Intellectual Technologies on Transport
No 3

Интеллектуальные технологии на транспорте
№ 3

Санкт-Петербург
St. Petersburg
2019

Интеллектуальные технологии на транспорте
№ 3, 2019

Сетевой электронный научный журнал, свободно распространяемый через Интернет.
Публикуются статьи на русском и английском языках с результатами исследований и практических достижений в области интеллектуальных технологий и сопутствующих им научных исследований.

Журнал основан в 2015 году.

Учредитель и издатель

Федеральное государственное бюджетное образовательное учреждение высшего образования «Петербургский государственный университет путей сообщения Императора Александра I» (ФГБОУ ВО ПГУПС)

Сопредседатели редакционного совета

Панычев А. Ю., ректор ПГУПС, С.-Петербург, РФ
Чаркин Е. И., директор по ИТ ОАО «РЖД», Москва, РФ

Главный редактор

Хомоненко А. Д., проф., С.-Петербург, РФ

Редакционный совет

Глухов А. П., вед. НС ГВЦ ОАО «РЖД», Москва, РФ
Дудин А. Н., д.т.н., проф., БГУ, Минск, Беларусь
Илларионов А. В., советн. «РФЯЦ-ВНИИЭФ», Саров, РФ
Корниенко А. А., проф., ПГУПС, С.-Петербург, РФ
Ковалец П., проф., Тех. ун-т, Варшава, Польша
Меркурьев Ю. А., проф., РТУ, Рига, Латвия
Нестеров В. М., проф., С.-Петербург, РФ
Пустарнаков В. Ф., ген. дир. «Газинформсервис», С.-Петербург, РФ
Титова Т. С., проф., проректор. ПГУПС, С.-Петербург, РФ
Федоров А. Р., ген. дир. «ДигДез», С.-Петербург, РФ
Юсупов Р. М., проф., чл.-корр. РАН, С.-Петербург, РФ

Редакционная коллегия

Бубнов В. П., проф., С.-Петербург, РФ – зам. гл. ред.
Ададуров С. Е., проф., Москва, РФ
Александрова Е. Б., проф., С.-Петербург, РФ
Атилла Элчи, проф., Аксарай, Турция
Безродный Б. Ф., проф., Москва, РФ
Благовещенская Е. А., проф., С.-Петербург, РФ
Булавский П. Е., д.т.н., доц., С.-Петербург, РФ
Василенко М. Н., проф., С.-Петербург, РФ
Гуда А. Н., проф., Ростов-на-Дону, РФ
Железняк В. К., проф., Полоцк, Беларусь
Заборовский В. С., проф., С.-Петербург, РФ
Зегжда П. Д., проф., С.-Петербург, РФ
Канаев А. К., д.т.н., проф., С.-Петербург, РФ
Котенко А. Г., д.т.н., доц., С.-Петербург, РФ
Куренков П. В., проф., Москва, РФ
Лецкий Э. К., проф., Москва, РФ
Мирзоев Т. А., асс. проф., Джорджия, США
Наседкин О. А., доц., С.-Петербург, РФ
Никитин А. Б., проф., С.-Петербург, РФ
Охтилев М. Ю., проф., С.-Петербург, РФ
Соколов Б. В., проф., С.-Петербург, РФ
Таранцев А. А., проф., С.-Петербург, РФ
Утепбергенов И. Т., проф., Алматы, Казахстан
Филипченко С. А., доц., Москва, РФ
Фозилов Ш. Х., проф., Ташкент, Узбекистан
Фу-Ниан Ху, проф., Джиангсу, Китай
Хабаров В. И., проф., Новосибирск, РФ
Ходаковский В. А., проф., С.-Петербург, РФ
Чехонин К. А., проф., Хабаровск, РФ
Яковлев В. В., проф., С.-Петербург, РФ
Ялышев Ю. И., проф., Екатеринбург, РФ

Адрес редакции:
190031, Санкт-Петербург, Московский пр., 9, ауд. 1–210
e-mail: itt-pgups@yandex.ru, сайт: http://itt-pgups.ru

ISSN 2413-2527

Журнал зарегистрирован Федеральной службой по надзору в сфере связи и массовых коммуникаций, свидетельство ЭЛ № ФС77-61707 от 07 мая 2015 г.

Журнал зарегистрирован в Российском индексе научного цитирования (РИНЦ).

© Федеральное государственное бюджетное образовательное учреждение высшего образования «Петербургский государственный университет путей сообщения Императора Александра I», 2019

Разрешается воспроизведение в прессе, а также сообщение в эфире или передача по кабелю опубликованных в составе периодического издания – журнала «Интеллектуальные технологии на транспорте» – статей по текущим экономическим, политическим, социальным и религиозным вопросам с обязательным указанием автора статьи и сетевого электронного научного периодического издания журнала «Интеллектуальные технологии на транспорте».

Intellectual Technologies on Transport
Issue № 3, 2019

Network electronic scientific journal, open access. It publishes articles in Russian and English with the results of research and practical achievements in the field of intelligent technologies and associated research

Founded in 2015

Founder and Publisher

Federal State Educational Institution of Higher Education
«Emperor Alexander I Petersburg State Transport University»

Co-chairs of the Editorial Council

Panychev A. Yu., rector of PSTU, St. Petersburg, Russia
Charkin E. I., director on IT of JSC “RZD”, Moscow, Russia

Editor-in-Chief

Khomonenko A. D., Prof., St. Petersburg, Russia

Editorial Council Members

Glukhov A. P., Lead. Res., CCC of JSC «RZD», Moscow, Russia
Dudin A. N., Prof., BSU, Minsk, Belarus
Illarionov A. V., advisor, «RFNC-VNIIEF», Sarov, Russia
Kornienko A. A., Prof., PSTU, St. Petersburg, Russia
Kovalets P., Prof., Tech. University, Warsaw, Poland
Merkuryev Yu. A., Prof., Academician of the Latvian Academy of Sciences, Riga, Latvia
Nesterov V. M., Prof., St. Petersburg, Russia
Pustarnakov V. F., CEO at «Gazinformservice» LTD., St. Petersburg, Russia
Titova T. S., Prof., PSTU, St. Petersburg, Russia
Fedorov A. R., CEO at «Digital Design» LTD., St.Petersburg, Russia
Yusupov R. M., Prof., Corr. Member of RAS, St.Petersburg, Russia

Editorial Board Members

Bubnov V. P., Prof., St. Petersburg, Russia – Deputy Editor-in-Chief
Adadurov S. E., Prof., Moscow, Russia
Aleksandrova E. B., Prof., St. Petersburg, Russia
Attila Elci, Prof., Aksaray, Turkey
Bezrodny B. F., Prof., Moscow, Russia
Blagoveshenskaya E. A., Prof., St. Petersburg, Russia
Bulavsky P. E., Dr. Sc., As. Prof., St. Petersburg, Russia
Vasilenko M. N., Prof., St. Petersburg, Russia
Guda A. N., Prof., Rostov-on-Don, Russia
Geleznyak V. K., Prof., Polotsk, Belarus
Zaborovsky V. S., Prof., St. Petersburg, Russia
Zegzda P. D., Prof., St. Petersburg, Russia
Kanayev A. K., Prof., St. Petersburg, Russia
Kotenko A. G., Dr. Sc., As. Prof., St. Petersburg, Russia
Kurenkov P. V., Prof., Moscow, Russia
Letsky Ad. K., Prof., Moscow, Russia
Mirzoev T., As. Prof., Georgia, USA
Nasedkin O. A., As. Prof., St. Petersburg, Russia
Nikitin A. B., St. Petersburg, Russia
Okhtilev M. Yu., Prof., St. Petersburg, Russia
Sokolov B. V., Prof., St. Petersburg, Russia
Tarantsev A. A., Prof., St. Petersburg, Russia
Utepbergenov I. T., Prof., Almaty, Kazakhstan
Filipchenko S. A., As. Prof., Moscow, Russia
Fozilov Sh. Kh., Prof., Tashkent, Uzbekistan
Fu-Nian Hu, Prof., Jiangsu, China
Khabarov V. I., Prof., Novosibirsk, Russia
Khodakovsky V. A., Prof., St. Petersburg, Russia
Chekhonin K. A., Prof., Khabarovsk, Russia
Jakovlev V. V., Prof., St. Petersburg, Russia
Jalyshev Yu. I., Prof., Ekaterinburg, Russia

Editorial adress:
190031, St. Petersburg, Moskovskiy pr., 9, aud. 1–210
e-mail: itt-pgups@yandex.ru, http://itt-pgups.ru

ISSN 2413-2527

The journal is registered by the Federal Service for Supervision of Communications and Mass Media, EL no. FS77-61707 testimony from May 7, 2015.

The journal is registered in the Russian Science Citation Index (RSCI).

© Federal State Educational Institution of Higher Education “Emperor Alexander I Petersburg State Transport University”, 2019

The reproduction in the press, as well as a message broadcast or cable published as part of the periodical – journal “Intellectual Technologies on Transport” – articles on current economic, political, social and religious issues with the obligatory indication of the author, and the network of electronic scientific periodical journal “Intellectual Technologies on Transport“.

