Research and Innovation Areas in Cyber-Physical Systems of Systems (CPSoS). Initial Findings of the Support Action CPSoS, November 26, 2014.
3. Prasad Sistla A., Zefran M., Feng Z. "Runtime monitoring of stochastic cyber-physical systems with hybrid state" in RV, 2012.

4. Оркин В. В. Модель функционирования автоматизированной системы управления, учитывающая информацию мониторинга производительности в узлах сетевой инфраструктуры // Интеллектуальные технологии на транспорте. 2019. № 1(17). С. 13—18.

5. Губская О. А., Кривцов С. П., Захарченко А. С., Ксенофонов Д. А. Моделирование подсистемы мониторинга и управления автоматизированной системы управления сети связи специального назначения // Вопросы оборонной техники. Сер. 16: Технические средства противодействия терроризму. 2020. № 7–8(145–146). С. 25—29.

6. ГОСТ Р 58606-2019. Системная и программная инженерия. Процесс измерения. — М.: Стандартинформ, 2019.

7. Соколов С. А., Стокипный А. Л. Диагностическая модель для задач выявления закономерностей функционирования компонентов корпоративной IP-сети // Восточно-Европейский журнал передовых технологий. 2009. Т. 5. № 2(41). С. 4—8.

8. Жукова Н. А. Общая и частная задачи многоуровневого синтеза моделей объектов мониторинга // Научно-техническая информация. Сер. 2. Информационные процессы и системы. 2019. № 11. С. 16—22.

9. Бычков Е. Д. Математические модели управления состояниями цифровой телекоммуникационной сети с использованием теории нечетких множеств. — Омск: Изд-во ОмГТУ, 2010.

10. ГОСТ Р ИСО 19439-2022. Интеграция предприятия. Основа моделирования предприятия. — М.: Стандартинформ.

11. Jayatilake Sh., Lai R. A Systematic Review of Requirements Change Management // Information and Software Technology. 2018. V. 93. P. 163—185.

12. Dasanayake S., Aaramaa S., Markkula J., Oivo M. Impact of requirements volatility on software architecture: How do software teams keep up with ever changing requirements? // J. Software: Evolution and Process. 2019.

13. Chua B., Verner J. Examining Requirements Change Rework Effort: A Study // International J. Software Engineering & Applications. 2010. V. 1. Iss. 3. P. 48.

14. ГОСТ Р 58494-2019. Оборудование горношахтное. Многофункциональные системы безопасно-

сти угольных шахт. Система дистанционного контроля опасных производственных объектов. — М., 2020.

15. Куликов Г. Г., Антонов В. В., Антонов Д. В. Построение формальной модели предметной области с применением нечеткой кластеризации // Вестник УГАТУ. 2011. Т. 15. № 5. С. 3—11.

16. Ignatyev M., Marley V., Mikhailov V., Spesivtsev A. Multi-model approach to city governance in the face of uncertainty // Communications in Computer and Information Science. Digital Transformation and Global Society. First International Conference. 2016. P. 469—477.

17. Щербатов И. А. Управление сложными слабоформализуемыми многокомпонентными системами: монография. — Ростов на Дону: Изд-во ЮНЦ РАН, 2015.

18. Бударевский И. С., Гришаков В. Г., Логинов И. В. Обобщенный подход к автоматизации оценивания бизнес-процессов организации связи на основе технологий process-mining // Инфокоммуникационные технологии. 2017. Т. 15. № 4. С. 371—379.

19. Ghanavati S., Amyot D., Rifaut A. Legal goal-oriented requirement language (legal GRL) for modeling regulations // In Proceedings of the 6th International Workshop on Modeling in Software Engineering (MiSE 2014). Association for Computing Machinery. New York, USA. 2014. P. 1—6.

20. Rodriguez Rosa M., Martir'nez Luis, Herrera Francisco. Hesitant Fuzzy Linguistic Term Sets for Decision Making // IEEE Transactions On Fuzzy Systems. 2012. Vol. 20. № 1. P. 109—119.

AN APPROACH TO EXPANDING THE MONITORING CAPABILITIES OF THE FUNCTIONAL STATE OF AUTOMATION SYSTEMS IN CONDITIONS OF CONTINUOUS CHANGE

I. V. Loginov

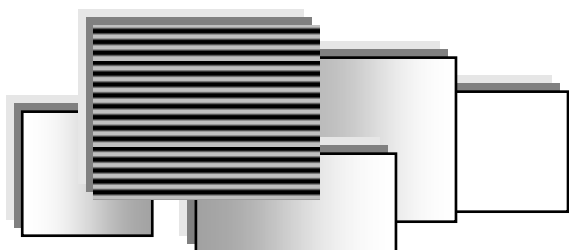
The Academy of the Federal Guard Service of the Russian Federation, Orel, Russia

The problem of monitoring the functionality of continuously changing automated systems is formulated. The solution to the problem involves revising the boundaries of the control object and additionally taking into account the parameters of the sources of requirements and resources in the process of monitoring functional suitability. Reducing the level of information uncertainty is achieved by reconfiguring the monitoring system based on a minimum of resource consumption and entering additional data from event logs and texts of regulatory documents using the provisions of fuzzy set theory.

Keywords: automation system, monitoring, change of purpose, functionality, reconfiguration, uncertainty.

Логинов Илья Валентинович, сотрудник.
E-mail: loginov_iv@bk.ru

Статья поступила в редакцию 12 января 2022 г.



СОВРЕМЕННЫЕ МЕТОДЫ И СРЕДСТВА РЕШЕНИЯ ЗАДАЧ ПРОЕКТИРОВАНИЯ И МОДЕЛИРОВАНИЯ. ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

УДК 654.1

DOI: 10.52190/2073-2597_2023_1_34

EDN: TJOTTG

АВТОМАТИЗИРОВАННАЯ СТАНЦИЯ ПРОВЕРКИ ИСПОЛНЕНИЯ АБОНЕНТСКОГО ПРОТОКОЛА СПЕЦИАЛИЗИРОВАННОЙ КОММУНИКАЦИОННОЙ СЕТИ

В. М. Соболев

АО «Ордена Трудового Красного Знамени научно-исследовательский институт
автоматической аппаратуры им. академика В. С. Семенихина», Москва, Россия

Рассмотрен автоматизированный имитационный аппаратно-программный комплекс контроля корректности реализации и точности исполнения и интерпретации абонентского протокола документального обмена, являющегося основой семейства телекоммуникационных комплексов, базирующихся на специализированной транспортной сети передачи данных. Проведение автономной и комплексной отладки с использованием интерпретатора предусматривает задание сценария проверок в фиксированной форме воспроизведения трафика документального обмена. Для подготовки проверки в фиксированной форме предложен соответствующий формальный язык задания сценария основных режимов функционирования. Отмечено, что для нескольких поколений коммуникационных комплексов интерпретатор стал неотъемлемым атрибутом предварительных, межведомственных, государственных испытаний.

Ключевые слова: абонентская служба, абонентский протокол, телекоммуникационный комплекс, детерминированный конечный автомат, система обмена данными, документальный обмен, виртуальный терминал, формальный язык задания сценариев.

Документальный обмен, осуществляемый по специализированной сети связи в форме автоматизированной системы документального обмена (АСДО), является основным видом обмена текстовой информацией для ряда ведомств и государственных организаций, обладающий соответствующим юридическим статусом [1–3]. Крупномасштабная сеть объединяет сотни взаимодействующих абонентов, подключенных к телекоммуникационным комплексам (ТКК), связанных между собой магистральными каналами и образующих АСДО. Вместе с другими современными средствами связи реально существующая защищенная АСДО также служит для надежного выполнения функций резервной системы в силу её меньшей совокупной уязвимости. Поддержание постоянной готовности составной части действующей АСДО необходимо для обеспечения эффективного управления как в мирное время, так и в экстремальный период.

В основе функционирования каждого ТКК находится единый абонентский протокол (АП) документального обмена специализированной коммуникационной сети, являющийся доминантным стержнем для целого семейства разных поколений ТКК [4]. В частности, системный подход использования унифицированного АП обеспечил при создании АСДО алгоритмическую переносимость основных алгоритмов, функциональных программ, а также сохранение возможности взаимодействия абонентов обширного терминального парка и абонентских каналов при различных видах модификации ТКК.

Вместе с тем, при любом переносе программ на новую информационную платформу и усовершенствованные аппаратные средства, а также при устранении обнаруженных ошибок неизбежной становится модификация, оптимизация, расширение номенклатуры предоставляемых услуг, что естественно влечет за собой

необходимость тщательной повторной проверки корректности исполнения всех без исключения положений АП. Учитывая сложность и трудоемкость упомянутых проверок, задача автоматизации контроля программной реализации и качества её исполнения является весьма актуальной, требующей для сопровождения ТКК обязательного наличия соответствующего технологического инструментария. Поскольку, по экспертным прогнозам, существующая АСДО еще надолго сохранит свое значение в системе оперативного управления, актуальность совершенствования указанных средств поддержки объективно возрастает.

При использовании рассматриваемых средств поддержки приходится не только ограничиваться автоматизацией проведения контрольного эксперимента, но и осуществлять принципиально новый подход — выполнение автоматического анализа полученного результата, который субъективно оценивался ранее только вручную соответствующими должностными лицами. Предлагаемая работа посвящена именно этой задаче автоматического анализа, решение которой позволяет однозначно оценивать результат контроля правильности реализации АП по формализованному сценарию проверки.

Постановка задачи

Формально функциональная реализация всех регламентов и положений АП канонически может быть представлена абстрактной моделью конечного детерминированного программного автомата (ДПА) [5].

Парадигма детерминированности предопределена тем, что функционирование и все состояния ТКК, реализующего языковые положения АП $\supset L_{АП}^{ВВ} \cup L_{АП}^{ВЫВ}$, полностью регламентированы и предсказуемы с учетом любых внешних воздействий, осуществляемых через удаленные терминалы операторами-абонентами АСДО на формальном языке путем использования входных лексем $L_{АП}^{ВВ}$ и результирующими значениями выводимых служебных и текстовых сообщений $L_{АП}^{ВЫВ}$. На рис. 1 представлена схема организации передачи сообщения в соответствии с АП, где $L_{АП}^{ВВ} = (\text{Введен заголовок}, \text{Введен текст}, \text{КП} — \text{конец передачи}, \text{Подтверждение готовности}, \text{подтверждение сообщения})$, $L_{АП}^{ВЫВ} = (\text{Квитанция за заголовок}, \text{Квитанция за текст}, \text{Запрос готовности}, \text{Вывод сообщения}, \text{таймер "подтверди"} — \text{подтверждено или НПТ} — \text{не подтверждено})$.

На рис. 2 представлен общий вид реализаций состояний ДПА в процессе передачи сообщения $(s_0, s_{зг}, \{s_{тхт}\}, s_{кп})$, организации режима прямых переговоров $(s_0, s_{уст}, \{s_{прг}\}, s_{кп})$, подтвержде-

ния принятого сообщения $(s_0, s_{прн})$, подтверждения готовности $(s_0, s_{гтв})$, тестирования абонентского канала $(s_0, s_{тст})$ после выполнения которых ДПА возвращается в исходное состояние s_0 . Нижние индексы идентифицируют следующие состояния $S_{ДПА} \supset \{s_{зг}, s_{тхт}, s_{уст}, s_{кп}, s_{прн}, s_{гтв}, s_{тст}\}$ где:

- $s_{зг}$ — ввод заголовка сообщения;
- $s_{тхт}$ — ввод текста сообщения;
- $s_{кп}$ — ввод команды конца передачи;
- $s_{уст}$ — установка виртуального соединения партнеров;
- $s_{прг}$ — ввод текста переговоров (ведение переговоров);
- $s_{прн}$ — ввод подтверждения за принятое сообщение;
- $s_{гтв}$ — ввод команды запроса готовности;
- $s_{тст}$ — ввод команды тестирования абонентского канала.

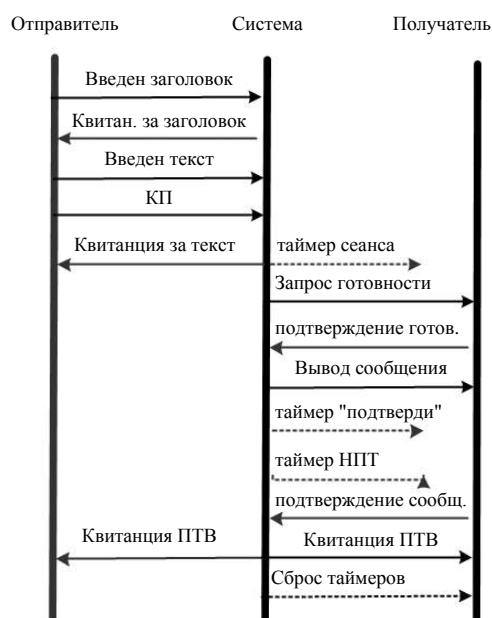


Рис. 1. Передача сообщения

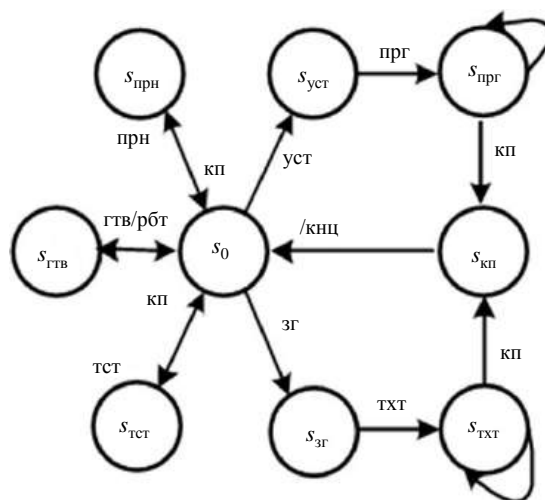


Рис. 2. Состояния детерминированного программного автомата

Детерминированность АП естественным образом влечет методологическую возможность однозначного составления минимизированного ряда сценариев проверки его функционирования, независимо от способов технической реализации ТКК, охватывающих всю конечную номенклатуру внешних входных воздействий в виде входных языковых лексем $L_{АП}^{вв}$ на ДПА.