Содержание

Андреев А.В.
Метод проектирования структуры хранения данных службы каталогов для уменьшения времени отклика при запросе к большим объемам данных 5

Барановский А.М., Ермолаев Ю.Л.
Алгоритмы автоматической визуализации 3D-моделей наноструктур для использования в программах расчета физических свойств наноматериалов методами молекулярной динамики . . . 11

Васнихин В.И., Хомоненко А.Д., Носова С.И.
Автоматизированная система адаптивного управления насосами наливных терминалов на основе логического контроллера 21

Дорожко И.В., Петухов А.Б., Салун О.А.
Модель определения периодичности контроля технического состояния сложных технических комплексов с использованием нечеткой логики. 27

Нго Т.К.
Концептуальная и графовая модели взаимодействия беспилотного летательного аппарата наземной робототехнической сервисной платформой 35

Баев А.В., Самонов А.В.
Комплекс программных средств разработки и верификации требований и проектных решений АСУ объектами транспортной инфраструктуры 42

Султонов Ш.Х., Султонова З.Р.
Программный продукт автоматизации технологических процессов моделирования конструкций вагонов на предприятиях АО «Узбекистон темир йуллари» 53

Ал-Нуайми В.А., Халил М.М., Аббас А.Х.
Расчет качества услуг интернет-провайдера в центральном регионе Ирака с использованием статистики (на англ.) 61

Contents

Andreyev A.V.
A Method for Designing a Directory Service Data Storage Structure to Reduce Response Time When Querying for Large Amounts of Data 5

Baranovsky A.M., Ermolaev Yu.L.
Algorithms for the Automatic Visualization of 3D Models of Nanostructures for Use in Software for Calculating the Physical Properties of Nanomaterials by Molecular Dynamics Methods 11

Vasnikhin V.I., Khomonenko A.D., Nosova S.I.
The Automated System of Adaptive Control of Pumps Filling Terminals on the Basis of the Logical Controller 21

Dorozhko I.V., Petukhov A.B., Salun O.A.
Model for Determining the Frequency of Monitoring the Technical Condition of Complex Technical Systems Using Fuzzy Logic 27

Ngo T.Q.
Conceptual and Graph Models of Interaction of Unmanned Aerial Vehicle with Ground Robotic Service Platform 35

Bayev A.V., Samonov A.V.
Complex of Software for Development and Verification of Requirements and Design Solutions Automated Control System by Transport Infrastructure Objects 42

Sultonov Sh.Kh., Sultonova Z.R.
Software for Automation of Technological Processes of Modeling of Car Structures at the Enterprises of JSC «Uzbekiston Temir Yullari» 53

Al-Nuaami W.A., Khalil M.M., Abbas A.Kh.
Calculation of Quality of ISP Services in Central Region of Iraq Using Statistics (in English) 61

Метод проектирования структуры хранения данных службы каталогов для уменьшения времени отклика при запросе к большим объемам данных

А.В. Андреев
Санкт-Петербургский государственный университет аэрокосмического приборостроения
Санкт-Петербург, Россия
aathexf@gmail.com

Аннотация. **Предлагается метод проектирования структуры хранения данных службы каталогов для уменьшения времени отклика при запросе к большим объемам данных. Рассматриваются службы каталогов LDAP и сетевая структура хранения данных, где записи каталога хранятся в соответствующих ветках каталога для каждого заданного сервиса, использующего службу каталогов. Записи группируются условиями приоритетности сервисов. Истинное выполнение нескольких условия для одной записи реализовано механизмом записей-ссылок, соответствующих стандарту LDAP.**

Ключевые слова: **служба каталогов, LDAP, метод, время отклика, структура каталога.**

Введение

Служба каталогов (Directory Service) – база данных, используемая как средство иерархического представления информации о некотором наборе объектов и предоставления услуг доступа к этой информации. В качестве объектов выступают материальные ресурсы, персонал, сетевые ресурсы и т. д.

LDAP (Lightweight Directory Access Protocol) – стандарт построения каталогов общего назначения. Протокол был разработан для предоставления разработчикам возможности четко определять, какие данные должен хранить каталог. Общая архитектура LDAP позволяет управлять большим количеством различных записей каталога. Большинство организаций различных отраслей производства используют одну из реализаций служб каталогов LDAP, например Microsoft Active Directory, OpenLDAP, 389-DS или другие.

Особенность служб каталогов LDAP заключается в стандартизированном способе построения структуры хранения данных службы каталогов в виде иерархической структуры [1]. В большинстве случаев записи таких каталогов структурированы по принципу физической принадлежности объекта некой структуре организации, например отделам организации, где все записи хранятся в рамках одной ветки каталога: так, например, записи пользователей хранятся в общей ветке каталога "Пользователи" или "Сотрудники" с начальной вершиной "ou=People, dc=example, dc=com". Такой же подход построения структуры каталога можно встретить в соответствующих монографиях [2].

Службы каталогов являются основным средством построения современных глобальных информационных систем, позволяющим решать как проблемы интеграции информационных ресурсов, так и вопросы управляемости ресурсами и сервисами сети любого масштаба, вплоть до столь обширной, как Интернет. Служба каталогов является средством построения стабильной и вместе с тем гибкой сетевой инфраструктуры, предназначенной для обеспечения и контроля взаимодействия пользователей или устройств с любыми сетевыми ресурсами и сервисами.

Основная характеристика работы службы каталогов

Различные сервисы сети используют службу каталогов для аутентификации, авторизации или хранения данных. Для доступа в сеть и работы с сетевыми ресурсами служба каталогов используется как единая система идентификации пользователей: при входе в сеть пользователь представляется системе, проходит процедуры аутентификации и авторизации, после чего может использовать любые ее сервисы без повторной аутентификации. Такая система не только облегчает работу пользователя, но и позволяет более эффективно организовывать работу других сервисов сети, например отправку пользователем электронных сообщений, хранение служебной информации пользователя и распределение прав доступа к ней, предоставление пользователю определенных полномочий. Процесс аутентификации и авторизации всех пользователей сети или поиска информации об объекте каталога может занимать длительное время даже с применением всех оптимизационных шагов, что приводит к простоям и задержкам всех последующих сервисов сети, использующих службу каталогов. Скорость получения данных от службы каталогов влияет на время последующей обработки данных следующим сервисом и вывода результатов пользователю, который инициализирует изначальный запрос к службе каталогов. Таким образом, основной характеристикой эффективности работы службы каталогов является временная характеристика, а именно время отклика на запрос (system response time).

Методы оптимизации работы службы каталогов

Для улучшения временной характеристики службы каталогов применяются следующие оптимизационные методы:

- кэширование данных;
- индексирование наиболее важных атрибутов [3].

Однако данные методы имеют ряд ограничений, а эффективность их применения зависит от того, как используются каталог и данные, хранящиеся в каталоге, а также от общего количества записей каталога [4].

Одним из таких ограничений, например для индексирования, является тип фильтра и параметры его использования. Тип фильтра влияет на эффективность индексирования, например индексы присутствия и подстроки. Одним из решений оптимизации такого каталога может быть использование изменения структуры каталога по определенному заранее критерию, где ветка поиска будет заменять основной фильтр поиска и уменьшать общее количество записей для выборки. Данное изменение может быть реализовано использованием сетевой структуры хранения данных [5].

Сетевая структура хранения данных службы каталогов

Согласно RFC 4512 [6], дуга между вершинами графа, задающего иерархическую структуру каталога, обозначает физическую принадлежность объектов, отраженных в каталоге. Такое построение структуры каталога удобно для хранения записей о сотрудниках организации, где записи группируются по принадлежности к отделам организации. Однако все записи сотрудников при такой структуре все равно хранятся в одной общей ветке каталога, например "ou=People", которая обозначает физическую принадлежность реальным объектам в рамках одной организации. Таким образом, записи группируются по одному признаку (принадлежность к отделу) в рамках одной общей группы всех сотрудников. Службы сети, такие как файловый сервер, вынуждены быть сконфигурированы на использование общей ветки каталога "ou=People" с методом поиска SUP для предоставления сервиса файлового сервера. Такое построение каталога, основанное на физической принадлежности объектов, соответствующих записям каталога, влияет на организации с большим количеством сотрудников, так как все записи хранятся в одной ветке, что, как было исследовано ранее, влияет на время отклика при запросе к данным.

Вместо указанного выше обозначения дугой между вершинами графа предлагается понимать отражение выполнения условия принадлежности:

- с сервисом, использующим службу каталогов (например, файловый сервер Samba);
- фильтром поиска;
- другими условиями, задающими приоритет объектов каталога.

Согласно RFC 4511 [7], позволяется хранить, записывать, осуществлять поиск записи в разных ветках каталога средствами записей-ссылок, а сервисам – работать с ними. Группируя записи по их общим условным параметрам, можно построить более упорядоченную структуру хранения данных на основе сетевой модели данных. Более того, отслеживая изменения в активности сервисов, можно организовать адаптивную службу каталогов, меняющую свою структуру в зависимости от динамики запросов к ней.

Таким образом, изменение структуры каталога по определенным условиям и критериям приоритетности условий следует рассматривать как дополнительный способ улучшения эффективности работы службы каталогов, позволяющий существенно уменьшить общее количество записей для поиска и, соответственно, время отклика на запрос. Данное изменение задает новый способ построения структуры хранения данных службы каталогов, которая организована по условиям приоритетности, задаваемым приложениями, использующими службу каталогов, а не по физическому или организационному признаку принадлежности записей.

На примере службы каталогов, работающей с двумя сервисами (файловый сервер и прокси-сервер), стандартную иерархическую структуру хранения данных, изображенную на рисунке 1, можно представить в виде сетевой структуры хранения данных, изображенной на рисунке 2, где записи каталога сгруппированы по сервисам, использующим их, а сервисы настраиваются на работу с соответствующей веткой каталога.

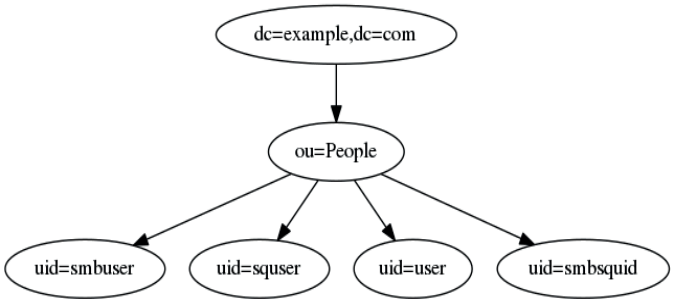

Рис. 1. Стандартная иерархическая структура каталога

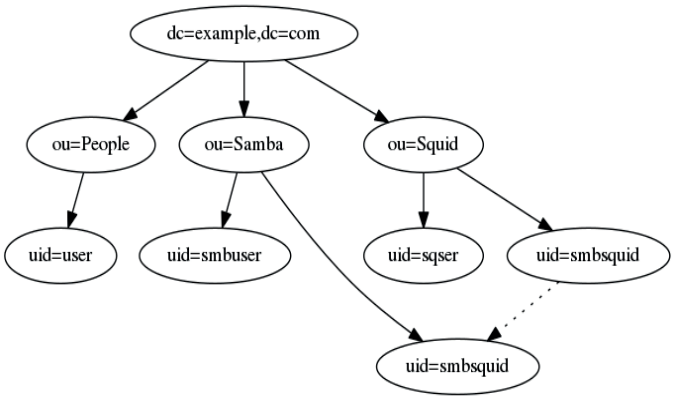

Рис. 2. Предлагаемая сетевая структура каталога

Основным ограничением применения такого способа проектирования и построения структуры хранения данных службы каталогов является то, что условие, по которому группируются записи каталога, не должно быть истиной для большинства записей исходной ветки каталога, так как эффективность применения метода зависит от того, насколько уменьшится исходное количество записей для поиска в заданной ветке.