На базе АП по известным данным [6], построены, внедрены, модернизированы и практически почти полувекна находятся в повседневной эксплуатации следующие версии ТКК:

- центр коммутации сообщений;
- стационарный и мобильный концентратор терминальной нагрузки;
- комплекс документального обмена;
- автоматизированный коммутационный центр.

Каждый из указанных ТКК после завершения отладочных работ прошел доскональные проверки на предварительных, межведомственных и государственных испытаниях, в том числе с использованием официально утвержденных методик сценариев проверки корректности реализации АП.

Очевидно, что при таком длительном использовании столь сложных изделий как ТКК, они на протяжении нескольких лет также неоднократно подвергались обновлению, модификации, совершенствованию по разным причинам. Прежде всего это устранение допущенных программных ошибок, оптимизация ряда алгоритмов наиболее трудоемких подзадач, а также расширение, добавление новых функций. Так как программный продукт ТКК относится к классу встроенных систем [7], создаваемых из множества взаимосвязанных модулей, понятно, что любая даже незначительная модификация требует обязательной перепроверки работоспособности всего ТКК, потому, что всегда остается вероятность случайного привнесения новой ошибки и прямого или косвенного нарушения даже хорошо ранее отлаженных функций.

Так, тридцать лет находившиеся в повседневной эксплуатации на многочисленных объектах серийно изготовленные концентраторы терминальной нагрузки сменили 47 версий программного обеспечения, а автоматизированные коммутационные центры глобальной АСДО — более 10 версий. При любой модификации версий программного обеспечения в обязательном порядке на стенде опытного участка АСДО предварительно производила перепроверку функционирования ТКК, как правило, с использованием методик с соответствующими сценариями проверки корректности реализации АП.

В связи с перманентно повторяющейся потребностью использования соответствующего инструментария проверки и сопровождения ТКК, уже на ранней стадии проектирования был поставлен вопрос о создании автоматизиро-

ванной станции проверки (АСП) на базе динамического имитационно-испытательного комплекса для верификации корректности реализации АП, в задачи которой входила бы автоматизированная проверка всех без исключения его положений в стационарном режиме [8—12]. Основные принципы и методы использования АСП предусматривали обеспечение ускоренных испытаний на корректность и надежность программной реализации АП. В частности, экземпляр АСП, установленный на стенде опытного участка, эмулирует все ручные технологические операции человека-оператора $L_{АП}^{вв}$, выступая в роли множества виртуальных абонентов АСДО — асинхронно взаимодействующих "интеллектуальных" виртуальных терминальных установок.

Как отмечалось, основная (статическая) концепция методики проверки корректности функционирования программ, реализующих АП, основывается на программной интерпретации всех положений протокола как детерминированного конечного автомата, подлежащего тестированию, при котором известны и контролируются каждая комбинация исходных данных $L_{АП}^{вв}$ и соответствующие ей заранее прогнозируемые однозначно предсказуемые результаты исполнения программы в виде выводимых лексем $L_{АП}^{выв}$.

Доктринальным аспектом рассматриваемой методологии тестирования по формально заданному сценарию, является автоматизация анализа результата эксперимента. В АП в соответствии с абстракциями ДПА в зависимости от текущего состояния $s_i \in S_{ДПА}$ комплекса ТКК однозначно определено, какой должна быть ответная реакция функции выходов системы в виде лексемы $L_{АП}^{выв}$ (принятые сообщения, квитанции, служебные посылки и т. п.) на входные воздействия $L_{АП}^{вв}$ (вводимые сообщения, команды и запросы сервиса). Поэтому любые отклонения от АП достаточно элементарно распознаются путем сопоставления с соответствующим константным шаблоном, заложенным в банке шаблонов.

Номенклатура и методология проверок

В АП однозначно определены все информационные объекты и порядок их использования, часть из которых представляют собой:

- формальный (командный) язык абонента $L_{АП}^{вв}$, используемый в технологическом процессе передачи и приема сообщения и организации прямых переговоров;
- последовательность корректных действий абонента при выполнении технологического процесса передачи отправляемого сообщения и

подтверждения принятого сообщения с конкретным указанием в лексемах $L_{АП}^{вв}$ и $L_{АП}^{выв}$ соответствующих атрибутов обслуживаемого сообщения;

- перечень услуг АСДО, запрашиваемых абонентом непосредственно со своего терминала: итоговой справки о переданных и принятых сообщениях, неподтвержденных сообщениях, ждущих подтверждения принятых и отправленных сообщений, случаях несанкционированного доступа;

- формат заголовков выводимых сообщений $L_{АП}^{выв}$ с указанием заданных атрибутов сообщения, адресов отправителя и получателя, регистрационного номера сообщения, номеров страниц текста сообщения, указания общего количества страниц текста и автоматической привязки ко времени выдачи текущей информации;

- форматы отправляемого и принимаемого сообщений;

- допустимые параметры значений категории срочности, грифа сообщения, варианта передачи, объемов сообщений и временных значений сеанса сообщения.

Входы $L_{АП}^{вв}$ и выходы $L_{АП}^{выв}$ ДПА типизированы — входные и выходные типы сопоставлены конечному множеству запросов и команд, определяющих характер обработки и разнообразие данных воспринимаемых/выдаваемых по соответствующему входу/выходу.

Обобщенно каждый информационный обмен документальным сообщением в ТКК определен двумя фразами с указанием конца передачи:

1) **передача** ::= <згл><текст> <кп> — выполняет отправитель сообщения;

2) **подтверждение** ::= <при> <кп> — выполняет получатель сообщения.

Вместе с тем, каждое передаваемое сообщение характеризуется параметрами, которые необходимо учитывать при реализации АСП:

- категорией срочности (обыкновенное, срочное и т. д.);
- грифом сообщения (открытое, закрытое и т. д.);
- вариантом передачи (одноадресное, многоадресное, циркулярное, прямые дуплексные переговоры);
- объемом сообщения (нулевое, максимальное и т. д.);
- регистром текста сообщения (русский, латинский, цифровой);
- адресностью (локальное, транзитное);
- наличием страничной организации (одностраничное, многостраничное).

Подтверждение сообщений может быть выполнено автоматически путем задания соответствующей опции, вручную с кратким дополнительным текстом или без текста, а также с автоматическим формированием групповой результирующей квитанции за подтверждение.

Одна из наиболее важных и трудоемких форм методологических проверок состоит в организации совпадающих по времени одномоментных приемов сообщений разной категории срочности, независимо отсылаемых несколькими отправителями в адрес одного и того же получателя. В результате одновременного приема приоритетные сообщения временно прерывают вывод сообщений с низкой категорией срочности с последующим упорядоченным их восстановлением и выводом с признаком "ПОВТОР" в заголовке сообщения. Теоретически допустимо несколько последовательных прерываний, число которых также учитывается.

В качестве примера далее в таблице представлен неформализованный фрагмент методики выполнения задания проверки прерывания при квазиодновременном приеме на виртуальную терминальную установку двух одноадресных односторонних сообщений разной категории срочности, посланных двумя разными отправителями.

В результате спонтанности обменов между абонентами сети во время приема сообщения низкой категории срочности (C_n) или ведения переговоров низкой категории срочности ($П_n$) при одновременном поступлении более приоритетного сообщения (C_v) или переговора ($П_v$) может возникать потенциально конфликтная ситуация, приводящая к прерыванию ζ . Все комбинации прерываний $C_n \rightarrow C_n \zeta C_v$, $C_n \rightarrow C_n \zeta П_v$, $П_n \rightarrow П_n \zeta П_v$, $П_n \rightarrow П_n \zeta C_v$ приходится отдельно учитывать в методиках проверки, поскольку их алгоритмическая обработка происходит поразному — сообщения C_n доводятся до абонента, а переговоры $П_n$ сбрасываются. Наглядным показателем и критерием сложности реализации АП может служить систематически оформленная статическая часть методики проверки комплекса программ ТКК, основная целевая задача которых — обеспечивать своевременное и эффективное доведение сообщений. При этом в процессе тестирования следует учитывать задание предельного объема сообщений для различных категорий срочности, допустимого числа страниц сообщения, индивидуального выполнения запросов оператора на повтор принятого сообщения, различные варианты обслуживания переговоров, синтаксические ошибки оператора в лексемах заголовка, функционирования на различных скоростях и различных состояниях абонентских трактов и т.п.

Проверка прерывания одноадресного сообщения

	Выполнить последовательность действий	Результат
	С ВТУ ₁ на ВТУ ₂ послать сообщение:	Вывод на ВТУ ₁ :
1	Ввести заголовок сообщения: КС ₁ =обыкновенная, ГС ₁ =открытое, ВП ₁ =одноадресное, АПС ₁ = адрес ВТУ ₂ ;	Квитанция за заголовок отправляемого сообщения (РНС ₁ ... КС ₁ ... ГС ₁ ... ВП ₁ ...АПС ₁ ...)
2	Ввести ТС=текст сообщения, Объем текста порядка 1000 знаков;	
3	Ввести знак конца передачи	Квитанция за ввод текста — СП "РНС ₁ ... ЧМС..." //"КП"
		Вывод на ВТУ ₂ : - СП "ПОДТВЕРДИ ГТВ" - заголовок принятого сообщения (РНС ₁ ... КС ₁ ... ГС ₁ ...ВП ₁ ...АОС ₁ = адрес ВТУ ₁ ...) - фрагмент текста сообщения
	С ВТУ ₃ на ВТУ ₂ послать сообщение: Ввести заголовок сообщения	Вывод на ВТУ ₃
4	КС ₂ =срочное, ГС ₂ =открытое, ВП ₂ =одноадресное; АПС ₂ = адрес ВТУ ₂	Квитанция за заголовок отправляемого сообщения (РНС ₂ ... КС ₂ ... ГС ₂ ... ВП ₂ ...АОС ₂ = адрес ВТУ ₃ ...)
5	ввести текст сообщения, объем текста порядка 20 знаков	
6	ввести знак конца передачи	Квитанция за ввод текста — "РНС ₂ ... ЧМС..." //"КП"
		Вывод на ВТУ ₂ : - "РНС ₁ ...ПРЕРВАН ПРЕОРИТЕТ" - заголовок принятого сообщения (РНС ₂ ... КС ₂ ... ГС ₂ ... ВП ₂ ...АОС ₂ = адрес ВТУ ₃ ...) - текст принятого сообщения РНС ₂ -квитанция за вывод текста — "РНС ₂ ... ЧМС..." //"КП" - "ПОДТВЕРДИ РНС ₂ ..." - заголовок принятого сообщения (РНС ₁ ... КС ₁ ... ГС ₁ ... ВП ₁ ...АОС ₂ = адрес ВТУ ₁ ... "ПОВТОР 1") - текст принятого сообщения РНС ₁ -квитанция за вывод текста — "РНС ₁ ... ЧМС..." //"КП" - "ПОДТВЕРДИ РНС ₁ ..."
7	Подтвердить на ВТУ ₂ сообщение без сопроводительного текста. ПРН РНС ₂ АОС ₂	"РНС ₂ ПОДТВЕРЖДЕН"
8	Подтвердить на ВТУ ₂ сообщение без сопроводительного текста. ПРН РНС ₁ АОС ₁	"РНС ₁ ПОДТВЕРЖДЕН"
		- вывод на ВТУ ₃ результирующей квитанции "РНС ₂ ... ПТВ" - вывод на ВТУ ₁ результирующей квитанции "РНС ₁ ... ПТВ"

Примечания: ВТУ — виртуальная терминальная установка; КС — категория срочности сообщения; ГС — гриф сообщения; ВП — вариант передачи; ТС — текст сообщения; АОС — абонент — отправитель сообщения; АПС — абонент-получатель сообщения; РНС — регистрационный номер сообщения; ЧМС — часы, минуты, секунды; ПРН — команда подтверждения принятого сообщения; ПТВ — квитанция подтверждения принятого сообщения; СП — служебная посылка.

Управление заданиями проверки абонентского протокола

Фиксированную форму планирования обмена используют для жесткой формализации модельных экспериментов, предназначенных для детальной оценки правильности реализации конкретного пункта абонентского протокола документального обмена. Использование фиксированной стратегии планирования особенно продуктивно с учетом необходимости многократно повторять одни и те же сценарии проверок при комплексной отладке и различных видах испытаний. Эту стратегию применяют также в интересах отладки при многократном воспроизведении исключительных коллизий, однозначная повторяемость которых при стохастической стратегии затруднена или практически невозможна в регламентированный срок времени. Наконец, общеизвестная практика разработки и/или использования программных

продуктов свидетельствует, что изменения, направленные на улучшения программного обеспечения, могут привести к искажению работы отдельных ветвей его сложного алгоритма. Поэтому для очередной сертификации качества модифицированной программы необходима повторная её проверка, которую целесообразно проводить опять же, используя набор представительных сценариев фиксированной стратегии.

Важным достоинством рассматриваемого метода тестирования по формально заданному сценарию является автоматизация анализа результата $L_{АП}^{выб}$ проводимого эксперимента, так как в протоколе АП в соответствии с канонами детерминированного конечного автомата однозначно определено, какой должна быть ответная реакция системы на входные воздействия $L_{АП}^{вв}$, и любые отклонения от протокола тривиально распознаются.

Упрощенная схема проверок в АСП состоит из следующих блоков (рис. 3):

- блок меню выбора сценария;
- банк сценариев и программных даталогических шаблонов лексем ввода $L_{АП}^{вв}$ и вывода результатов $L_{АП}^{выв}$;
- интерпретатор сценариев и компаратор результатов;
- узел виртуальных абонентов $V_1...V_n$ и канального сопряжения АСП с ТКК по стандартному стыку С2 ТЛГ-телеграфной установки.

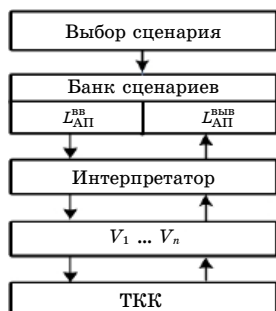


Рис. 3. Схема организации проверок абонентского протокола

Выбор сценария проверок производится путем указания номера методики в директории меню. На рис. 4 представлен упрощенный граф каталога меню.