Метод проектирования структуры хранения данных службы каталогов для уменьшения времени отклика при запросе к большим объемам данных

Для построения предлагаемой сетевой структуры хранения данных службы каталогов предлагается использовать соответствующий метод проектирования структуры хранения данных службы каталогов для уменьшения времени отклика при запросе к большим объемам данных, который можно сформулировать следующей последовательностью шагов:

1. Определить структуру и типы записей используемого каталога.

Следует определить расположение записей в ветках каталога, к которым следует применять метод. Например, записи пользователей Microsoft Active Directory хранятся в ветке каталога "cn=Users, dc=example, dc=com", а записи службы каталогов OpenLDAP могут быть расположены в ветке каталога "ou=People, dc=example, dc=com", где "dc=example, dc=com" – корневая вершина структуры каталога.

Также следует определить отображение объекта каталога в реальном мире. Например, записи в исходной ветке поиска могут быть отображением пользователей корпоративной сети, или ЭВМ, или принтеров сети. В зависимости от этого можно определять различные условия группирования записей.

2. Определить сервисы сети, работающие со службой каталогов.

Для определения условий группирования записей следует определить набор сервисов, работающих со службой каталогов. Для этого следует определиться с набором ролей серверов или приложений, использующих службу каталогов. Например, для роли файлового сервера используется сервис Samba. Наиболее часто используемые роли сервера: файловый сервер, прокси-сервер, почтовый сервер, сервер аутентификации. Как правило, используют следующее соответствие:

- файловый сервер – сервис Samba;
- прокси-сервер – сервис squid;
- почтовый сервер – сервисы postfix и dovecot;
- сервер аутентификации – PAM.

Данное соответствие позволит далее четко определить атрибуты записей, соответствующие выбранному набору сервисов.

3. Определить частоту использования сервисом службы каталогов.

Следует определить, как часто сервис использует службу каталогов: если сервис редко использует службу каталогов, его не следует выбирать как условие группирования записей. Данный параметр следует определять по журналам работы службы каталогов в режиме отладки.

4. Определить типы запросов сервисов к службе каталогов.

Так как для служб каталогов операции чтения преобладают над операциями записи [8], следует отдавать предпочтение сервисам, где запросы чтения преобладают над запросами добавления или изменения. Данный параметр следует определять по журналам работы службы каталогов в режиме отладки.

5. Определить, эффективно ли работает служба каталогов с сервисом.

Для определения приемлемого значения времени отклика на запрос для человеко-компьютерного взаимодействия (а человек – основной пользователь службы каталогов) предлагается использовать результаты текущих исследований по данной тематике [9–11], на основании которых можно сделать вывод, что для определения эффективности работы службы каталогов следует использовать следующие параметры:

- ожидаемое время поиска записи T в условиях сложности задачи;
- среднеквадратическое отклонение n результатов времени поиска записи каталога t_i при одинаковых параметрах поиска [12].

Общие условия эффективности работы службы каталогов могут быть сформулированы как

$$\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(t_i-\frac{1}{n}\sum_{i=1}^{n}t_i\right)^2}\le Tv; \quad (1)$$

$$\frac{1}{n}\sum_{i=1}^{n}t_i\le T, \quad (2)$$

где Tv – допустимая величина вариации результатов времени отклика на запрос [13].

При невыполнении условий (1) и (2) можно судить о том, что служба каталогов неэффективно работает с заданным сервисом.

6. Определить условия группирования записей.

Следует сформулировать окончательный набор условий группирования записей для выбранной ветки каталога на основании результатов выполнения шагов 1–5.

7. Определить атрибуты записей, соответствующие выбранному набору условий.

Сервисы или внутренние приложения организации, работающие со службой каталогов, используют определенный набор атрибутов, идентифицирующий запись или набор записей. Некоторые сервисы используют атрибуты, предоставляемые разработчиками реализации служб каталогов, а некоторые, например файловый сервер Samba, используют специальные наборы атрибутов. Перед определением множеств записей сервисов каталога с учетом приоритета их использования следует определить минимально необходимые множества атрибутов для каждого сервиса.

Множества атрибутов записей, запрашиваемые сервисами, определяются одним из двух возможных способов:

- анализом журналов службы каталогов в режиме отладки;
- анализом документации разработчика сервиса.

8. Определить множества записей каталога для переноса с учетом приоритета их использования.

Для построения предлагаемой структуры каталога (сетевой) необходимо учитывать критерии принадлежности записи к определенному сервису. Одна запись может использоваться сразу несколькими сервисами, при этом запись следует располагать в ветке сервиса с наибольшим приоритетом.

Для определения множеств записей каталога для переноса с учетом приоритета их использования предлагается использовать алгоритм разбиения множества параметров

каталога, хранящихся в службе каталогов, на подмножества с учетом приоритета их использования с целью уменьшения времени отклика на запрос [14].

В структуре хранения данных данные влияют на временные характеристики служб каталогов, поэтому записи ка-талога можно рассматривать как параметр каталога наравне с аппаратными параметрами сервера, обслуживающего службу. В данном алгоритме под термином «параметр» понимается запись каталога.

9. Сформировать структуру хранения данных службы каталогов с учетом приоритетов использования параметров службы каталогов.

Определив все параметры каталога, требующие переноса в заданные ветки сервисов, предлагается использовать алгоритм перехода от иерархической структуры хранения данных без учета приоритетов использования параметров к структуре с учетом приоритетов использования параметров [14].

Предложенная последовательность шагов была применена к службе каталогов, использующей стандартную иерархическую структуру хранения данных и работающей с двумя сервисами: файловым сервером и прокси-сервером. Изначальная база данных записей каталога формировалась случайным образом записями трех видов:

- содержащими атрибуты только файлового сервера;
- содержащими атрибуты только прокси-сервера;
- содержащими атрибуты обоих сервисов.

Все записи располагались в одной ветке каталога, общее количество записей составило 150 000. Служба каталогов была сконфигурирована без использования индексирования и кэширования для минимизации вариации результатов времени отклика на запрос поиска. При добавлении 10 000 записей для каждого сервиса создавались запросы поиска соответствующего сервису атрибута записи, где фильтр поиска был сгенерирован случайным образом атрибутом uid и атрибутом объектного класса сервиса.

Количество записей по принадлежности к виду:

- записи, содержащие атрибуты только файлового сервера, – 77 171;
- записи, содержащие атрибуты только прокси-сервера, – 37 458;
- записи, содержащие атрибуты обоих сервисов, – 37 315.

После применения предложенного метода проектирования структуры хранения данных службы каталогов записи были сгруппированы и перераспределены по двум веткам каталога, соответствующим сервисам, где наибольший приоритет был отдан записям файлового сервера. Для полученной сетевой структуры хранения данных создавались аналогичные запросы поиска, соответствующего сервису атрибута записи, где фильтр поиска был сгенерированным случайным образом атрибутом uid и атрибутом объектного класса сервиса.

Результаты средних значений времени отклика на запрос поиска значения атрибута записи файлового сервера для обеих структур изображены на рисунке 3, результаты средних значений времени отклика на запрос поиска значения атрибута записи прокси-сервера для обеих структур – на рисунке 4.

По полученным результатам видно, что время отклика на запрос поиска значения атрибута записи для обоих сервисов уменьшилось. Время отклика на запрос поиска значения атрибута прокси-сервера меньше, чем соответствующее значение для файлового сервера.

В первую очередь это обусловлено количеством записей файлового сервера и прокси-сервера, так как общее количество записей прокси-сервера меньше, чем общее количество записей файлового сервера: 74 773 и 112 542 соответственно.

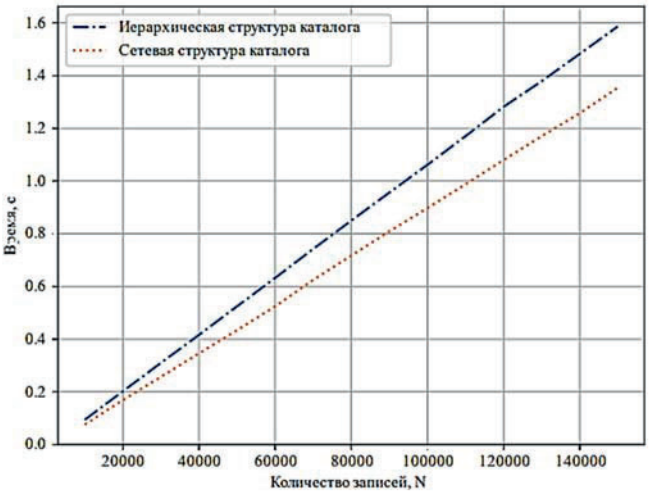

Рис. 3. Зависимость времени отклика на запрос поиска значения атрибута записи файлового сервера для обеих структур

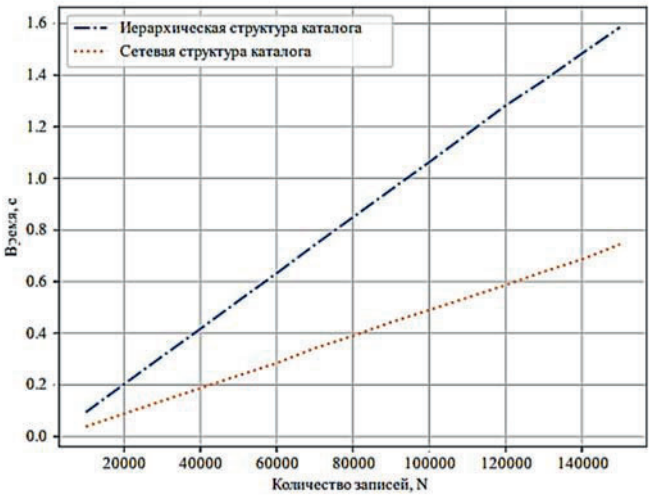

Рис. 4. Зависимость времени отклика на запрос поиска значения атрибута записи прокси-сервера для обеих структур

Во-вторых, записи в ветке прокси-сервера, а именно записи прокси-сервера и записи-ссылки, меньше по размеру, чем записи в ветке файлового сервера, что также влияет на время отклика на запрос поиска значения атрибута.