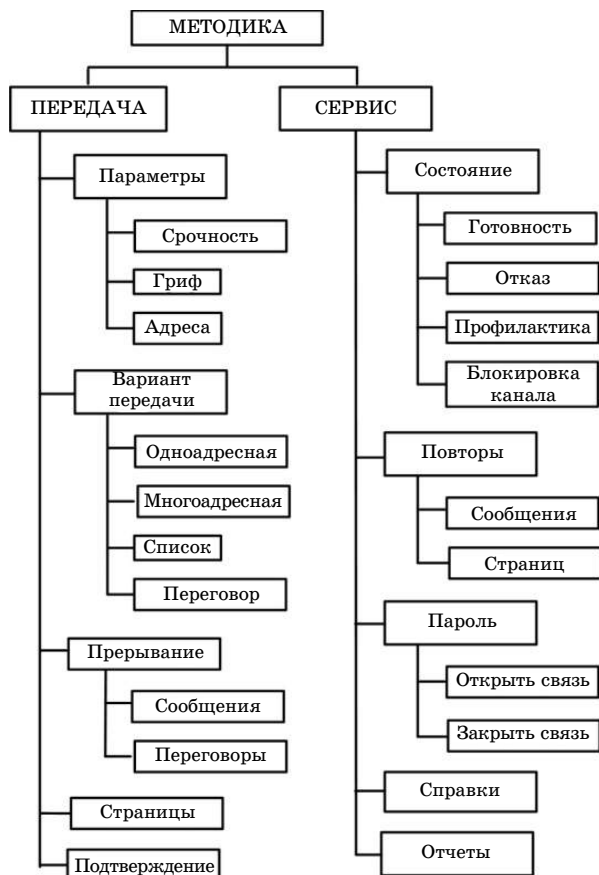


Рис. 4. Граф каталога методик проверки абонентского протокола

В нем предусмотрены основные формы проверок, связанные с разнообразными видами передачи сообщений и проверок исполнения всех сервисных услуг, регламентированных АП. Следует заметить, что в АСП помимо однократного тестирования заданной функции, регламентированной в АП, предусмотрен также режим многократного повтора её проверки с требуемым темпом цикличности.

Упрощенно формальный язык задания сценария проверок документального обмена в фиксированной форме состоит из следующих основных Б/Н-определений [13]:

```

набор_сценариев ::= сценарий эксперимента
{, сценарий эксперимента}
сценарий эксперимента ::= (передача_сообщений | прямые_переговоры)
передача_сообщений ::= передача сообщения {, передача сообщения}
прямые_переговоры ::= переговоры {, переговоры}
передача сообщения ::= <список_получателей> = <отправитель>
список_получателей ::= <адрес>(<автоответ>, <подтверждение>)
{, <адрес>(<автоответ>, <подтверждение>)}
адрес ::= цифра {цифра}k-1
автоответ ::= да | нет
подтверждение ::= 0÷100%
отправитель ::= <адрес>(<срочность>, <объем>, <пауза>)
срочность ::= цифра
объем ::= число
пауза ::= число | r // фиксированная или случайная задержка запуска сообщения
переговор ::= <абонент>==<инициатор>
абонент ::= <адрес>(<автоответ>)
инициатор ::= <адрес>(<срочность>, <объем>, <пауза>)
    
```

По заданному сценарию автоматически формируется проект ожидаемого результата обмена в виде набора из перечня служебных посылок (СП):

```

проект ::= <адрес> : <список_служебных_посылок>
список_служебных_посылок ::= <подтверди_готовность> | <заголовок_введен> | <сообщение_введено> | <заголовок_выведен> | <сообщение_выведено> | <подтверждено> | <неподтверждено> | <ПТВ> | <НПТ> | <прерван_приоритет> | <сброс_время> | <сброс_сбой> | <повтор> | <сброс_приоритет> | <переговор_установлен> | <переговор_закончен> | и т. п.
    
```

Для того чтобы продемонстрировать монотонность многократно повторяемого натурного эксперимента с учетом различных значений атрибутов, в качестве примера рассмотрим простейший фрагмент методики проверки прерывания сообщения *msgj*, с использованием трех

реальных терминалов T_i , $i = 1—3$ на опытном участке АСДО. Упрощенная схема его работы заключается в следующем (рис. 5):

- послать с терминала T_2 низкоприоритетное сообщение msg_n в адрес терминала T_1 ;
- при выводе на терминал T_1 некоторой части текста сообщения msg_n послать с терминала T_3 в адрес терминала T_1 высокоприоритетное сообщение msg_e ;
- убедиться, что на терминале T_1 последовательно произошло прерывание вывода сообщения msg_n с выдачей СП о прерывании (СП_{пер}), вывод сообщения msg_e , запрос на подтверждение сообщения msg_e , повторный вывод сообщения msg_n ;
- на терминале T_1 подтвердить принятое сообщение msg_e и убедиться, что на терминале T_3 получена СП с результирующим признаком подтверждения (СП_{пТВ});
- убедиться, что после завершения повторного вывода на терминале T_1 сообщения msg_n выдана СП_{пТВ} за ранее подтвержденное сообщение msg_e ;
- на терминале T_1 подтвердить принятое сообщение msg_n и убедиться, что на терминалы T_1 и T_2 выведены СП_{пТВ} за сообщение msg_n .

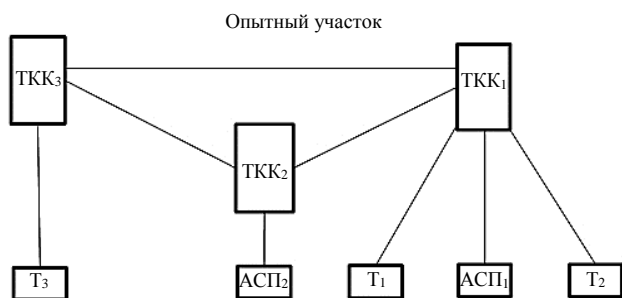


Рис. 5. Схема опытного участка АСДО

Сценарий передачи одноадресного сообщения, прерываемого более приоритетным сообщением с использованием виртуальных терминалов АСП, выглядит следующим образом:

$(A_1(a,p) = A_2(s_1, v_1, 0), A_1(a,p) = A_3(s_2, v_2, r)), s_2 > s_1$

A_2 : заголовок_введен, сообщение_введено, ПТВ

A_3 : заголовок_введен, сообщение_введено, ПТВ

A_1 : подтверди_готовность, заголовок_выведен, прерван_приоритет, заголовок_выведен, сообщение_выведено, подтверждено, заголовок_выведен, повтор, сообщение_выведено, подтверждено.

Для задания фиксированного сценария в АСП использовано диалоговое меню в виде визуального бланка из каталога методик проверки АП, результатом заполнения которого является

соответственно поименованный программный файл исполнения.

Для иллюстрации на рис. 6 представлен фрагмент каталога банка методик.

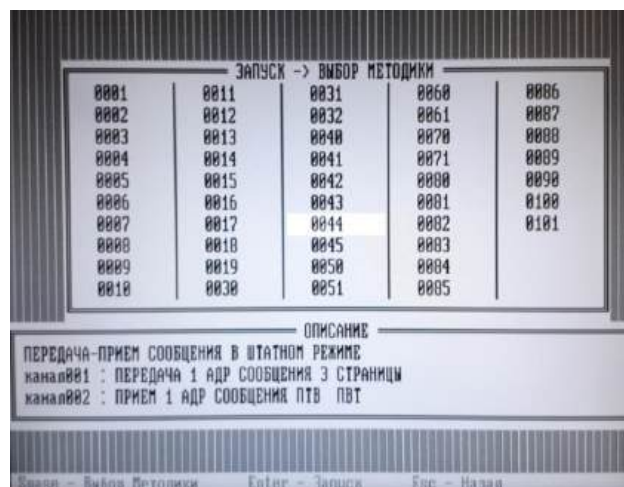


Рис. 6. Каталог банка методик

В общем случае при идентификации нескольких номеров в банке методик и использовании соответствующих команд функциональных клавиш обеспечивается групповое или циклическое исполнение сценария проверок АП.

Заключение

Изложенная методология с использованием АСП позволяет принципиально усовершенствовать технологию тестирования систем документального обмена с учетом его специфики и может быть рекомендована также для организации аналогичных проверок обширного класса комплексов средств автоматизации, что будет способствовать повышению полноты контроля и сокращения сроков их отладки.

Проведенные работы по применению АСП и полунатурные эксперименты показали возможность оптимизации проверок АП в полном объеме его регламентных положений. Интерактивные средства АСП обеспечивают возможность выбора требуемого режима проверок из подготовленной базы данных и предусматривают возможность оперативного создания дополнительных сценариев проверки. Апробация использования АСП позволяет также заключить, что предложенная методология на базе рассмотренной модели конечного детерминированного программного автомата достаточно адекватно отображает реальные процессы и механизмы в классе систем обработки документальных сообщений и способствует реализации ключевой парадигмы переносимости алгоритмов программ на ряд последующих поколений ТКК, перманентно совершенствуемой на обновленной элементной базе.

Автоматная концепция АП вносит ясность в понимание сложной структуры коммуникационной системы и ее функционирования, упрощает способ выполнения контроля технологических процессов документального обмена, способствует осмыслению методологии и реализации задач исследования взаимосвязей между совокупностью требований технической эксплуатации и результирующим качеством обслуживания пользователей АСДО.

Проведенное конструктивное программирование позволяет заключить, что результаты многолетнего развития методов форсированных испытаний АСДО на базе АСП позволили получить значительный выигрыш в сокращении длительности испытаний и затрат на их проведение.

Литература

1. Зацаринный А. А. Комплексный системный подход основа научно-технической политики Управления начальника связи ВС РФ // Тематический сборник, посвященный 85-й годовщине Войск связи. — М.: Наука, 2002. С. 95—108.
2. Зацаринный А. А., Коротаев В. Б., Иванов В. Н., Ионенков Ю. С. Сеть обмена данными как интегрирующая основа перспективной автоматизированной системы органов государственного управления // Качество и жизнь. Спец. выпуск. 2016. № 3(11). С. 16—18.
3. Соколов И. А., Оганян Г. А. Этапы формирования перспективной архитектуры АСУ ВС РФ // Вооружение России на рубеже веков. 2012. Т. 2. С. 20—31.
4. Коротаев В. Б., Машин В. П., Нестеров Б. И. Об одном подходе к построению информационной платформы проектирования коммуникационных средств документального обмена // Промышленные АСУ и контроллеры. 2018. № 4. С. 24—28.
5. Анисимов Н. А., Перчук В. Л. Представление протоколов обмена секвенциальными автоматами и сетями Петри // Техническая кибернетика. 1986. № 1. С. 74—80.
6. Институт военной связи. История и современность (1923—2008 гг.). — Мытищи, 1998.
7. Терехов А. Н. Технология программирования: учеб. пособие. Изд. 2-е. — М.: Интернет-Университет Информационных Технологий: БИНОМ. Лаборатория знаний, 2007.
8. Соболев В. М., Ударцев И. В. Имитатор абонентской нагрузки для тестирования и испытания коммуникационных комплексов специального назначения // Качество и жизнь. 2016. № 3. С. 41—45.
9. Рыжов А. С. Создание программного комплекса для анализа работоспособности устройств в составе автоматизированных систем управления: сб. трудов XIV Российской НТК "Новые информационные технологии в системах связи и управления". — Калуга, 2015. С. 79—83.
10. Лупаев В. В. Системное проектирование сложных программных средств для информационных систем. — М: Синтез, 2002.
11. Средства отладки больших систем / Под ред. Растина Р. — М.: "СТАТИСТИКА", 1977.
12. Соболев В. М., Ударцев И. В. Организация полунатурного тестирования и моделирования коммуникационных комплексов передачи сообщений специализированной сети документального обмена // Промышленные АСУ и контроллеры. 2018. № 5.
13. Соболев В. М. Обслуживание сообщений и переговоров в сети специального назначения // Промышленные АСУ и контроллеры. 2015. № 1. С. 62—70.

AUTOMATED STATION FOR CHECKING THE EXECUTION OF THE SUBSCRIBER PROTOCOL OF SPECIALIZED TELECOMMUNICATIONS

V. M. Sobol

Stock Company "Research Institute for Automated Apparatus
named after academician V. S. Semenikhin", Moscow, Russia

The article deals with an automated simulation hardware and software complex for monitoring the correctness of implementation and accuracy of execution and interpretation of the subscriber protocol of documentary exchange, which is the basis of a family of telecommunication complexes based on a specialized transport data transmission network. Autonomous and complex debugging using the interpreter involves setting a test script in a fixed form of playback of document exchange traffic. To prepare the test in a fixed form, an appropriate formal language for setting the scenario of the main modes of operation is proposed. It is noted that for several generations of communication complexes, the interpreter has become an integral attribute of all types of tests: preliminary, interdepartmental, state.

Keywords: subscriber service, subscriber protocol, telecommunication complex, deterministic state machine, data exchange system, documentary exchange, virtual terminal, formal scripting language.

Соболев Вадим Маркович, консультант-программист.
E-mail: sobolvm@yandex.ru

Статья поступила в редакцию 21 декабря 2022 г.

УДК 536.25

DOI: 10.52190/2073-2597_2023_1_42

EDN: TQARBY

ПРОЕКТИРОВАНИЕ ОКОННЫХ СТЕКЛОПАКЕТОВ НА ОСНОВЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ ТЕПЛОМАССОПЕРЕНОСА С УЧЕТОМ КЛИМАТИЧЕСКОЙ ЗОНЫ ЭКСПЛУАТАЦИИ

¹ А. А. Хвостов, д-р техн. наук; ² В. В. Ткач, канд. техн. наук; ² А. А. Журавлев, канд. техн. наук;

¹ О. А. Семенихин, канд. техн. наук; ² Е. А. Татаренков, канд. техн. наук

¹ Воронежский государственный технический университет, г. Воронеж, Россия

² Военный учебно-научный центр Военно-воздушных сил «Военно-воздушная академия имени профессора Н. Е. Жуковского и Ю. А. Гагарина», г. Воронеж, Россия

Предложен подход к учёту климатических условий эксплуатации при проектировании оконных систем на основе расчёта термического сопротивления многокамерного оконного стеклопакета, его геометрических размеров и количества камер. Разработана математическая модель неизотермического ламинарного движения воздуха в пространстве между стёклами, учитывающая теплопередачу через стекло. Исследованы зависимости термического сопротивления от перепада температур на границах остекления относительно базовой температуры, расстояния между стеклами, высоты и положения разделительного стекла.