Заключение

Таким образом, применением данного метода достигается уменьшение изначального количества записей каталога для заданного сервиса, использующего службу каталогов, соответственно уменьшается и время поиска записи сервиса. Распределение записей с учетом их приоритета, формируемое в результате применения метода, позволяет задействовать различные дополнительные методы оптимизации службы каталогов для различных сервисов и веток каталога.

Данный метод проектирования структуры хранения данных службы каталогов имеет достоинства и недостатки.

К достоинствам такого способа проектирования и построения структуры хранения данных службы каталогов относятся следующие:

- при одинаковых условиях запроса время отклика при запросе меньше по сравнению со стандартной иерархической структурой;
- возможность применения различных методов оптимизации для разных групп записей.

К недостаткам такого способа проектирования и построения структуры хранения данных службы каталогов относятся следующие:

- увеличение общего количества записей каталога и увеличение дискового пространства, требуемого для хранения данных;
- сложность структуры.

Литература

1. Butcher M. Mastering OpenLDAP: Configuring, Securing and Integrating Directory Services, Packt Publishing, 2007. – 484 p.

2. Howes T.A., Smith M.C., Good G.S. Understanding and Deploying LDAP Directory Services, 2nd edition, Addison-Wesley Professional, 2003. – 936 p.

3. OpenLDAP Software 2.4 Administrator`s Guide: Tuning. – URL: http://www.openldap.org/doc/admin24/tuning.html (дата обращения 20.05.2019).

4. Андреев А.В. Математическая модель службы каталогов // Информационно-управляющие системы. – 2014. – № 4 (71). С. 85–87.

5. Гордеев А. В., Андреев А. В. Структура каталога как фильтр поиска записей службы каталогов OpenLDAP // Информационно-управляющие системы. – 2019. – № 2 (99). – С. 52–56.

6. RFC 4512 – Lightweight Directory Access Protocol (LDAP): Directory Information Models. – URL: http://tools.ietf.org/html/rfc4512 (дата обращения 20.05.2019).

7. RFC 4511 – Lightweight Directory Access Protocol (LDAP): The Protocol. – URL: http://tools.ietf.org/html/rfc4511 (дата обращения 20.05.2019).

8. Glenn W., Simpson M.T. MSCE Self-Paced Training Kit (Exam 70-297): Designing a Microsoft Windows Server 2003 Active Directory and Network Infrastructure, Redmond, Washington, Microsoft Press, 2004. – 459 p.

9. Miller R. B. Response Time in Man-Computer Conversational Transactions, *AFIPS Conference Proceedings*, 1968, Vol. 33, Pt. 1. – Pp. 267–277.

10. Shneiderman B. Designing the user interface: Strategies for effective human-computer interaction, 1st edition, Reading, MA, Addison-Wesley, 1987. – 448 p.

11. Seow S.C. Designing and Engineering Time: The Psychology of Time Perception in Software, Addison-Wesley Professional, 2008, 224 p.

12. Jain R.K. The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, New York, NY, John Wiley & Sons, 1991. – 720 p.

13. Stupak N. Time delays and system response times in human-computer interaction (2009). Thesis. Rochester Institute of Technology, 102 p. – URL: http://scholarworks.rit.edu/theses/1370.

14. Андреев А.В. Сетевая модель данных службы каталогов // Труды МФТИ. – 2014. – Т. 6, № 3 (23). – С. 27–36.

A Method for Designing a Directory Service Data Storage Structure to Reduce Response Time When Querying for Large Amounts of Data

Andreyev A.V.
Saint Petersburg State University of Aerospace Instrumentation
Saint Petersburg, Russia
aathexf@gmail.com

Abstract. **A method for designing the directory service data structure to reduce the directory services`s response time during requesting a large amount of data is proposed. This article discusses the LDAP directory services and network data structure, where directory entries are stored in the appropriate directory branches for each given service using the directory service. Entries are grouped by service priority conditions. True fulfillment of several conditions for a single record is implemented by the alias entries that conform to the LDAP standard.**

Keywords: **directory service, LDAP, response time, method, directory structure**

REFERENCES

1. Butcher M. Mastering OpenLDAP: Configuring, Securing and Integrating Directory Services, Packt Publishing, 2007. – 484 p.

2. Howes T.A., Smith M.C., Good G.S. Understanding and Deploying LDAP Directory Services, 2nd edition, Addison-Wesley Professional, 2003. – 936 p.

3. OpenLDAP Software 2.4 Administrator`s Guide: Tuning. Available at: http://www.openldap.org/doc/admin24/tuning.html (accessed 20.05.2019).

4. Andreyev A. V. Mathematical Model of a Directory Service [Matematicheskaya model' sluzhby katalogov], *Information and Control Systems [Informatsionno-upravliaiushchie sistemy]*, 2014, No. 4 (71). – Pp. 85–87.

5. Gordeyev A.V., Andreyev A.V. OpenLDAP Directory Service Structure as a Search Filter [Struktura kataloga kak fil'tr poiska zapisey sluzhby katalogov OpenLDAP], *Information and Control Systems [Informatsionno-upravliaiushchie sistemy]*, 2019, No. 2 (99). – Pp. 52–56.

6. RFC 4512 – Lightweight Directory Access Protocol (LDAP): Directory Information Models. Available at: http://tools.ietf.org/html/rfc4512 (accessed 20.05.2019).

7. RFC 4511 – Lightweight Directory Access Protocol (LDAP): The Protocol. Available at: http://tools.ietf.org/html/rfc4511 (accessed 20.05.2019).

8. Glenn W., Simpson M.T. MSCE Self-Paced Training Kit (Exam 70-297): Designing a Microsoft Windows Server 2003 Active Directory and Network Infrastructure, Redmond, Washington, Microsoft Press, 2004. – 459 p.

9. Miller R. B. Response Time in Man-Computer Conversational Transactions, *AFIPS Conference Proceedings*, 1968, Vol. 33, Pt. 1, Pp. 267–277.

10. Shneiderman B. Designing the user interface: Strategies for effective human-computer interaction, 1st edition, Reading, MA, Addison-Wesley, 1987, 448 p.

11. Seow S.C. Designing and Engineering Time: The Psychology of Time Perception in Software, Addison-Wesley Professional, 2008. – 224 p.

12. Jain R.K. The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, New York, NY, John Wiley & Sons, 1991. – 720 p.

13. Stupak N. Time delays and system response times in human-computer interaction (2009). Thesis. Rochester Institute of Technology. – 102 p. – URL: http://scholarworks.rit.edu/theses/1370.

14. Andreyev A.V. Network Data Model of Directory Services [Setevaya model' dannykh sluzhby katalogov], *Proceedings of Moscow Institute of Physics and Technology [Trudy MFTI]*, 2014, Vol. 6, No. 3 (23). – Pp. 27–36.

Алгоритмы автоматической визуализации 3D-моделей наноструктур для использования в программах расчета физических свойств наноматериалов методами молекулярной динамики

Канд. техн. наук А.М. Барановский, Ю.Л. Ермолаев
АО «НИИ программных средств»
Санкт-Петербург, Россия
bamvka@mail.ru, infolab@list.ru

Аннотация. **Представлены алгоритмы и программные решения задач визуализации 3D-моделей наноструктур, обеспечивающие автоматическое построение практически значимых наноструктур – графенов, углеродных нанотрубок, фуллеренов и др. – с учетом их множественных повторяющихся подструктур и поддерживающие пространственные операции над элементами 3D-сцены, визуальное «конструирование» наноструктур, моделирование различных деформаций и характерных дефектов. Алгоритмы реализованы в составе программного комплекса для расчета физических свойств наноматериалов и нанокомпозитов в целях исследования новых материалов и технологий для нужд космической отрасли, могут использоваться как автономно, так и совместно с программами моделирования методами молекулярной динамики.**

Ключевые слова: **алгоритм, трехмерная визуализация, программный комплекс, наноструктура, графен, углеродная нанотрубка, фуллерен, деформация наноструктуры, дефект наноструктуры, молекулярная динамика.**

Введение

Актуальность задач определения физических свойств наноматериалов и нанокомпозитов определяется растущими потребностями современных технологий, в частности исследованиями в целях создания новых перспективных материалов для транспортных систем и космической техники. Важными примерами таких материалов являются графен и структуры на его основе – углеродные нанотрубки (УНТ), фуллерены, а также основанные на них нанокомпозиты и др. [1–3].

Одним из основных современных подходов, применяемых при получении расчетных значений физических свойств различных веществ, является метод молекулярной динамики (ММД) [4]. В процессе моделирования этим методом для понимания физического смысла задаваемых параметров и анализа результатов расчета крайне желательно выполнять пространственную визуализацию соответствующих атомных, молекулярных или наноразмерных структур. Алгоритмы решения задачи визуализации, будучи в принципе согласованы с алгоритмами самого физического моделирования, обладают достаточной самостоятельностью, чтобы их программную реализацию имело смысл выделять в автономные программные модули или отдельные приложения. Такие приложения могут работать совместно с различными программами ММД либо использоваться самостоятельно для иллюстративных и учебных целей.

В настоящее время известен ряд автономных программных приложений визуализации, выполняющих подобные задачи, в частности Avogadro [5], RasMol [6], NanoXplorer [7] и др. Существует также ряд вычислительных пакетов, в которых есть встроенные средства визуализации атомных и молекулярных структур. В качестве примера можно указать Chem3D [8]. Однако подобные разработки имеют определенные недостатки:

- они ориентированы в основном на визуализацию сложных молекул, собираемых в пошаговом режиме из отдельных атомов, и их прямое использование для наноструктур и нанокомпозитов затруднительно, так как в них, как правило, отсутствуют развитые средства автоматического построения 3D-моделей конкретных практически значимых наноструктур – графенов, УНТ, фуллеренов и т. п. с учетом их множественных повторяющихся подструктур;
- в них не поддерживаются или имеют существенные ограничения пространственные операции над элементами 3D-сцены – вращение относительно выбираемых локальных осей, масштабирование, управление параметрами виртуальной «камеры» (фокусное расстояние, тип перспективы), выделение и независимое перемещение отдельных подструктур, их соединение с учетом локальных контактных потенциалов межатомного взаимодействия, т. е. визуальное «конструирование» наноструктур, а также моделирование различных деформаций, характерных дефектов и т. п.

Работа выполнена по договору № 2581/НИИПС с Научно-исследовательским институтом космических систем имени А.А. Максимова – филиалом АО «Государственный космический научно-производственный центр имени М.В. Хруничева» на выполнение составной части научно-исследовательской работы «Технология СГ-3.3.4.1» для государственных нужд в рамках программы Союзного государства «Технология СГ».