Ключевые слова: математическое моделирование, термогравитационная конвекция, оконные системы.

Повышение энергоэффективности зданий является одной из ключевых задач в условиях постоянного роста цен на энергоносители. Несмотря на то, что показатели энергоэффективности окон жилых и промышленных зданий приближаются к показателям стеновых конструкций, улучшение их теплоизоляционных характеристик по-прежнему остается актуальной задачей. Это обусловлено необходимостью при сохранении прочностных характеристик максимизировать термическое сопротивление оконной конструкции, с одной стороны, и максимизировать площадь остекления, которая определяет комфортность нахождения человека в помещении, с другой.

Передача тепла через оконную конструкцию определяется тремя механизмами теплопереноса: излучением, теплопроводностью и конвекцией [1]. Доля конвективной составляющей в полном тепловом потоке через оконную систему зависит от расстояния между стеклами, и эта зависимость носит нелинейный характер [2–4].

Подходы к проектированию оконных систем

Проектирование оконных систем осуществляется с помощью методик, основанных на ГОСТ 24866-2014 «Стеклопакеты клееные» [5]. При этом данные методики не учитывают диапазон колебаний температур окружающего воздуха при выборе толщины воздушной прослойки в камере стеклопакета, а также количества стеклопакетов. ГОСТ 24866-2014 допускает достаточно большой разброс значений расстояний между стеклами — для внутреннего остекления он составляет 8-36 мм, а для внешнего 6-36 мм. При этом в [4] исследована зависимость термического сопротивления от расстояния между

стеклами и показано её изменение от разницы температур снаружи и внутри помещения и, следовательно, от соответствующих климатических условий эксплуатации. В работах [6, 7] показано, что энергоэффективность оконной системы является функцией многих параметров, включая геометрию окна и число стеклопакетов при определенных температурных диапазонах. Учёт совокупности всех факторов, влияющих на энергоэффективность окна, возможен только при решении полной сопряжённой задачи тепло-массопереноса с учётом явления термоконвекции, геометрии расчётной области и температурного диапазона эксплуатации оконной системы.

Существует довольно большое количество систем автоматизированного проектирования (САПР), позволяющих с высокой точностью разрабатывать оконные конструкции с заданными характеристиками. Наиболее популярными САПР являются *WinDoPlan*, *Optima Win*, *Klaes*, *Окнограф* [8]. Функционал данных программ во многом схож, однако присутствуют значимые различия. *WinDoPlan* позволяет при проектировании в точном соответствии с действующими нормативами задать все основные потребительские свойства оконного блока, учитывая при этом реальные условия его эксплуатации — форму и материал стен, конфигурацию четверти, температуру наружного воздуха и температуру в помещении, ветровые и иные нагрузки [9]. Преимуществом *Optima Win* является простота и легкость в управлении. Немецкая программа *Klaes* подходит как небольшим, так и крупным компаниям — изготовителям. Программа *Окнограф* дает возможность создавать визуализированные проекты различных конструкций, в том числе необычные варианты. Она проста, удобна в пользовании и имеет ши-

рокий функционал. Некоторые программы, в частности *WiDoPlan*, позволяют визуализировать температурные поля в оконной и примыкающих конструкциях на основе заданных коэффициентов теплопередачи.

Однако при расчётах геометрических размеров окна в этих САПР не учитывается влияние климатических условий эксплуатации на совокупность процессов тепломассопереноса в заполненном воздухом или другим газом пространстве между стеклами и их взаимосвязь с конструкцией и размерами оконной системы. Так как среднегодовые температуры, определяющие характер процессов теплопереноса в камерах стеклопакета, в каждой климатической зоне могут существенно различаться, при проектировании окон желательно учитывать климатические особенности и обоснованно закладывать в конструкцию оконной системы расстояние между стеклами и число стеклопакетов, обеспечивающие заданное значение термического сопротивления.

Расчёт термического сопротивления стеклопакета

Рассмотрим оценку энергоэффективности многокамерного стеклопакета на основе математической модели тепломассопереноса в камерах стеклопакета и дадим обоснование рациональных геометрических размеров (высоты H , расстояния между стёклами h и положения разделительного стекла δ в многокамерном стеклопакете) в зависимости от температурного режима эксплуатации. Температурный режим задаётся базовой температурой воздуха t_{base} и отклонениями от неё Δt . В качестве оценки энергоэффективности принимается термическое сопротивление стеклопакета R_t .

Для расчёта R_t необходимо решить полную сопряжённую задачу тепломассопереноса с учётом явления термоконвекции, геометрии расчётной области и температурного диапазона эксплуатации оконной системы. Ранее в [4] была представлена математическая модель тепломассопереноса в воздухе в пространстве между стёклами, описывающая процесс теплопереноса за счёт механизмов конвекции и теплопроводности на основе уравнений сохранения массы, энергии и импульса для случая стационарного неизоэтермического ламинарного движения воздуха в поле действия силы тяжести.

Авторы данной работы применили математическую модель тепломассопереноса в воздухе в пространстве между стёклами из [4] для расчётов параметров многокамерного стеклопакета.

Модель была дополнена расчётными областями для стекла, где перенос тепловой энергии осуществляется только за счёт механизма теплопроводности. Ввиду того, что толщина стеклопакета много меньше его ширины, задача была сведена к двумерной и рассмотрена в

вертикальном медианном сечении, нормальном плоскости стёкол [10].

Расчётная схема для двухкамерного стеклопакета показана на рис. 1.

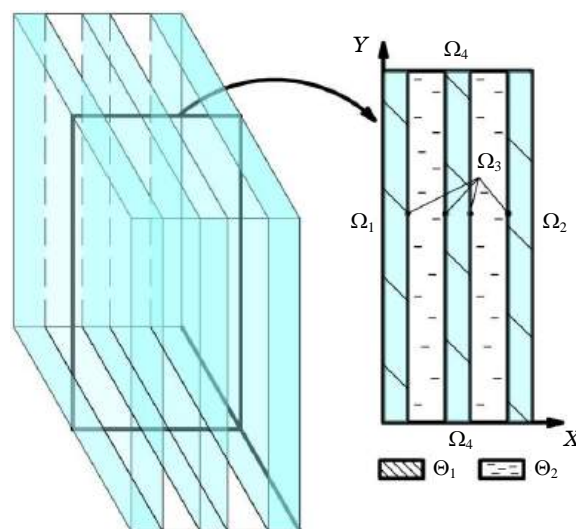


Рис. 1. Расчётная схема для двухкамерного стеклопакета

Стеклопакет представлен вертикальным сечением и содержит области Θ_1 , соответствующие стеклу, и Θ_2 , заполненные воздухом. На границах Ω_1, Ω_2 заданы граничные условия первого рода для «горячей» $t_h = t_{base} + \Delta t$ и «холодной» $t_c = t_{base} - \Delta t$ среды. На границах контакта воздуха и стекла Ω_3 заданы условия равенства температур и тепловых потоков. Верхняя и нижняя границы Ω_4 теплоизолированы как для стекла, так и для воздуха.

В экспериментальных расчётах использовались температурные зависимости плотности ρ_{air} , динамической вязкости μ_{air} , теплоёмкости $C_{p,air}$ и теплопроводности λ_{air} воздуха, аппроксимированные по данным [11], а также плотности ρ_{gl} , теплопроводности λ_{gl} и теплоёмкости $C_{p, glass}$ для стекла [12].

Уравнения сохранения энергии для стекла (Θ_1) и сохранения импульса, энергии и массы для воздуха (Θ_2) имеют вид:

$$\left\{ \begin{array}{l} \Theta_1 : \lambda_{glass} \nabla^2 t_{glass} = 0; \\ \Theta_2 : \rho_{air} (\bar{u} \nabla) \bar{u} = \nabla \left\{ \begin{array}{l} -p + \\ + \mu_{air} [\nabla \bar{u} + (\nabla \bar{u})^T] \end{array} \right\} - \rho_{air} |\bar{g}| \bar{e}_y; \\ \rho_{air} C_{p,air} \bar{u} \nabla t_{air} = \nabla (\lambda_{air} \nabla t_{air}); \\ \nabla (\rho_{air} \bar{u}) = 0, \end{array} \right. \quad (1)$$

где \bar{u} — скорость, м/с;

t_{air}, t_{glass} — температура воздуха и стекла, К;

p — давление, Па;
 \bar{g} — модуль ускорения свободного падения, м/с²;
 \bar{e}_y — орт оси y , верхний индекс T — операция транспонирования.

Граничные условия Ω_1 :

$$t_{glass} = t_c; \Omega_2: t_{glass} = t_h; \Omega_3: \bar{u}_{air} = 0, t_{glass} = t_{air},$$

$$\frac{\partial t_{glass}}{\partial x} = \frac{\partial t_{air}}{\partial x}; \Omega_4: \bar{u}_{air} = 0, \frac{\partial t_{glass}}{\partial y} = 0, \frac{\partial t_{air}}{\partial y} = 0.$$

Термическое сопротивление R_t определяется как отношение перепада температур на границах ($t_h - t_c$) к среднеинтегральной компоненте Q_x вектора плотности полного теплового потока:

$$R_t = \frac{S(t_h - t_c)}{\int_S Q_x dx dy}, \quad (2)$$

где S — площадь интегрирования.

Для решения системы (1) использован метод конечных элементов, реализованный в среде *Comsol Multiphysics*TM [13]. Поскольку разброс среднегодовых температур на территории России доходит до 30°C, а в рамках одной климатической зоны в течение года такой разброс может превышать 50°C [14], для вычислительного эксперимента на вертикальных стенках приняты температуры 283±10 К и 303±15 К. Далее проведен расчёт температурно-скоростных полей (рис. 2, 3) и по (2) рассчитан коэффициент термического сопротивления.

Для анализа структуры тепловых потоков в соответствии с [15] также вычислены поля конвективного теплового потока (3) и кондуктивного теплового потока (4):

$$\bar{q}_{conv} = \rho_p \bar{u} \left(H_t - \frac{p}{\rho_p} \right) \quad (3)$$

$$\bar{q}_{cond} = -\lambda_p \nabla T_p, \quad (4)$$

где H_t — энтальпия, Дж/кг.

Использованы проекции векторов потоков на ось x , учитывающие вертикальное расположение окна.

Разработанная математическая модель не-изотермического ламинарного движения воздуха в пространстве между стёклами позволяет получать поля температуры, векторное поле скорости (рис. 2), модуля скорости и линии тока (рис. 3), а также скалярные поля модуля и векторные поля конвективного и кондуктивного тепловых потоков (рис. 4, 5).

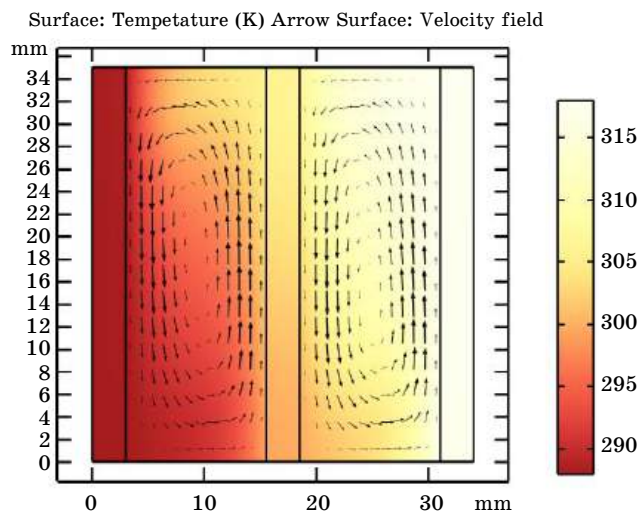


Рис. 2. Температурное поле стекла воздуха и векторное поле скорости воздуха

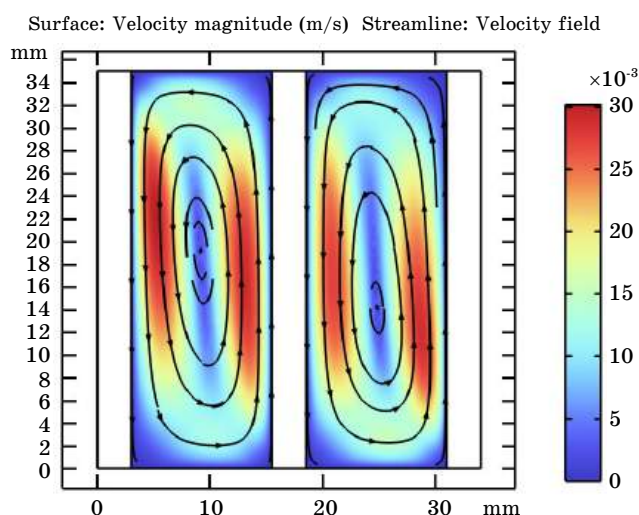


Рис. 3. Поле модуля вектора скорости и линии тока воздуха

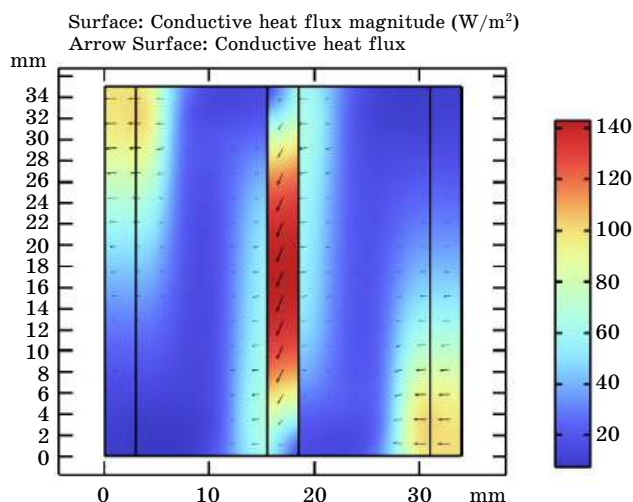


Рис. 4. Скалярное и векторное поля конвективного потока

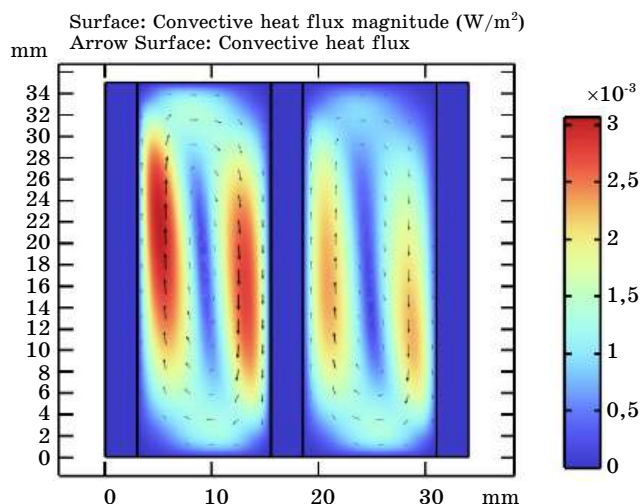


Рис. 5. Скалярное и векторное поля кондуктивного потока

Анализ результатов расчетов

Цветовая шкала на рис. 2 показывает поле температур, которое практически однородно в области стекла и имеет выраженный температурный градиент в области, заполненной воздухом. Векторное поле скорости на рис. 2 показано стрелками. При этом видно, что верхняя часть окна прогревается больше (рис. 3), что обусловлено явлением термоконвекции. На рис. 3 показано скалярное поле модуля скорости воздуха (цветовая шкала) и линии тока (стрелки), демонстрирующее типичное термоконвекционное циркуляционное движение воздуха, вызванное подъемом нагретых масс воздуха вверх за счёт силы Архимеда около стенки с более высокой температурой и нисходящее движение этих масс за счёт силы тяжести при их охлаждении более холодной стенкой.

Цветовой шкалой на рис. 4 показано скалярное поле проекции кондуктивного теплового потока на ось x , а стрелками – векторное поле кондуктивного потока. Видно, что наибольшее значение потока наблюдается в разделяющем камере стекле. Также поток неоднороден по высоте в месте контакта воздуха со стеклом и повышается в области наибольшего перепада температур воздуха и стекла (скапливающийся внизу камеры холодный воздух у нагретой стенки и теплый воздух вверх у холодной стенки).

На рис. 5 цветовая шкала показывает скалярное поле проекции кондуктивного теплового потока на ось x , а стрелки – векторное поле кондуктивного потока. Видно, что по структуре поле повторяет характер поля скоростей с противоположным направлением векторов. При этом сравнительный анализ полей кондуктивного и конвективного потоков на рис. 4 и 5 показывает доминирование конвективного потока

для рассмотренных условий вычислительного эксперимента, что подтверждает необходимость учета эффекта термоконвекции при анализе энергоэффективности стеклопакетов.

Проведённые вычислительные эксперименты показали монотонный рост $R_t(h)$ с увеличением h (рис. 6). При этом зависимость термосопротивления от расстояния между стёклами носит нелинейный характер. При увеличении расстояния между стеклами происходит монотонный рост термического сопротивления определённого значения. При достижении этого значения рост прекращается по причине начала доминирования конвективного механизма теплопереноса.

Прекращение роста температуры говорит о том, что дальнейшее увеличение расстояния между стеклами, которое потребует увеличения размеров всей оконной конструкции, не даёт прироста в термосопротивлении и экономически нецелесообразно.

Смещение точки прекращения роста термосопротивления R_t по оси абсцисс определяется базовой температурой (рис. 6), что свидетельствует о возможности учета положения R_t при проектировании оконных систем изменением рекомендуемых значений расстояний между стеклами для разных температурных режимов эксплуатации, соответствующих разным климатическим зонам.

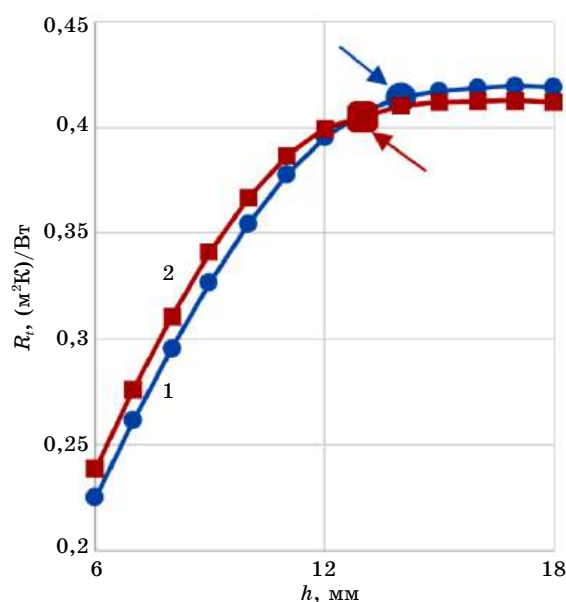


Рис. 6. Зависимость R_t от расстояния между стёклами h для базовых температур 303 ± 15 K (кривая 1) и 283 ± 10 K (кривая 2)

Из анализа результатов видно, что максимум термического сопротивления будет обеспечиваться при размещении разделительного стекла в среднем положении, где оно делит расстояние между внешними стеклами в соотношении 1:1 (рис. 7).

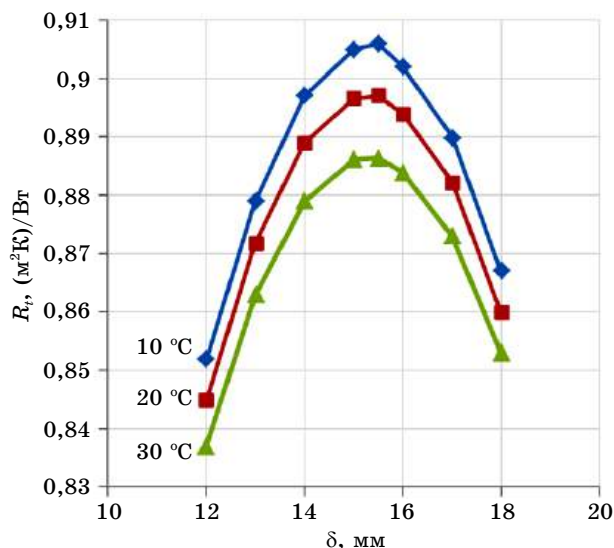


Рис. 7. Зависимость R_t от положения разделяющего камеры стекла δ для базовых температур 10, 20, 30 °C

Это отношение, обеспечивающее экстремум R_t , не зависит от базовой температуры. Также выявлено, что высота окна влияет на его термическое сопротивление только при $H < 300$ мм. Дальнейшее увеличение этого показателя не оказывает существенного влияния на R_t (рис. 8).

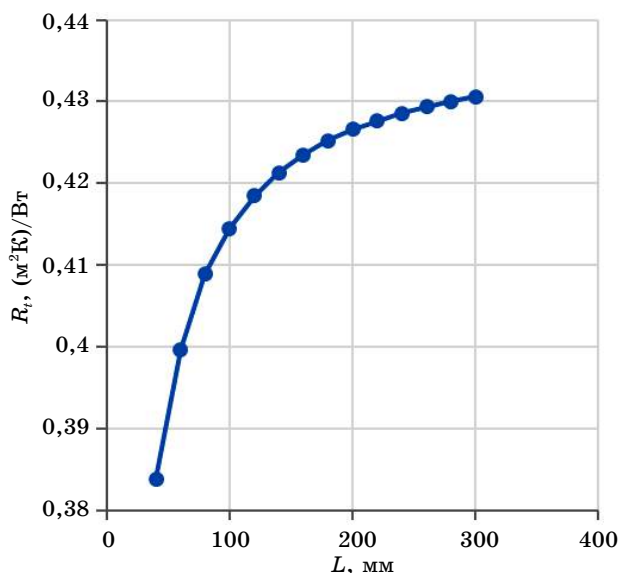


Рис. 8. Зависимость R_t от высоты окна L

Использование идентифицированной точки прекращения роста термосопротивления даёт возможность наращивать величину R_t добавлением дополнительной камеры стеклопакета и,

соответственно, разделительного стекла (рис. 9), тем самым повышая энергоэффективность окна.

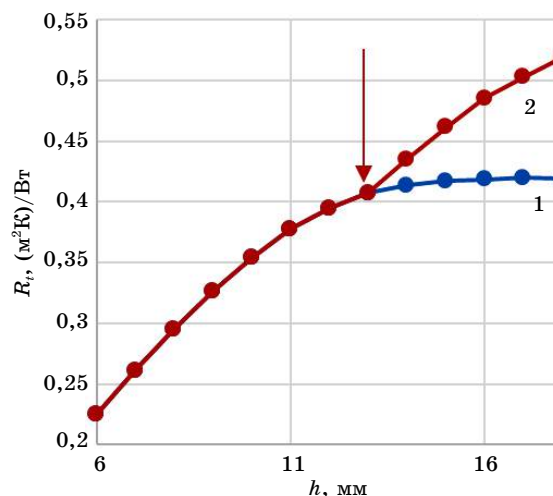


Рис. 9. Зависимость R_t от высоты оконного проёма:

- 1 — $t_{base} = 303$ К, $dt = 15$ К, одна камера;
2 — $t_{base} = 303$ К, $dt = 15$ К, 2 камеры

Таким образом, имея в качестве результатов расчёта функции изменения термосопротивления при различных комбинациях толщины между стеклами для разного количества камер стеклопакета, проектировщик может принять рациональное обоснованное решение о расстоянии между стеклами и количестве камер.

Результаты расчёта можно использовать как автономное справочное руководство в виде составленных по проведенным расчётам таблиц, так и в качестве универсального оперативного средства расчета необходимых параметров для оригинальной конструкции окна или не анализированных ранее климатических зон. Для этого необходимо интегрировать математическую модель в информационную систему САПР. Такую интеграцию можно выполнить с помощью платформы *Comsol Server*TM, которая, используя базы данных теплофизических свойств анализируемых сред, данные о температурном режиме эксплуатации [16] и полученные от проектировщика данные о размерах проектируемого окна, позволяет осуществлять расчёт температурных и скоростных полей и строить в подсистеме поддержки принятия решений функции термического сопротивления для различных конструкций окна (рис. 10). После этого проектировщик может использовать полученные рекомендации и, при необходимости, внести изменения в конструкцию окна в среде САПР.

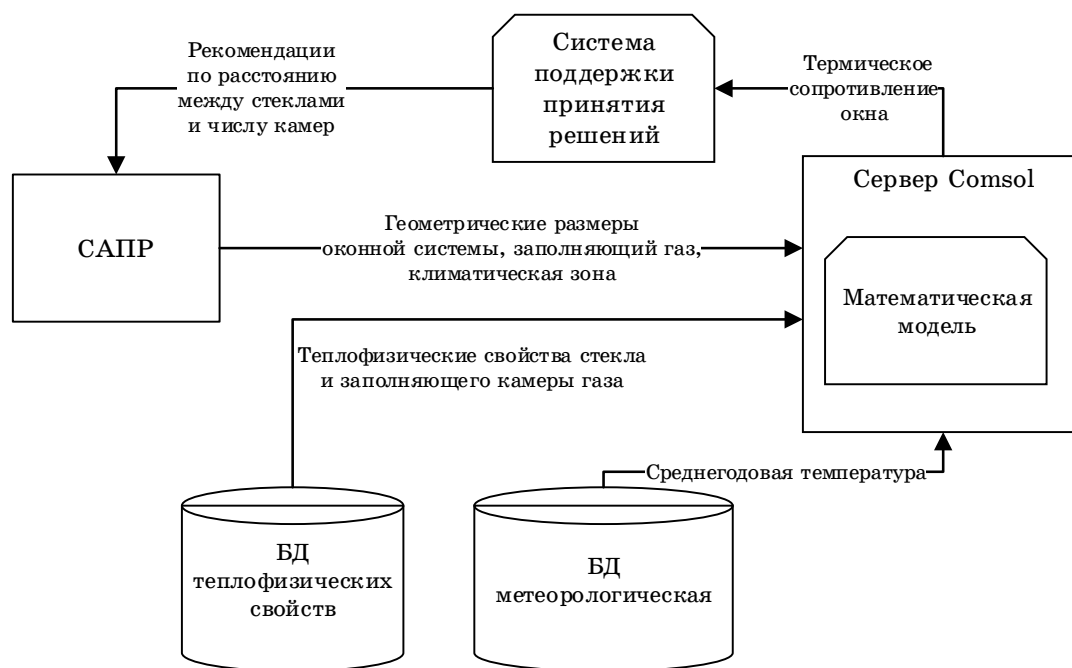


Рис. 10. Структурная схема взаимодействия математической модели с САПР

Заключение

Разработанная математическая модель делает возможным учитывать климатические условия эксплуатации окон при проектировании многокамерных стеклопакетов. Она позволяет подбирать расстояние и количество стекол стеклопакета, опираясь на среднегодовую температуру и отклонения от неё в заданном регионе, обеспечивая таким образом максимум термического сопротивления оконной системы для заданных климатических условий эксплуатации изделия.

Полученные зависимости термического сопротивления от перепада температур на границах остекления относительно базовой температуры, расстояния между стеклами, высоты и положения разделительного стекла позволяют специалистам принимать обоснованные решения при проектировании оконных конструкций для разных климатических зон и могут быть интегрированы в подсистемы поддержки принятия решений по изменению конструкции окна для заданной климатической зоны.

На основе представленной математической модели возможно разработать таблицу рекомендаций к параметрам ГОСТ 24866-2014 для конкретного региона, которая может быть применена в САПР оконных конструкций. Использование данных метеорологических сервисов позволит автоматизировать вычисление среднегодовых температур и их разброса и в дальнейшем применять их для расчётов и выработки рекомендаций по расстоянию между стеклами и необходимости установки дополнительной камеры стеклопакета.