Настоящая работа посвящена описанию алгоритмов и программных решений, предназначенных для задач автоматической визуализации 3D-моделей наноструктур, позволяющих преодолеть указанные недостатки прототипов. Алгоритмы реализованы в составе разрабатываемого программного комплекса для расчета физических свойств наноматериалов и нанокомпозитов, в том числе на основе ММД, в целях исследования свойств новых материалов и технологий для нужд космической отрасли. Они также могут быть выделены в автономное приложение для самостоятельного использования и интеграции с другими алгоритмами и программами расчета физических свойств наноструктур.

Общая концепция программного решения

Для визуального представления различных молекулярных соединений и наноструктур в 3D стандартом де-факто стало использование так называемых «шаро-стержневых моделей». Они встречаются в различных вариантах, но общим является то, что для изображения атомов используются 3D-модели сфер различных радиусов и расцветки, а для изображений межатомных связей (главным образом ковалентных) – 3D-модели тонких цилиндров, тоже имеющих различную окраску поверхности в зависимости от принятых условных обозначений. Для большинства приложений в рассматриваемой области этого вполне достаточно, т. к. позволяет и наглядно представить саму структуру, и показать некоторые преобразования ее топологии (деформации, разрывы и пересоединения связей, контакты обособленных структур и проч.). Широко используется задание различных значений цвета и радиусов «шаров» и «стержней», а также анимация 3D-сцены.

В разрабатываемом программном комплексе Graphene, а также автономном программном модуле NanoDesigner последовательно использован объектно-ориентированный подход (ООП) к программной реализации подобных моделей на языке C++. Их пространственные характеристики и топология учитываются путем хранения соответствующих параметров как ООП-свойств невизуальных объектов, описывающих «шары», «стержни», а также построенные из них (в том числе повторяющиеся) подструктуры и устроенные иерархически более сложные комбинированные структуры. Преобразования, описывающие процессы соединения, деформации, трансляции, поворотов и прочих изменений, представлены ООП-методами объектов соответствующего уровня иерархии. Координаты атомов хранятся в объектах-»шарах» и могут синхронно изменяться при моделировании деформаций, а в объектах-»стержнях» хранятся только указатели на те объекты-»шары», которые они соединяют. Таким способом моделируются связи между атомами, сохраняющиеся для широкого класса неразрушающих – при соблюдении определенных условий – деформаций. Группа невизуальных классов, отвечающая за все это ООП-представление модели, выделена в специализированную библиотеку NDLib. В ней применяется ООП-принцип полиморфизма, позволяющий манипулировать множеством объектов производных классов через указатели на объекты базовых классов, хранимых в контейнерах, и выполнять единым образом специфические преобразования данных, реализованные в каждом производном классе по-своему.

Основная часть иерархии невизуальных классов библиотеки NDLib приведена на рисунке 1. В абстрактном базовом классе наноструктур ND_BaseStruct реализованы только общие для всех типов наноструктур свойства и описаны чисто виртуальные методы. В производном классе ND_GrapheneStruct, обеспечивающем моделирование наноструктуры листа графена произвольного типа и размеров, имеется контейнер указателей на подобъекты класса ND_GrapheneBand, представляющие собой наноленты – цепочки ячеек графена, каждая из которых, в свою очередь, тоже хранится в контейнере и реализована в объекте класса ND_GrapheneCell. Аналогично моделируются УНТ: за них отвечает класс NT_NanotubeStruct, который содержит контейнер подобъектов ND_GrapheneRing, представляющих собой отдельные нанокольца – замкнутые цепочки ячеек графена, представленных теми же объектами класса ND_GrapheneCell. Похожим образом реализованы и другие типы наноструктур: класс ND_MultiLayerStruct обеспечивает моделирование многослойных наноструктур, состоящих из тех же составных элементов; класс ND_FullereneStruct – фуллеренов, использующих модифицированные ячейки ND_GrapheneCell; класс ND_CompositeStruct – нанокомпозитов, содержащих дополнительно высокомолекулярные структуры полимерной матрицы; класс ND_MolecularStruct – построение молекулярных структур, заданных символическим описанием на языке nanoML [7] (используемом в программе NanoXplorer) на основе нотации SMILES [9] и других способов описания сложных молекул; наконец, класс ND_Constructor обеспечивает ряд операций над готовыми моделями, позволяющих выполнять их визуальное конструирование и изучать контактные взаимодействия различных наноструктур.

Рендеринг 3D-сцен, состоящих из множества подобных объектов, может быть осуществлен при помощи самых разных внешних подсистем и библиотек 3D-визуализации и анимации, выбор которых практически не зависит от программной реализации алгоритмов построения и трансформации самих структур. В частности, в данной работе использовались свободно распространяемые библиотеки: универсальная Qt и ее расширение для научно-технических приложений QSE [10–12]; при этом для рендеринга окончательно сформированных 3D-моделей наноструктур оказалось вполне достаточно возможностей входящего в состав библиотеки Qt модуля Qt3D, в котором имеются готовые визуальные классы QSphereMesh для сфер и QCylinderMesh для цилиндров, отображаемые при помощи моделей освещения Гуро и Фонга. При формировании 3D-сцены на основе объектов модуля Qt3D в нее из невизуальных объектов NDLib, описывающих топологию созданной наноструктуры, передаются координаты, размеры и цвета моделей-»шаров» для формирования сфер QSphereMesh, а объекты-»стержни», хранящие указатели на соединяемые ими «шары», используются для вычисления координат центров торцевых окружностей – направляющих цилиндров QcylinderMesh – с одновременной передачей их радиусов и цветов. Для итоговой 3D-сцены выполняется рендеринг (включающий возможность интерактивного управления направлением просмотра, вращением, сдвигом, масштабированием, характеристиками виртуальной «камеры» и фонового освещения) средствами модуля Qt3D, результат отображается при помощи объекта класса QSE_3DViewWin библиотеки QSE.

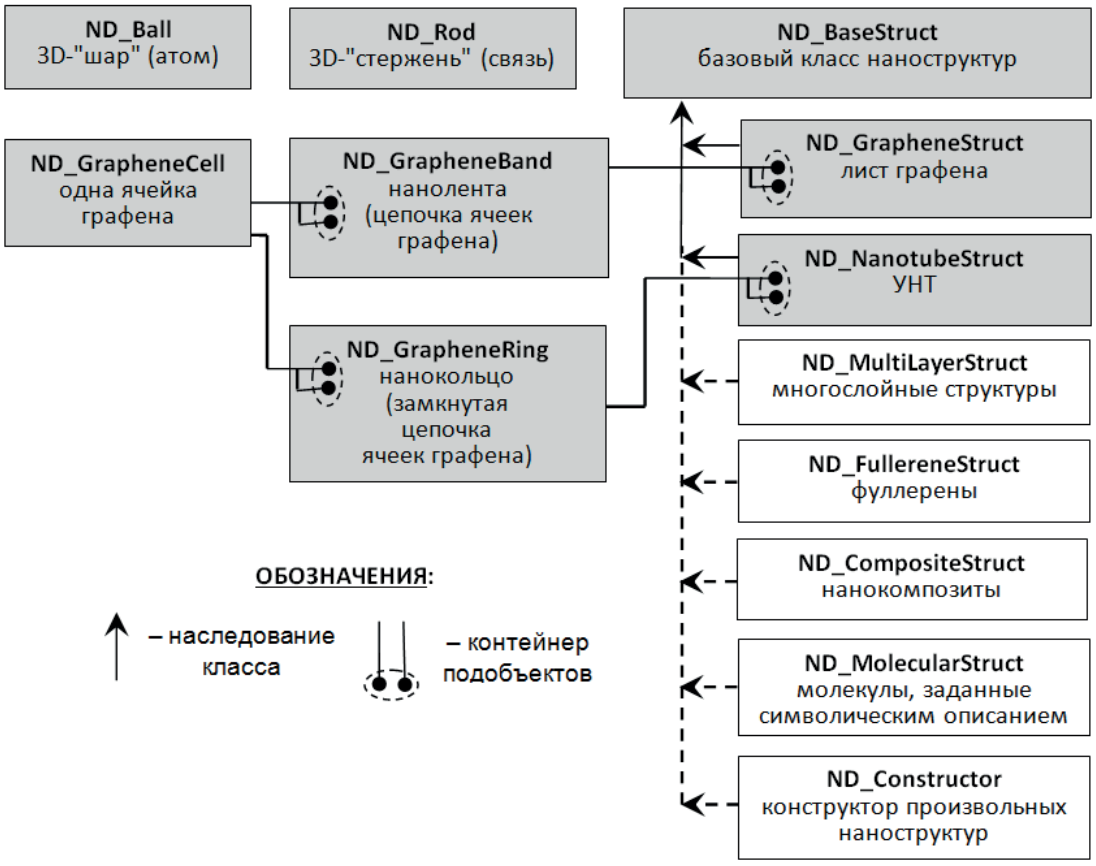

Рис. 1. Часть иерархии классов библиотеки NDLib

Графическое окно 3D-просмотра, формируемое этим объектом, средствами библиотеки QSE может быть встроено в любое место главного окна приложения, так же как и отдельная форма с панелью его настроек. Если для визуализации готовой 3D-сцены потребуется более высокое качество графики или анимации либо специальные эффекты, модуль рендеринга может быть заменен на другое аналогичное решение, например на более развитую библиотеку OpenSceneGraph [13], без переделки классов NDLib.

Для удобного начального расположения 3D-наноструктуры в графическом окне выполняется операция определения координат ее габаритного параллелепипеда, после чего масштаб изображения и параметры виртуальной «камеры» подстраиваются таким образом, чтобы наноструктура располагалась в центре окна, занимая его большую часть.

Визуализация наноструктур является только одной из ряда задач, реализованных в программном комплексе Graphene, вместе с моделями расчета на основе ММД и другими физическими моделями для углеродных наноструктур. На рисунке 2 приведен общий вид главного окна программы. В левой верхней части присутствует навигационный элемент – «дерево» задач и моделей, под ним – одна из динамически сменяемых панелей управления, выбор которых синхронизирован с перемещением по «дереву» (эта функциональность обеспечивается средствами библиотеки QSE). В правой части рисунка в одном из окон просмотра представлена готовая 3D-сцена с примером моделирования одной из конкретных наноструктур (в данном случае – листа графена при наличии сдвиговой деформации). Окно автономного приложения NanoDesigner выглядит аналогично, но в нем отсутствуют те ветви «дерева», которые относятся к остальным физическим задачам и моделям комплекса, тогда как средства управления построением наноструктур и просмотром 3D-изображений – те же самые.