Литература

1. Novak M. H., Nowak E. S. Natural convection heat transfer in slender window cavities // Journal of Heat Transfer-Transactions of The Asme. Vol.115. 1993. P. 476—479.
2. Aydin O. Determination of optimum air-layer thickness in double-pane windows // Energy and Buildings. 32 (3). 2000. P. 303-308.
3. Gorantla K., Shaik S., Setty A. B. T. P. Effect of different double glazing window combinations on heat gain in buildings for passive cooling in various climatic regions of India // 5th International Conference of Materials Processing and Characterization (ICMPC 2016). Materials Today: Proceedings 4. 2017. P.1910-1916.
4. Хвостов А. А., Ряжских В. И., Журавлев А. А., Семенихин О.А. Оптимизация геометрии оконного остекления с учётом климатических особенностей // Математические методы в технологиях и технике. 2022. № 2. С. 64—68.
5. ГОСТ 24866-2014. Межгосударственный стандарт. Стеклопакеты клееные. — М: Стандартинформ, 2015.
6. Amaral A. R., Rodrigues E., Gaspar A. R., Gomes B. A thermal performance parametric study of window type, orientation, size and shadowing effect // Sustainable Cities and Society. Vol. 26. 2016. P.456-465.
7. Gorantla K., Shaik S., Setty A. B. T. P. Effects of Single, Double, Triple and Quadruple Window Glazing of Various Glass Materials on Heat Gain in Green Energy Buildings // Materials, Energy and Environment Engineering: Select Proceedings of ICACE 2015. 2017. P.45-50.
8. Лучшие ПО для расчета пластиковых окон. [Электронный ресурс]. URL: <https://domsdelat.ru/poleznie-soveti/besplatnyie-programmy-dlya-rascheta-pvx-okon.html> (дата обращения 16.03.2022).

9. WinDoPlan – правильное и удобное проектирование окон по ГОСТу. [Электронный ресурс]. URL: <https://voronezh.veka.ru/partners/spravochnik/articles/profile/windoplan-pravilnoe-i-udobnoe-proektirovanie-okon-po-gostu> (дата обращения 16.03.2022).
10. Dalal R., Naylor D., Roeleveld D. A CFD study of convection in a double glazed window with an enclosed pleated blind // *Energy and Buildings*, 2009. 41(11). P. 1256–1262.
11. Варгафтик Н. Б. Справочник по теплофизическим свойствам газов и жидкостей. — М.: Наука, 1972.
12. Немилов С.В. Оптическое материаловедение: Оптические стекла. Учебное пособие. — СПб: СПбГУ ИТМО, 2011.
13. CFD Module User's Guide. [Электронный ресурс]. URL: <https://doc.comsol.com/6.0/doc/com.comsol.help.cfd/CFDModuleUsersGuide.pdf> (дата обращения 16.03.2022).
14. Климат России: монография / Под ред. д-ра геогр. наук, проф. Н. В. Кобышевой. — СПб.: Гидрометеиздат, 2001.
15. Ильинский И.А. Строительная теплофизика. Ограждающие конструкции и микроклимат. — М.: Дин Софт, 1997.
16. Дубинин М., Поспелов И. Источники метеорологических данных на территории РФ по станциям. [Электронный ресурс]. URL: <https://gis-lab.info/qa/meteo-station-sources.html> (дата обращения 16.03.2022).

DESIGN OF WINDOW PANES BASED ON A MATHEMATICAL MODEL OF THERMAL CONVECTION, TAKING INTO ACCOUNT THE CLIMATIC ZONE OF OPERATION

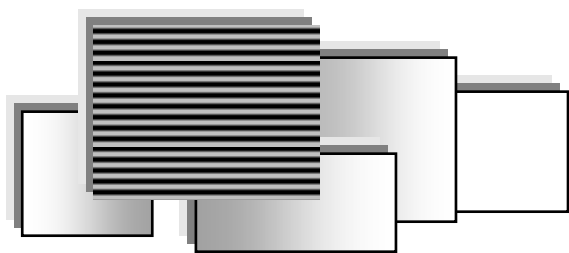
¹ A. A. Khvostov, ² V. V. Tkach, ² A. A. Zhuravlev, ¹ O. A. Semenikhin, ² E. A. Tatarenkov
¹Voronezh State Technical University, Voronezh, Russia
²Military Educational and Scientific Center of the Air Force "Air Force Academy named after Professor N. E. Zhukovsky and Yu. A. Gagarin", Voronezh, Russia

An approach is proposed to take into account the climatic conditions of operation in the design of window systems based on the calculation of the thermal resistance of a multi-chamber double-glazed window, its geometric dimensions and the number of chambers. A mathematical model of non-isothermal laminar air movement in the space between glasses has been developed, which takes into account heat transfer through the glass. The dependences of thermal resistance on the temperature difference at the boundaries of the glazing relative to the base temperature, the distance between the glasses, the height and position of the separating glass are investigated.

Keywords: mathematical modeling, thermogravitational convection, window systems.

Хвостов Анатолий Анатольевич, профессор кафедры "Прикладная математика и механика".
 E-mail: khvtol1974@yandex.ru
Ткач Владимир Владимирович, доцент кафедры "Общепрофессиональные дисциплины".
 E-mail: tkachbalian@yandex.ru
Журавлев Алексей Александрович, доцент кафедры "Математика", «ВУНЦ ВВС ВВА».
 E-mail: zhuraal@rambler.ru
Семенихин Олег Александрович, доцент кафедры "Прикладная математика и механика".
 E-mail: semenihin72@yandex.ru
Татаренков Евгений Анатольевич, доцент кафедры "Общепрофессиональные дисциплины".
 E-mail: tea_86@mail.ru

Статья поступила в редакцию 11 марта 2023 г.



ПРИКЛАДНЫЕ ВОПРОСЫ И ЗАДАЧИ ПРИМЕНЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

УДК 681.518.2

DOI: 10.52190/2073-2597_2023_1_49

EDN: TQRXXL

РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ ТЕХНИЧЕСКИМИ ДАННЫМИ НА ОСНОВЕ КОМПЛЕКСА СТРУКТУРНО-ПАРАМЕТРИЧЕСКОГО МОДЕЛИРОВАНИЯ

А. О. Бутко, канд. техн. наук; И. Н. Каверная

Московский авиационный институт (национальный исследовательский университет),
Москва, Россия

Рассмотрены особенности разработки программных средств для консолидации в единой инвариантной среде больших объемов разнородной информации по сложным техническим изделиям. Предложено разрабатывать систему управления техническими данными на основе комплекса структурно-параметрического моделирования, реализующего концепцию единой среды управления для любых видов данных. Представленное решение может быть использовано в любой проектной сфере, которая требует управления большими объемами технических данных.

Ключевые слова: автоматизация, программирование, интеграция, инженерные расчеты, нормирование, моделирование, структурное моделирование, параметрическое моделирование, базы данных.

Всё большую актуальность приобретает разработка российских аналогов импортных программных продуктов. Не являются исключением и системы управления техническими данными. На рынке уже присутствует несколько отечественных решений, наиболее популярные и качественные продукты это *TFLEX DOCs* от компании "ТОП Системы" и *1C PDM* от "1C". Эти продукты имеют широкий функционал и гибкость с точки зрения настройки, однако первый продукт довольно громоздкий, построен на зарубежной СУБД и привязан к конкретной САПР, а второй, напротив, никак не интегрирован с какой-либо САПР, что многие корпоративные пользователи отмечают в качестве недостатка. В связи с этим видится перспективным направление по разработке полностью отечественной системы управления техническими данными без привязки к конкретной САПР.

В рамках проводимых в Московском авиационном институте (МАИ) научных работ по тематике автоматизации проектирования была поставлена задача формирования комплексного решения по управлению техническими данными,

хранимыми в разных автоматизированных системах от разных поставщиков ПО. В процессе построения итогового продукта были рассмотрены существующие способы программной реализации интерфейсов между системами разных классов.

Исходя из поставленной задачи, требовалось описать структуру системы, определить функции и составить пакет приложений. Схожую задачу решает система *TFLEX*, имеющая в своем составе широкий набор компонент различного назначения, включая функционал *PDM*.

Методы интеграции данных из разных систем

Первый метод интерфейсного решения на базе адаптера [1] заключается в использовании специальной интерфейсной программы — адаптера — для осуществления прямой передачи данных из формата одной системы в формат другой системы.

Адаптер позволяет обеспечить доступ к информационным ресурсам на уровне прикладных программных интерфейсов с *API (Application*

Programming Interface) — функциями, например *BAPI for SAP R/3*, а также на уровне протоколов систем управления базами данных, например *ODBC/JDBC*. Такие инструментальные средства поддерживают широкий набор функций для двухстороннего взаимодействия систем, однако реализация программного интерфейса между системами с помощью специального адаптера имеет определенные специфические недостатки. Например, использование встроенных функций *API* приводит к дополнительным трудностям при реализации интерфейса. Поэтому иногда время на внедрение этого решения и его стоимость значительно увеличиваются. К тому же не каждое предприятие обладает возможностью разработать интерфейсное решение собственными силами или имеет достаточно средств, чтобы заказать такой проект у сторонней компании-разработчика. Поэтому в качестве адаптера часто используют язык *XML*, во многом благодаря тому, что основные системы поддерживают экспорт и импорт данных в *XML*-формат.

При реализации интерфейсного решения на базе адаптера на первое место выходят вопросы защиты форматов и соблюдение всех правил безопасности систем и контроля доступа.

Второй метод — построение решений на основе *XML*-сервера [2]. Средствами *XML* очень просто и удобно описывать достаточно сложные информационные структуры данных в виде *XML*-объектов. В интегрированных решениях роль таких объектов обычно выполняют универсальные бизнес-объекты, которые практически всегда имеют древовидную структуру. Представленные на языке *XML* бизнес-объекты также являются удобным инструментом для обмена информацией между различными приложениями и системами.

Структура информационной системы представляется в виде дерева. Так как вся информация об изделии в системе обычно представлена также в виде дерева, она достаточно просто переносится в формат *XML*. Поэтому этот формат часто поддерживается *PDM*-системами по умолчанию. Для древовидной структуры задается список *XML*-тэгов, описывающий базовые элементы системы. По завершении работы получается описание структуры информационной системы до объектов нижнего уровня. Описываются свойства и методы для всех объектов.

Сервер *XML*-данных создавался в целях представления информации, хранящейся в базе данных (БД) в виде объектов, и реализации бизнес-логики приложения. Иными словами, он позволяет обеспечивать объектный интерфейс доступа к *XML*-данным, их обработку, а также реализацию алгоритмов системы интерфейса. В состав *XML*-объекта может входить информация, не являющаяся частью БД, например ресурсы, иллюстрации или материал, получаемый

из другого приложения. Сервер может быть реализован на основе *WEB*-сервера (*MS IIS*, *Netscape* или *Apache*) или в форме готового приложения, написанного на одном из языков программирования, например, на *C++*.

В основе третьего метода использования *общего формата данных* [1, 3] лежит принцип унификации представления информации в системах разных классов или одного класса, но использующих отличающиеся по структуре, содержанию и описанию форматы хранения данных. Он подразумевает решение, аналогичное применению языка *XML* — по сути, это то же самое, но более узкого применения и определенной специфики.

Метод использования общего формата данных имеет ряд преимуществ: прост в применении с точки зрения пользователя; не требует внедрения дополнительных программных продуктов к уже имеющимся на предприятии; имеет относительно невысокую стоимость (поддержка общего формата включена в систему как одна из ее функций); отсутствует потребность дополнительной подготовки кадров.

Однако при реализации данного метода возникают и проблемы. Одна из них — это необходимость принятия соглашения разработчиками автоматизированных систем (АС) о добавлении в АС функции экспорта информации в универсальном формате. Данная проблема, в свою очередь, порождает несколько задач, которые придется решать в процессе достижения поставленной цели для обеспечения поддержки общего формата данных. Во-первых, само заключение соглашения требует времени на поиск компромисса, поскольку разработчики АС чаще всего по-разному понимают оптимальный общий формат. Во-вторых, возникает вопрос выбора между созданием принципиально нового формата и принятием одного из уже существующих. Эта задача также решается путем поиска компромиссного решения и заключения договоренности между разработчиками.

Еще одной проблемой является различие в составе хранимой и обрабатываемой информации в интегрируемых системах, в частности, отсутствие в одной системе класса данных, необходимых для второй системы. Это значит, что придется добавлять их как новый вид и дополнительно вводить перед экспортом, либо учитывать неполноту информации для представления передаваемого объекта, что в результате может привести к его искажению после импорта. Устранение данной проблемы не всегда реализуемо, что является недостатком метода.

Другой серьезный недостаток — это отсутствие контроля и управления процессом обмена информацией, что приводит к передаче избыточных данных и неконтролируемым сбоям в процессе экспорта, а также возможному искажению их содержания.

Структура комплекса структурно-параметрического моделирования

В основу единой среды управления техническими данными из разных систем был положен разработанный в МАИ комплекс структурно-параметрического моделирования (СПМ), включающий в себя несколько модулей:

- модуль управления проектированием (главный модуль, управляющий запуском и работой других модулей комплекса);
- модуль технологического моделирования (реализует функционал построения технологических решений и формирования технологической документации);
- модуль геометрического моделирования (отвечает за трансляцию модели, построенной лингвистическими средствами с помощью языка СПМ);
- модуль визуализации (отвечает за визуализацию модели изделия, построенной лингвистическими средствами с помощью языка СПМ);
- модуль имитационного моделирования (обеспечивает имитацию функционирования моделируемой системы с помощью анимации или итерационных расчетов);
- модуль интеграции с системами управления проектами (автоматически формирует проект в системе, подобной *Microsoft Project*, и загружает в него информацию по ресурсам);
- модуль обмена данными с системой *TFLEX* (обеспечивает передачу данных между моделями изделия в СПМ и *TFLEX* на уровне параметров);
- модуль обмена данными с системой *NX* (обеспечивает передачу данных между моделями изделия в СПМ и *NX* на уровне параметров);
- модуль обмена данными с *AutoCAD* (обеспечивает передачу данных между моделями

изделия в СПМ и *AutoCAD* на уровне параметров);

- модуль нормирования (выполняет нормирование для технологических операций);
- модуль проверки и корректировки параметров (служит для проверки целостности БД и корректировки параметров при необходимости).