Возможность использования разработанного программного модуля комплекса Graphene, а также автономного приложения NanoDesigner не только для визуализации наноструктур, но и для подготовки процесса моделирования их физических свойств, обеспечивается тем, что предусмотрен ввод массива координат атомов и всей связующей информации для заданной наноструктуры из файла и их вывод в файл. Такие файлы могут использоваться для физических расчетов другими модулями комплекса либо внешней программой ММД.

Построение 3D-модели листа графена

Разработанные в данном исследовании алгоритмы автоматического построения 3D-моделей наноструктур основаны на использовании операций множественного клонирования-трансляции повторяющихся подструктур, а также разделении этапов построения топологии модели и перевода ее для рендеринга в 3D-сцену, состоящую из стандартных визуальных объектов.

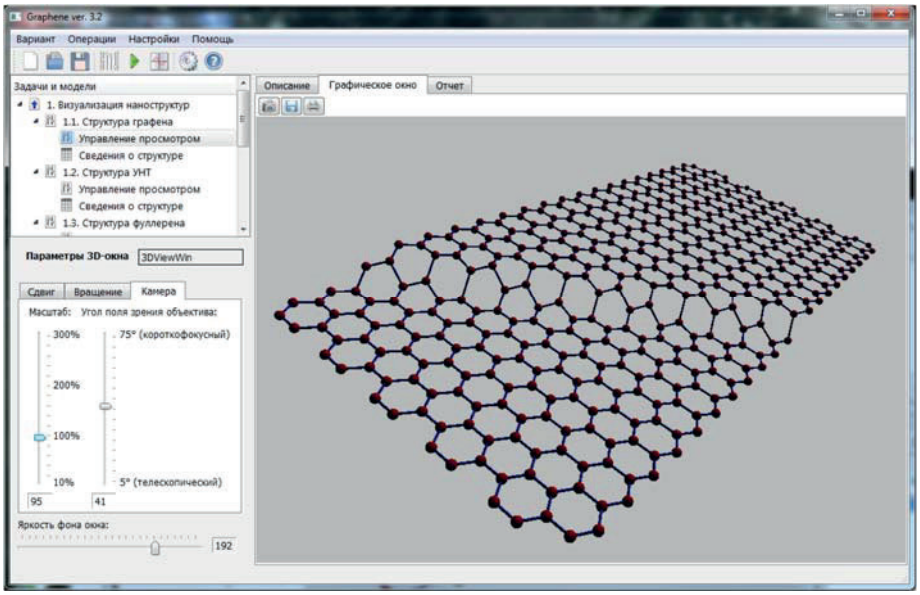

Рис. 2. Главное окно программы с примером наноструктуры листа графена при сдвиговой деформации

Алгоритм моделирования наноструктуры листа графена включает следующие последовательные этапы:

1. Построение объектной «шаро-стержневой» модели правильной гексагональной ячейки графена. Используется локальная декартова система координат; расположение готовой ячейки относительно мировых координат всей структуры может быть в дальнейшем скорректировано без изменения относительных расстояний между атомами ячейки. Координаты атомов ячейки задаются как вершины правильного шестиугольника с использованием масштабных множителей $a_o \approx 0{,}142$ нм (расстояние между ближайшими атомами углерода) и $a = a_o\sqrt{3} \approx 0{,}246$ нм (размер ячейки по диагонали):

{(-0,5a_o, 0,5a); (0,5a_o, 0,5a); (a_o, 0); (0,5a_o, -0,5a); (-0,5a_o, -0,5a); (-a_o, 0)} – для горизонтально расположенной ячейки;

{(-0,5a, -0,5a_o); (-0,5a, 0,5a_o); (0, a_o); (0,5a, 0,5a_o); (0,5a, -0,5a_o); (0, -a_o)} – для вертикальной ячейки.

Координата z всех углов ячейки задается равной 0.

Так формируется объект класса ND_GrapheneCell заданной ориентации. Необходимые модификации вносятся в этот алгоритм при генерации правильных пяти- и семиугольных ячеек, которые необходимы для моделирования некоторых дефектов наноструктур, а также фуллеренов.

2. Построение нанолент с различными значениями параметров хиральности (кромки типа «armchair» и «zigzag» [1, 2], а также нанолент с углами хиральности 0...30° при горизонтальном или вертикальном расположении ячеек). Для создания наноленты первая ячейка клонируется заданное число раз n_x с периодической трансляцией вдоль координаты x в зависимости от заданного типа края наноленты на величину $dx = La$ для кромки типа «zigzag» и $dx = 1{,}5La_o$ для кромки типа «armchair», где L – число уже имеющихся в контейнере ячеек на данном шаге. Каждая созданная ячейка (новый объект класса ND_GrapheneCell) добавляется в динамический контейнер, имеющийся в классе ND_GrapheneBand. Алгоритм множественного клонирования и переноса целых ячеек (а впоследствии – и нанолент) в плоскости обеспечивает обнаружение и фиксацию частичных наложений атомов и связей с целью последующего исключения подобного дублирования перед рендерингом. Для этого используется механизм хэш-таблиц, позволяющий хранить единственные экземпляры уникальных объектов, описывающих атомы и связи. Тождественность объектов проверяется в классе ND_GrapheneCell путем сравнения между собой 3D-координат атомов с использованием средств нечеткой логики для исключения ложных срабатываний из-за ошибок округления. Так формируется объект класса ND_GrapheneBand для наноленты графена заданной длины и типа хиральности.

3. Алгоритм построения листа графена с заданными параметрами заключается в выполнении аналогичного этапу 2 множественного клонирования и трансляции уже целых нанолент вдоль направлений x и y в зависимости от типа края листа заданное число раз n_y на величины: для горизонтальной кромки типа «zigzag» – ($dx = 0$, $dy = 1{,}5Ma_o$), если в контейнере четное число нанолент, и ($dx = 0{,}5a$, $dy = 1{,}5Ma_o$), если нечетное; для горизонтальной кромки типа «armchair» – ($dx = 0$, $dy = Ma_o$), здесь M – число уже имеющихся в контейнере нанолент на данном шаге. При этом также используются хэш-таблицы для исключения дублирования накладывающихся на границах нанолент атомов. Каждая созданная наноленты (объект класса ND_GrapheneBand) добавляется в динамический контейнер нанолент, имеющийся в классе ND_GrapheneStruct. Так формируется объект класса ND_GrapheneStruct для листа графена заданных размеров и типа хиральности.

4. Для подготовки к рендерингу объектных моделей осуществляется проход при помощи итераторов, предоставляемых классами библиотеки Qt, по всем объектам контейнеров с учетом иерархии вложенных подобъектов и, в конечном счете, атомам и связям моделируемой наноструктуры. В итоге для каждого атома создается невизуальный объект класса ND_Ball, а для каждой связи – клас-

са ND_Rod (см. рис. 1), помещаемые в общие линейные контейнеры, не учитывающие топологию наноструктуры. Это позволяет при рендеринге 3D-сцены использовать один и тот же внешний алгоритм для любой наноструктуры, проходящий по всем «шарам» и «стержням» и генерирующий для них соответствующие визуальные 3D-объекты.

Моделирование деформаций поверхности листа графена

В ряде физических задач возникает необходимость построения моделей графеновых листов с деформациями разного рода. Это требуется как для их визуального представления, так и для расчетов механических и других физических характеристик, в частности, при помощи ММД.

В разработанном в данном исследовании алгоритме моделирования деформаций используется локальное изменение координат атомов подготовленной наноструктуры, приводящих к растяжениям-сжатиям связей между ними, при возможности контроля допустимой степени деформации. Плоскость сформированной предварительно модели листа графена (объекта класса ND_GrapheneStruct) совмещается с одной из координатных плоскостей декартовой системы, например *XY*. Алгоритм заключается в проходе итератором по всем атомам иерархической модели наноструктуры и смещении локальной координаты z каждого атома в точке плоскости (x, y) на величину Δz, определяемую значением одной из заданных функций $\Delta z = f(x, y)$, соответствующих разным типам деформаций. При этом задаются также величины предельного сближения соседних атомов (то есть изменения относительных длин связей), при достижении которого дальнейшие изменения z ограничиваются, а также предельного растяжения, при превышении которого соответствующая связь разрывается – удаляется из описания топологии наноструктуры или отображается другим цветом. Это позволяет наглядно представить разрушение наноструктуры при запредельных деформациях. При использовании средств анимации на этапе рендеринга может быть также показана динамика процесса разрушения наноструктуры во времени.

В качестве примеров деформаций поверхности графена реализованы функции $f(x, y)$, моделирующие гармонические волновые процессы (распространение звука):

$$\Delta z(x, y) = D \sin 2\pi \frac{x}{dx} \cos 2\pi \frac{y}{dy};$$

сдвиговые волны деформации (см. рис. 2):

$$\Delta z(x, y) = \begin{cases} D \text{ при } \frac{x}{dx} + \frac{y}{dy} > 1 + s; \\ 0 \text{ при } \frac{x}{dx} + \frac{y}{dy} < 1; \\ \frac{D\left(\frac{x}{dx} + \frac{y}{dy} - 1\right)}{s} \text{ в промежутке;} \end{cases}$$

статические сосредоточенные нагрузки, описываемые функцией Гаусса (см. рис. 3):

$$\Delta z(x, y) = D \exp\left(\frac{-(x - dx)^2 - (y - dy)^2}{2s^2}\right),$$

где D, dx, dy – задаваемые значения амплитуды и пространственных масштабов процесса; s – подбираемый параметр численной аппроксимации.