В комплексе СПМ реализован встроенный язык моделирования, позволяющий описывать модели изделий любой сложности. Модели в таком случае представляются в виде набора файлов с "кодом" модели (рис. 1), которые в дальнейшем транслируются по аналогии с программным кодом в "объектный" файл, после чего может быть проведена его визуализация. Данная схема работы позволяет описывать системы без ограничений по степени сложности и составу [4].

```

+ОБЪЕКТ;
КОД=shnek; ИМЯ=Шнек; ТИП=Д;

+ЭЛЕМЕНТЫ;

ЭЛЕМЕНТ=sh1; КОД=T_zag; T=1.2; кол=кол; ИМЯ=подготовка лепестков;
ЭЛЕМЕНТ=sh2; КОД=T_sbor; T=0.8; кол=кол; ИМЯ=сварка винта;
ЭЛЕМЕНТ=sh3; КОД=T_sles; T=0.5; кол=кол; ИМЯ=правка поверхности;
ЭЛЕМЕНТ=sh4; КОД=T_mech; T=0.8; кол=кол; ИМЯ=обработка венца;

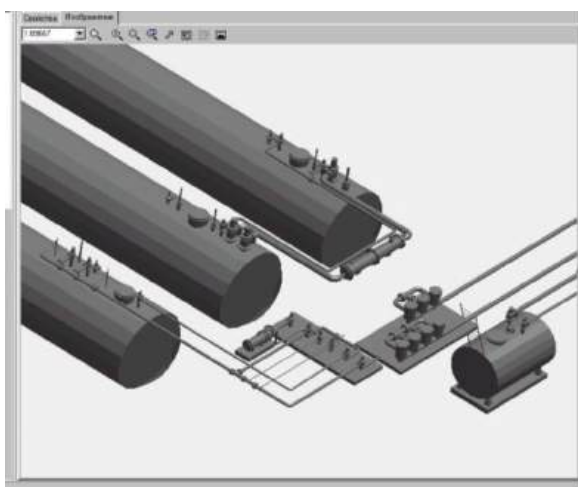
+ПОЛОЖЕНИЕ;
+ПАРАМЕТРЫ;
кол=;
DUR (+) =;

STD (T:)=Wed 02.10.02; !PRD (T:)=none;BAS (+)=8700;

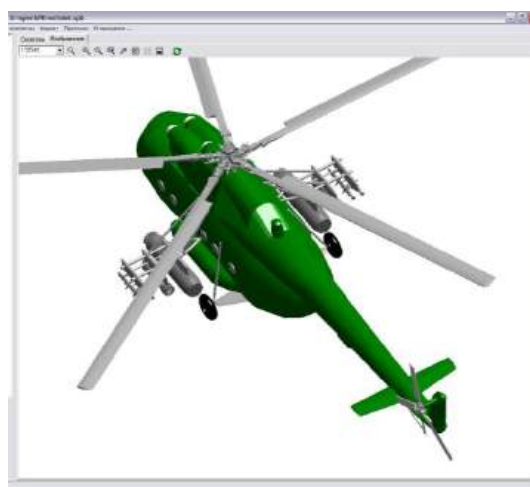
Металлы лист (м:)=22.9;
Труба (м:)=30.9;
+ФОРМУЛЫ;
  
```

Рис. 1. Примеры "кода" модели изделия

В качестве примеров использования комплекса СПМ для моделирования можно привести результаты научно-исследовательских работ студентов магистратуры МАИ. На рис. 2 показаны фрагменты моделей в комплексе СПМ.



а



б

Рис. 2. Примеры моделей в комплексе СПМ:

- а — системы заправки наземного стартового комплекса для ракет космического назначения;
б — модель вертолета Ми-8

Комплекс СПМ использует базу данных собственной разработки и соответствующий функционал для работы с ней. Исходный формат БД разработан сотрудниками МАИ, процедуры доступа к ней реализованы в универсальной библиотеке *spb2.dll*.

В базовой версии комплекс СПМ дает возможность моделировать геометрию, технологию и реализовывать инженерные расчеты. При необходимости оперативно формируется проект в автоматизированной системе управления проектами.

Особенности моделирования в комплексе СПМ

В комплексе СПМ можно проводить инженерные, прочностные, функциональные, экономические и любые другие расчеты, необходимые для проектирования объекта. В "код" модели включены разделы ФОРМУЛЫ и ПАРАМЕТРЫ, в которых можно использовать стандартные способы описания расчетов, схожие с языками программирования (рис. 3). Это позволяет соз-

давать переменные и включать их в формулы по ходу вычислений [5].

Параметрам модели можно задавать следующие свойства:

- признак контроля, позволяющий определять его включение в перечень наиболее важных значений и характеристик;
- признак принадлежности к категории "Трудовой ресурс";
- признак принадлежности к категории "Материал".

Последние два признака нужны для выделения проектных параметров и ресурсов, которые в дальнейшем используются при автоматизированном формировании проектов в системе управления проектами (СУП) при запуске макроса загрузки ресурсов (по аналогии с типами *w-Work m-Material*).

Комплекс СПМ позволяет описывать технологические процессы в виде "кода" с помощью лингвистических средств (рис. 4) и формировать технологическую документацию, которая в итоге генерируется в формате *pdf* или *doc* (по формам из соответствующих ГОСТов).

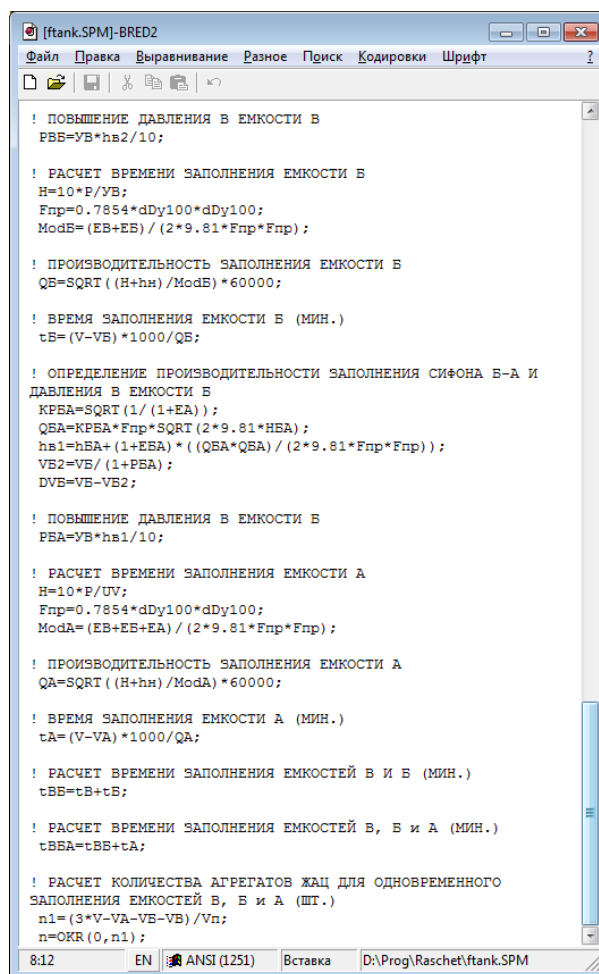
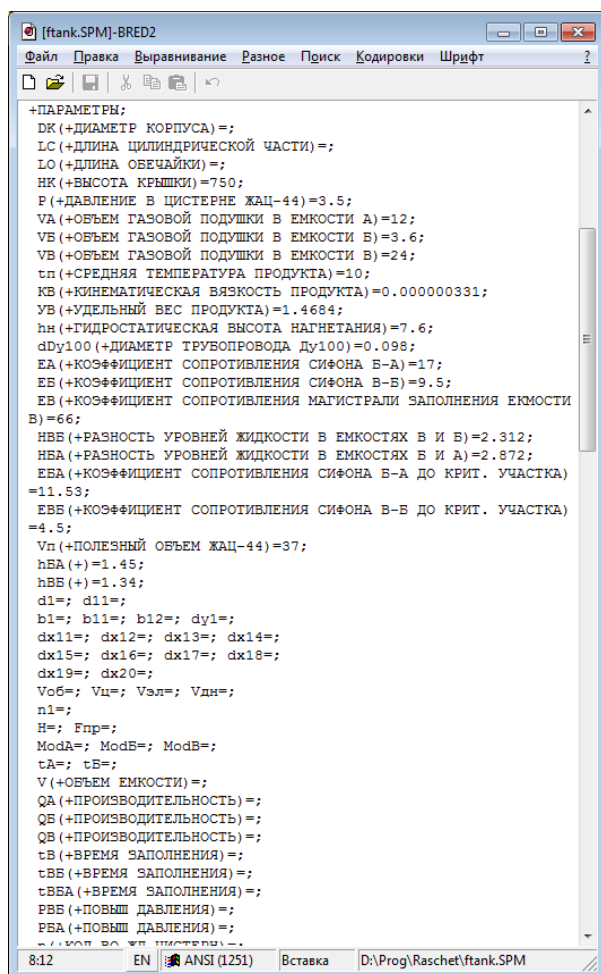


Рис. 3. Примеры расчета в комплексе СПМ

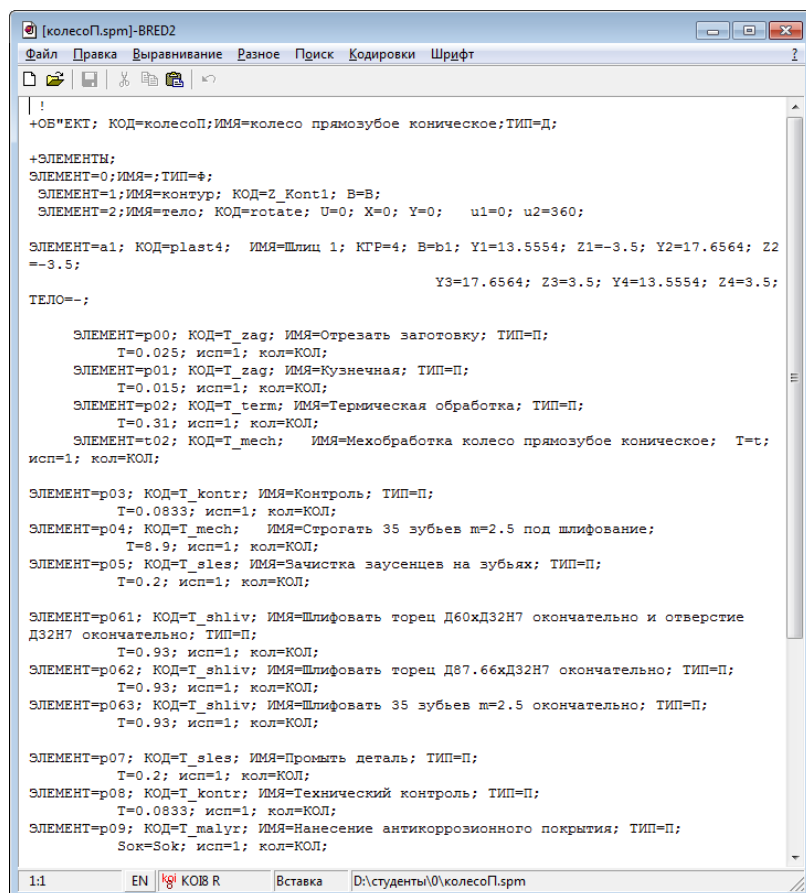


Рис. 4. Фрагмент "кода" в модуле технологического моделирования

Для формирования технологических решений используется иерархическая структура данных технологического процесса (рис. 5), включающая маршруты, операции, переходы.

При этом модель изделия может не содержать никакой геометрии. По результатам проектирования формируются маршрутные или операционные карты.

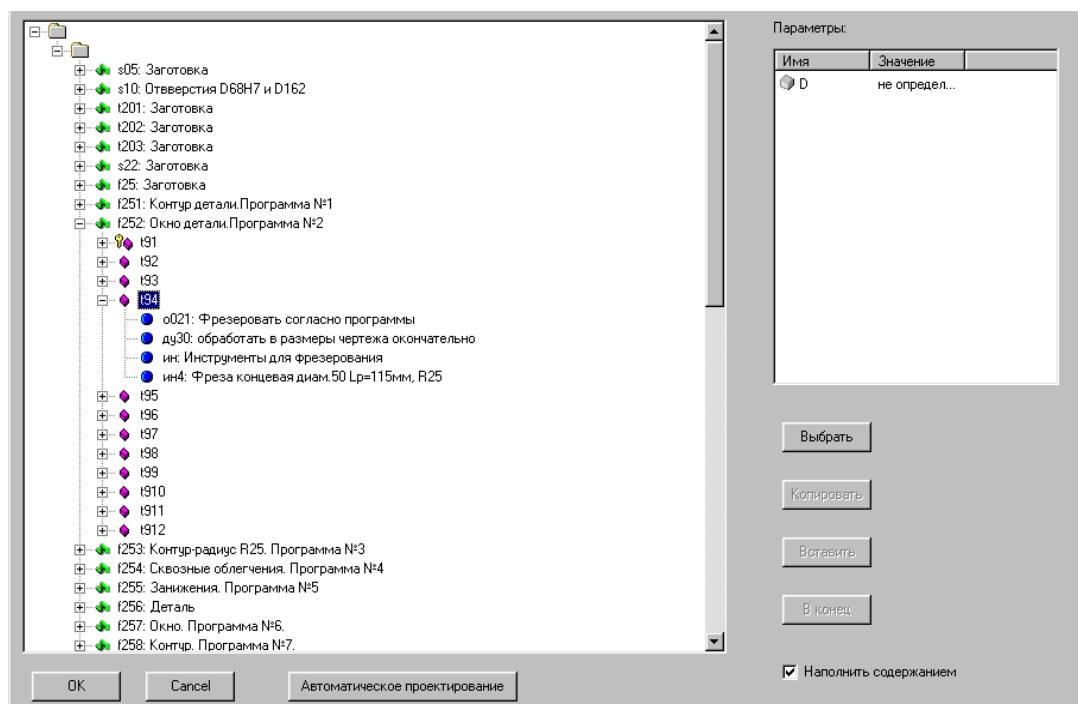


Рис. 5. Формирование технологических решений в комплексе СПМ

Функции технологического проектирования выполняет моделиер *PrTech*. На рис. 6 показано рабочее окно модуля с загруженной моделью. Типичная структура включает операции, переходы и элементарно обрабатываемые поверхности. Приложение проводит обработку данных и выдает набор решений, по которым в дальнейшем можно сформировать документацию в стандартном виде. Обычно файлы документов генерируются в директории с файлами модели изделия, к которому они относятся.

Функции формирования проекта в СУП выполняет модуль интерфейса интеграции с *MS Project*, позволяющий на основе СПМ строить каркас и исходную структуру проекта. Данная операция выполняется через импорт с последующей загрузкой ресурсов макросом [6]. Таким образом, на основе проектных данных, описанных в модели изделия, можно сразу получить проект, практически готовый к использованию в процессах планирования производства. Пример работы модуля показан на рис. 7.

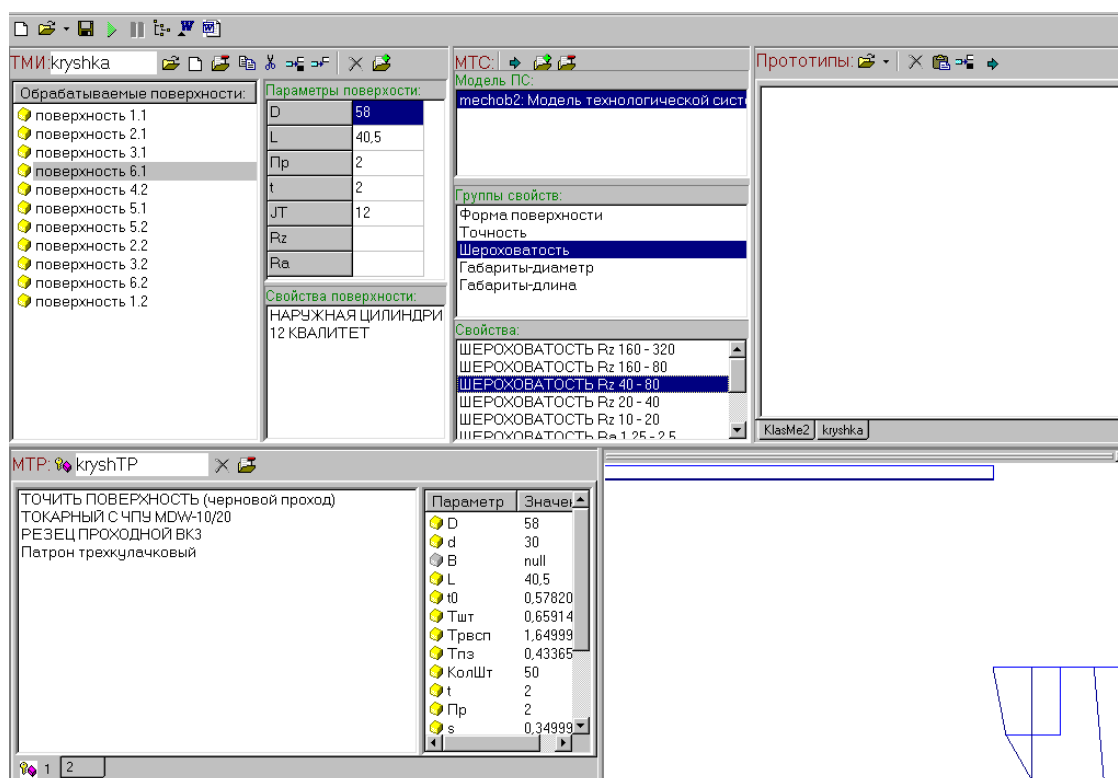


Рис. 6. Технологическое проектирование в комплексе СПМ

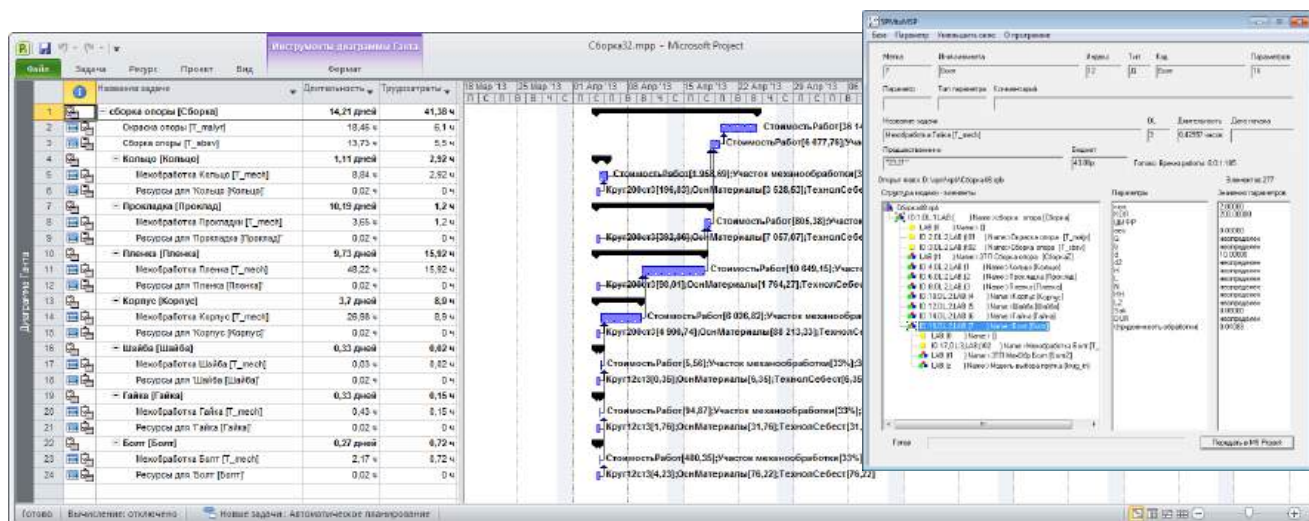


Рис. 7. Формирование проекта в MS Project

Реализация функций управления данными

Опишем функции управления данными, созданными или полученными из других сред проектирования. Для реализации этих функций в составе блока ПАРАМЕТРЫ используют специальные переменные, позволяющие подключать к модели изделия файлы любых форматов. В итоге каждый файл с элементом модели может содержать полный комплект данных по объекту. Такая схема работы механизма интеграции является наиболее гибкой и способствует упорядочению структуры документации по моделируемому изделию любой сложности.

Окно со списками специальных переменных доступно в модуле визуализации *ShowSPB*. Эти переменные описывают свойства элемента (параметры, код, имя, метка), выделенного в дереве модели (рис. 8). Если у выделенного элемента нет специальных переменных, называемых "параметры-файлы", то список в правом нижнем углу отображаться не будет.

На рис. 9 показано окно модуля визуализации *ShowSPB* и диалог со списками параметров и параметров-файлов, которые описывают все подключаемые к модели изделия файлы документации. Как правило, эти файлы располагаются в директории с файлами модели изделия, к которому они относятся.

Пример описания параметров-файлов в файле модели:

+ПАРАМЕТРЫ;

рис1(Ф: модель)=object.tif;

текст1(Ф: проект)=object_proj.mpp;
текст2(Ф: диаграмма)=object_diagr.idd;
текст3(Ф: описание)=object_data.doc;
рис2(Ф: кза)=part1.tif;
рис3(Ф: кза)=part2.tif;
рис4(Ф: модель)=part3.tif;
рис5(Ф: структура)=proton.tif;
рис6(Ф: заправка)=szo7.idd;
текст4(Ф: проект)=proton.mpp;
рис6(Ф: кза)=dnishe.tif;
рис7(Ф: кза)=dnishe1.tif;
рис8(Ф: модель)=dnishe2.tif;

Параметр	Значение
DUR	48.000
PRD	3.519
STD	Thu 10.10
BAS	4500.000
RS1	Стандарт
RV1	3.500
RC1	500.000
RS2	Сварочный
RV2	-100.000
RC2	90.000
RS3	Сборочный

Параметр	Файл
текст1 (проект)	dnishe.mpp
рис4 (кза)	leaf.tif
рис5 (кза)	shpang.tif
рис6 (модель)	flange.tif

Рис. 8. Окно свойств элемента

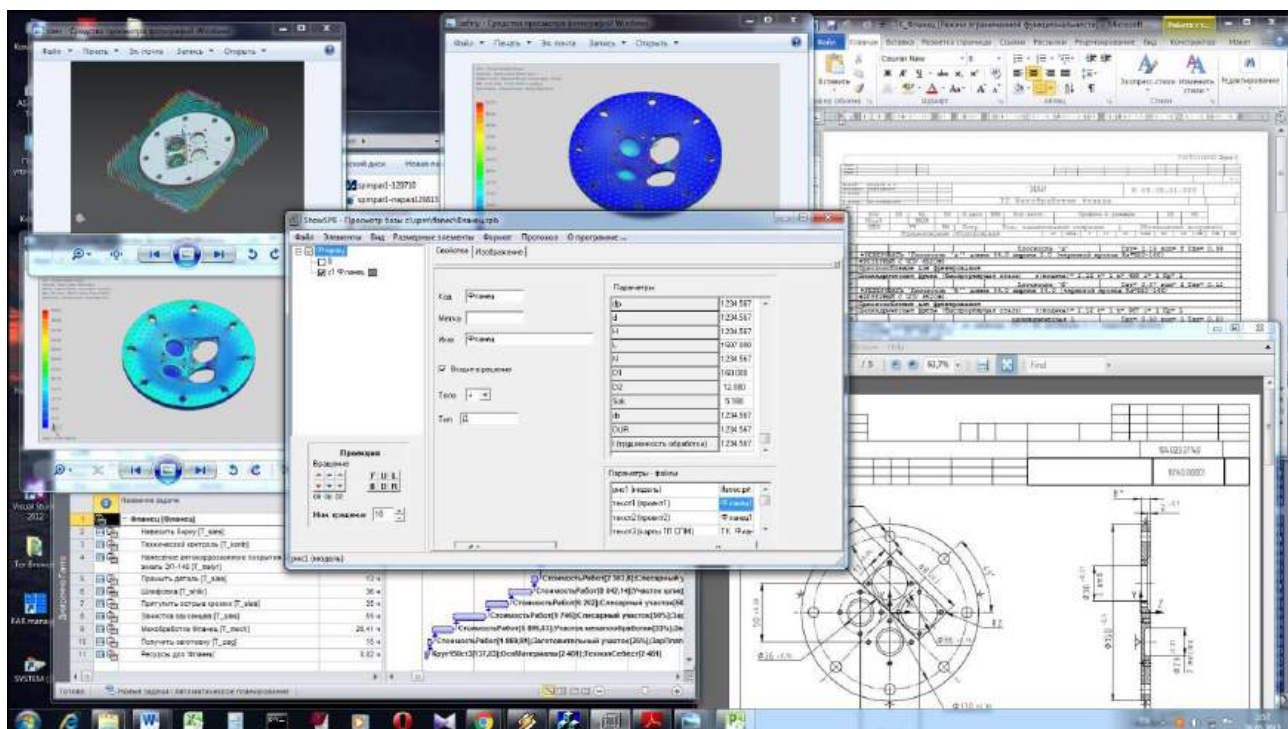


Рис. 9. Загрузка данных по изделию из комплекса СПМ

Возможность интеграции ресурсов позволяет консолидировать в комплексе СПМ любую документацию в любом формате, что обеспечивает удобство, наглядность и быстрый доступ ко всем компонентам, представляющим моделируемый технический объект или систему.

Большинство модулей комплекса СПМ было реализовано в среде *Microsoft Visual Studio* с использованием традиционных инструментов программирования и разработки интерфейсов [7, 8].

Комплекс СПМ обладает преимуществами по сравнению с продуктом *TFLEX*:

- способен автоматически формировать проект в СУП;
- способен реализовать любые расчеты любой сложности;
- имеет низкие требования к аппаратным ресурсам (может работать на низкопроизводительных процессорах и с небольшим объемом памяти);
- использует собственный формат базы данных (нет необходимости приобретать и ставить СУБД типа *SQL* для *PDM*).

Заключение

Представленная система управления данными реализует процесс управления техническими данными по изделиям любого назначения в единой инвариантной среде моделирования. Описанная система является отечественной разработкой со своей базой данных, что, очевидно, является преимуществом в условиях перехода к импортозамещению. Система управления данными состоит из модулей, которые удобно варьировать, и имеет простой пользовательский интерфейс, разработанный с учетом принципов

разработки интерфейсов, что облегчает работу проектировщиков.

Апробация системы управления данными проведена в ГКНПЦ им. М.В. Хруничева. Предлагаемое решение управления техническими данными на основе комплекса СПМ отличается большой трудоемкостью подготовки модели изделия. В дальнейшем для снижения трудоемкости планируется разработка модуля формирования программ для станков с ЧПУ и использование визуальной среды моделирования вместо лингвистических средств описания технологических процессов, трансляции и расчета модели.

Литература

1. Думченков И. А. Обзор методов интеграции информационных систем, их преимуществ и недостатков // Молодой ученый. 2018. № 23(209). С. 176—177.
2. Морозова О. А. Интеграция корпоративных информационных систем: М80 учеб. пособие. — М.: Финансовый университет, 2014. — 140 с.
3. Рындин А. А., Саргсян Э. Р. Современные стандарты информационного взаимодействия систем: учеб. пособие. — Воронеж: Изд-во ВГТУ, 2021. — 144 с.
4. Норенков И. П. Основы автоматизированного проектирования: учеб. для ВУЗов. Изд. 2-е, перераб. и доп. — М.: Изд-во МГТУ им. Н. Э. Баумана, 2002.
5. Цырков А. В. Методология проектирования в мультиплексной информационной среде: Монография. — М.: ВИМИ, 1998.
6. Куперштейн В. Microsoft Project в делопроизводстве и управлении. Серия: Мастер решений. БНВ. — СПб., 2003.
7. Тидвелл Д. Разработка пользовательских интерфейсов. — СПб: Питер, 2008.
8. Кариев Ч. А. Разработка Windows-приложений на основе Visual C# и связь. — М., 2007.

DEVELOPMENT OF A TECHNICAL DATA MANAGEMENT SYSTEM BASED ON SPM

A. O. Butko, I. N. Kavernaya

Moscow Aviation Institute (National Research University), Moscow, Russia

The features of the development of software tools for the consolidation in a single invariant environment of large volumes of heterogeneous information on complex technical products are considered. It is proposed to develop a technical data management system based on a complex of structural-parametric modeling (SPM), which implements the concept of a single control environment for any type of data. The presented solution can be used in any design field that requires the management of large amounts of technical data.

Keywords: automation, programming, integration, engineering calculations, normalization, modeling, structural modeling, parametric modeling, databases.

Бутко Антон Олегович, доцент.

E-mail: mati-tias@yandex.ru

Каверная Ирина Николаевна, старший преподаватель.

E-mail: kavernaya@list.ru

Статья поступила в редакцию 22 ноября 2022 г.