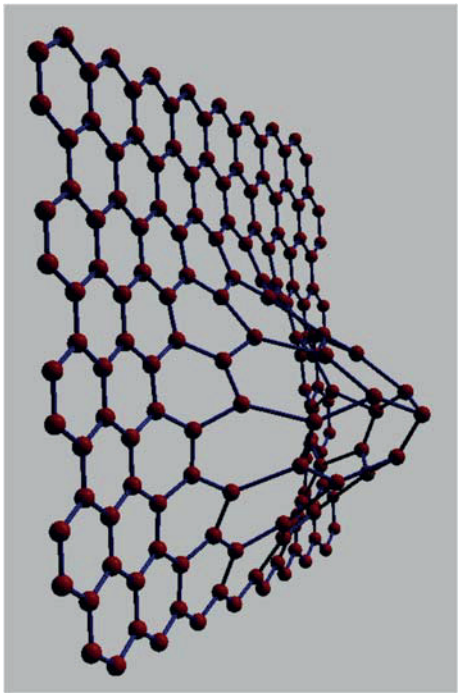

Рис. 3. Пример визуализации деформированной наноструктуры листа графена при сосредоточенной нагрузке

Построение 3D-моделей УНТ и их деформации

Построение однослойной УНТ в целом аналогично построению листа графена, но требует несколько более сложных промежуточных операций.

1. Алгоритм начинается с создания наноленты графена заданной длины (объекта класса ND_GrapheneBand) и с заданным типом края, который будет соответствовать типу края торца будущей УНТ, т. е. с результата, полученного на этапе 2 алгоритма, описанного в предыдущем разделе. Необходимая длина наноленты N_{cell} определяется в зависимости от заданного диаметра будущей УНТ (в единицах целых ячеек графена).

2. Вычисляется радиус окружности, перпендикулярной оси УНТ, которая будет служить направляющей для цилиндрической поверхности:

$$R = \frac{N_{cell} \cdot a}{2\pi},$$

3D-координаты центра этой окружности равны

$$(x_o, y_o, z_o + R),$$

где (x_o, y_o, z_o) – координаты начального угла первой ячейки. Затем эта окружность разбивается на правильный 12-угольник и выполняются последовательные (слева направо) переносы в пространстве части атомов наноленты, при которых каждая ее ячейка поворачивается относительно локальной оси, проходящей через неподвижную левую сторону ячейки так, чтобы ее начальный и конечный углы совмещались с очередной парой углов этого направляющего 12-угольника.

Каждое следующее такое преобразование применяется ко всем оставшимся ячейкам до конца наноленты по отношению к одному повороту предыдущей ячейки. Для каждого типа хиральности используются свои величины поворотов. Так нанолента (объект класса ND_GrapheneBand) постепенно преобразуется в нанокольцо (объект класса ND_GrapheneRing). Накладывающиеся при замыкании нанокольца последние атомы исключаются описанным выше методом (при помощи хэш-таблицы), гарантирующим их уникальность.

3. Выполняется множественное клонирование нанколец (объектов класса ND_GrapheneRing) с их последовательной трансляцией в перпендикулярном направлении вдоль оси формируемой УНТ с исключением накладывающихся атомов при помощи хэш-таблицы до окончания наращивания УНТ до заданной длины. Так формируется объект класса ND_NanotubeStruct для однослойной УНТ заданных размеров и типа хиральности.

4. Подготовка к рендерингу осуществляется способом, совершенно аналогичным описанному для наноструктуры листа графена.

Построение многослойных (соосных) УНТ включает кратное повторение этапов 1–3 с заданием одной и той же оси УНТ для разных диаметров направляющей окружности, пересчитываемых в длины исходных нанолент этапа 1. При этом учитывается, что из-за квантовых ограничений диаметры соседних слоев многослойной УНТ не могут отличаться меньше чем на величину $a/2\pi$. В объектах класса ND_NanotubeStruct предусмотрен контейнер отдельных слоев УНТ, в который они последовательно заносятся. Есть также возможность приписать разные цвета для поверхности «шаров» в каждом слое, чтобы обеспечить их наглядное различение. Последний этап алгоритма – подготовка этой наноструктуры к рендерингу – не имеет принципиальных особенностей. Пример визуализации модели многослойной УНТ приведен на рисунке 4.

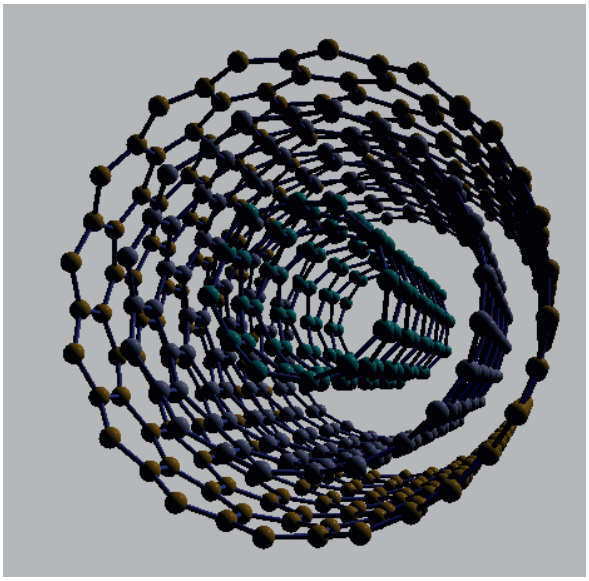

Рис. 4. Пример визуализации модели многослойной УНТ

Для обеспечения возможности моделирования деформаций УНТ требуются некоторые модификации. В исходном алгоритме в каждом объекте сцены хранятся декартовы координаты атомов. Для визуализации деформаций УНТ как целого необходимо перейти к цилиндрической системе координат (r, φ, z), ось z которой совпадает с локальной осью УНТ. Тогда легко получить функции вида $\Delta r = f(z, \varphi)$, описывающие изменения локального

радиуса УНТ r с учетом геометрических особенностей каждого из возможных типов деформаций, в числе которых:

– осевые сжатия и растяжения;

– радиальные симметричные и несимметричные сжатия;

– эйлеровы деформации («гармошка»);

– торсионные деформации («скручивания»).

Несколько более сложные формулы описывают отклонения и изгибы УНТ в целом, не обладающие аксиальной симметрией.

По подобным формулам рассчитываются перемещения каждого атома в составе УНТ, а связи визуализируются по их принадлежности к смещенным атомам (без нарушения структуры). Если локальное растяжение какой-то связи превышает заданный порог, эта связь удаляется или отображается другим цветом, в результате будет показано разрушение УНТ при закритических деформациях. Пример моделирования разрушающей аксиально-симметричной деформации УНТ с параметрами: пространственный масштаб воздействия вдоль оси 0,2 нм, радиальная амплитуда 0,5 нм, максимально допустимое относительное удлинение межатомных связей 10 % – приведен на рисунке 5.

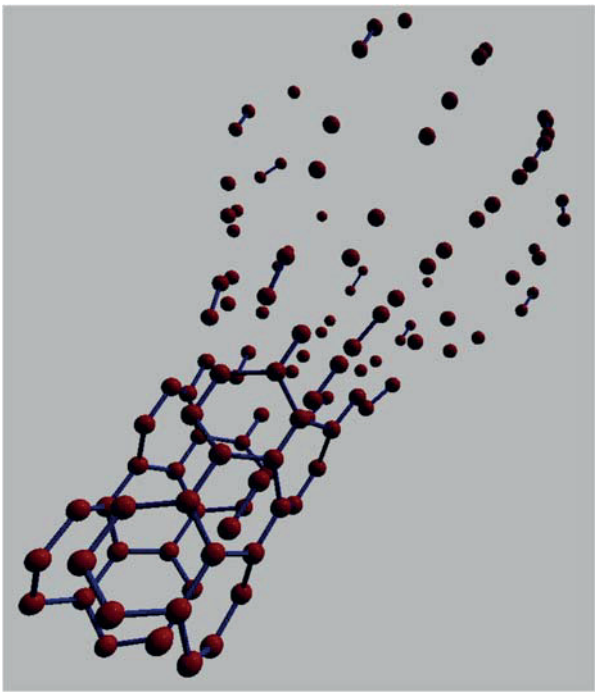

Рис. 5. Пример визуализации модели УНТ при деформации и разрушении

Моделирование других видов наноструктур, их деформаций и дефектов

Сходные, хотя зачастую несколько более громоздкие, алгоритмы могут использоваться для построения моделей фуллеренов, комбинированных и иных наноструктур, содержащих повторяющиеся подструктуры типа ячейки графена.

В фуллеренах атомы углерода расположены в вершинах правильного усеченного икосаэдра. Его построение основано на все тех же базовых алгоритмах множественного клонирования-трансляции простых подструктур с учетом того, что поверхность икосаэдра состоит из чередующихся пяти- и шестиугольных ячеек. Для этого используются соответствующие модификации описанных алгоритмов. Кроме того, построение объекта класса ND_FullereneStruct ведется в сферической системе координат (r, φ, θ), в которой выполняются подобные описанным в предыдущем разделе последовательные операции пространственных переносов части атомов соответствующей ячейки так, чтобы это было эквивалентно ее повороту в целом на нужные углы в меридиональном и широтном направлениях.

Алгоритмы построения моделей нанокомпозитов (класс ND_CompositeStruct) включают два отдельных самостоятельных этапа: а) построение модели полимерной матрицы, для которой используются общие методы символического описания молекулярных структур при помощи нотации SMILES или на языке nanoML [7, 9], реализованные в классе ND_MolecularStruct; б) применение одного из описанных выше алгоритмов построения наноструктур для выбранного типа наночастиц наполнителя – фрагментов листов графена, УНТ, фуллеренов и др. При этом в классе ND_CompositeStruct предусмотрены средства, позволяющие задавать величину относительной объемной доли наполнителя в полимерной матрице и распределение вероятностей случайного пространственного расположения и ориентации наночастиц. Обнаружение наложений атомов наночастиц и матрицы осуществляется при помощи их общей хэш-таблицы; для исправления этого нежелательного явления выполняются малые перемещения наночастицы в целом относительно матрицы перед добавлением следующей наночастицы на минимально допустимое расстояние, определяемое видом используемого потенциала межатомного взаимодействия. Рендеринг итого-

вой композитной структуры осуществляется описанным способом в едином процессе для всех атомов, однако соответствующая процедура требует значительных вычислительных ресурсов и ее результат может быть представлен только на крупноформатных дисплеях высокого разрешения ввиду значительного (более 10^4) числа участвующих атомов и связей.

Другие наноструктуры и в еще большей степени наноустройства, имеющие менее стандартную и регулярную топологию, могут строиться в «ручном режиме», поддерживаемом классом ND_Constructor. Он позволяет создавать сложные комбинации из описанных выше готовых элементов, представленных в «галерее», с возможностью выбора их типа, размеров, ориентации и помещения в произвольное место конструируемой модели. При этом учитываются статические характеристики допустимого расположения атомов, обусловленные выбранным видом потенциала межатомного взаимодействия, для известных равновесных конфигураций, определяемых локальными минимумами потенциала, даже без полного расчета при помощи ММД. В классах библиотеки NDLib заданы хорошо исследованные классические модельные потенциалы Леннард-Джонса и его модификации, а также потенциал Морзе, отличающийся тем, что получающиеся с его помощью упругие механические свойства наноструктур близки к экспериментально измеренным [10].

К описанным задачам примыкают задачи моделирования ряда наблюдавшихся пространственных дефектов наноструктур [14, 15], в частности:

– типичных для графена дефектов Стоуна–Уэйлса: пересоединения углеродных связей с формированием двух 5-угольных и двух 7-угольных ячеек;

– сворачивания атомной плоскости графена в конус при наличии 5-угольных ячеек (структура с 12 такими дефектами эквивалентна фуллерену);

– образование седловидных искривлений атомной плоскости графена в присутствии 7-угольных ячеек;

– комбинаций этих дефектов с нормальными ячейками, приводящих к образованию различных искажений формы поверхности графена.

При соответствующих модификациях те же алгоритмы применимы для моделирования всех таких видов дефектов, позволяя как наглядно их визуализировать, так и предоставить массив вычисленных координат атомов с учетом связей между ними для последующих расчетов при помощи ММД, что дает возможность исследовать влияние дефектов на физические (механические, тепловые и др.) свойства наноматериалов.

Заключение

Описанные алгоритмы, реализованные в составе программного комплекса Graphene, используются для визуализации разнообразных наноструктур и расчета их физических свойств (при передаче результирующих массивов координат атомов наноструктуры в модуль ММД). Их адекватность и эффективность проверена в ходе исследований экспериментального образца комплекса путем сравнений с результатами, полученными при помощи других программ – прототипов и аналогов, а также данными экспериментальных исследований структуры и свойств аналогичных наноматериалов, публиковавшимися в научно-технической литературе.

Из комплекса выделено автономное приложение NanoDesigner, предназначенное только для построения и визуализации наноструктур, а также моделирования деформаций и дефектов с учетом задаваемых ограничений. Это приложение может использоваться совместно с самыми разными вычислительными программами, реализующими ММД или другие методы расчета свойств наноматериалов, использующими данные об их топологии.

Представляется перспективным практическое применение разработанных программных средств в ходе расчетно-теоретических исследований тепловых, электромагнитных, оптических и механических свойств наноматериалов, а также визуализации произвольных молекул и химических соединений, имеющих стандартные описания или сконструированных в режиме «ручного» 3D-редактирования. Предполагается дальнейшее развитие описанных алгоритмов и программ и их внедрение на предприятиях госкорпорации «Роскосмос», а также в других заинтересованных организациях.

Литература

1. Гусев А.И. Наноматериалы, наноструктуры, нанотехнологии / А.И. Гусев. – 2-е изд., испр. – М. : Физматлит, 2007. – 414 с.

2. Елецкий А.В. Углеродные нанотрубки / А.В. Елецкий // Успехи физических наук. – 1997. – Т. 167, № 9. – С. 945–972.

3. Созыкин С.А. Компьютерное моделирование нанокомпозитов : учебное пособие / С.А. Созыкин, А.Г. Воронцов, Г.П. Вяткин. – Челябинск : Издательский центр ЮУрГУ, 2015. – 67 с.

4. Ибрагимов И.М. Основы компьютерного моделирования наносистем : учебное пособие / И.М. Ибрагимов, А.Н. Ковшов, Ю.Ф. Назаров. – СПб. : Лань, 2010. – 384 с.

5. Hanwell M.D. et al. Avogadro: An Advanced Semantic Chemical Editor, Visualization, and Analysis Platform, *Journal of Cheminformatics*, 2012, Vol. 4, No. 17 (published 13 August 2012). Available at: http://jcheminf.biomedcentral.com/articles/10.1186/1758-2946-4-17.

6. Bernstein H.J. Recent Changes to RasMol, Recombining the Variants, *Trends in Biochemical Sciences (TIBS)*, 2000, Vol. 25, No. 9. – Pp. 453–455.

7. Рогов В.А. Технология конструкционных материалов. Нанотехнологии : учебник для вузов / В.А. Рогов. – 2-е изд., перераб. и доп. – М.: Юрайт, 2019. – 190 с. – (Авторский учебник).

8. Полещук О.Х. Методические указания к лабораторным работам по компьютерному моделированию химических реакций / О.Х. Полещук, Д.М. Кижнер ; под ред. В.Д. Филимонова. – Томск : Изд-во ТГПУ, 2007. – 160 с.

9. Weininger D. SMILES, a Chemical Language and Information System, *Journal of Chemical Information and Computer Sciences*, 1988, No. 28. – Pp. 31–36.

10. Саммерфилд М. Qt. Профессиональное программирование. Разработка кроссплатформенных приложений на C++ = Advanced Qt Programming. Creating Great Software with C++ and Qt4 / М. Саммерфилд ; пер. с англ. А. Слинкина. – СПб. : Символ-Плюс, 2011. – 560 с.

11. Lee Zhi Eng Qt 5 C++ GUI Programming Cookbook, Birmingham, Packt Publishing Ltd., 2016. – 300 p.

12. Шлее М. Qt 5.10. Профессиональное программирование на C++ / М. Шлее. – СПб. : БХВ-Петербург, 2018. – 1072 с. – (В подлиннике).

13. Nordvik T., Harding C. Interactive Geovisualization and Geometric Modelling of 3D Data – A Case Study from the Åknes Rockslide Site, Norway, *In: Ruas A., Gold C. (eds) Headway in Spatial Data Handling, 13th International Symposium on Spatial Data Handling, Montpellier, France, 23–25 July 2008. Lecture Notes in Geoinformation and Cartography*, Springer-Verlag, Berlin, 2008. – Pp. 367–384.

14. Stone A.J., Wales D.J. Theoretical Studies of Icosahedral C_{60} and Some Related Structures, *Chemical Physics Letters*, 1986, Vol. 128, Is. 5–6. – Pp. 501–503.

15. Алексенко А.Г. Графен / А.Г. Алексенко. – М. : БИНОМ. Лаборатория знаний, 2014. – 168 с.

Algorithms for the Automatic Visualization of 3D Models of Nanostructures for Use in Software for Calculating the Physical Properties of Nanomaterials by Molecular Dynamics Methods

A.M. Baranovsky, Yu.L. Ermolaev
Software Research Institute,
St. Petersburg, Russia
bamvka@mail.ru, infolab@list.ru

Abstract. **Algorithms and software solutions to the visualization of 3D models of nanostructures are presented, which ensure the automatic construction of practically significant nanostructures – graphenes, carbon nanotubes, fullerenes, etc., taking into account their multiple repeating substructures, and supporting spatial operations on 3D scene elements, visual "design" nanostructures, modeling of various deformations and typical defects. The algorithms are implemented as part of a software package for calculating the physical properties of nanomaterials and nanocomposites in order to study new materials and technologies for the needs of the space industry, and can be used both independently and in conjunction with molecular dynamics simulation programs.**

Keywords: **algorithm, 3D visualization, software package, nanostructure, graphene, carbon nanotube, fullerene, nanostructure deformation, nanostructure defect, molecular dynamics.**

REFERENCES

1. Gusev A.I. Nanomaterials, nanostructures, nanotechnologies [Nanomaterialy, nanostruktury, nanotekhnologii], Moscow, FIZMATLIT, 2007. – 416 p.

2. Eletsky A.V. Carbon Nanotubes [Uglerodnye nanotrubki], *Advances in Physical Sciences [Uspekhi fizicheskikh nauk]*, 1997, vol. 167, No. 9. – Pp. 945–972.

3. Sozykin S.A., Vorontsov A.G., Vyatkin G.P. Computer simulation of nanocomposites: Study guide [Komp'yuternoe modelirovanie nanokompozitov: Uchebnoe posobie], Chelyabinsk, South Ural State University, 2015. – 67 p.

4. Ibragimov I.M., Kovshov A.N., Nazarov Yu.F. Fundamentals of computer simulation of nanosystems: Study guide [Osnovy komp'yuternogo modelirovaniya nanosistem: Uchebnoe posobie], Saint Petersburg, Lan Publishing House, 2010. – 384 p.

5. Hanwell M.D. et al. Avogadro: An Advanced Semantic Chemical Editor, Visualization, and Analysis Platform, *Journal of Cheminformatics*, 2012, Vol. 4, No. 17 (published 13 August 2012). Available at: http://jcheminf.biomedcentral.com/articles/10.1186/1758-2946-4-17.

6. Bernstein H.J. Recent Changes to RasMol, Recombining the Variants, *Trends in Biochemical Sciences (TIBS)*, 2000, Vol. 25, No. 9. – Pp. 453–455.

7. Rogov V.A. Technology of construction materials. Nanotechnology: Textbook for high schools [Tekhnologiya konstruktsionnykh materialov. Nanotekhnologii: Uchebnik dlya vuzov], Moscow, Urait Publishing House, 2019. – 190 p.

8. Poleshchuk O.Kh., Kizhner D.M. Guidelines for laboratory work on computer simulation of chemical reactions [Metodicheskie ukazaniya k laboratornym rabotam po komp'yuternomu modelirovaniyu khimicheskikh reaktsiy], Tomsk, Tomsk State Pedagogical University, 2007. – 160 p.

9. Weininger D. SMILES, a Chemical Language and Information System, *Journal of Chemical Information and Computer Sciences*, 1988, No. 28. – Pp. 31–36.

10. Summerfield M. Advanced Qt Programming. Creating Great Software with C++ and Qt4, Saint Petersburg, Symbol Plus Publ. House, 2011. – 560 p.

11. Lee Zhi Eng Qt 5 C++ GUI Programming Cookbook, Birmingham, Packt Publishing Ltd., 2016. – 300 p.

12. Shlee M. Qt 5.10. Professional C ++ Programming [Professional'noe programmirovanie na C++], St. Petersburg, BHV-Petersburg, 2018. – 1072 p.

13. Nordvik T., Harding C. Interactive Geovisualization and Geometric Modelling of 3D Data - A Case Study from the Åknes Rockslide Site, Norway, *In: Ruas A., Gold C. (eds) Headway in Spatial Data Handling, 13th International Symposium on Spatial Data Handling, Montpellier, France, 23–25 July 2008. Lecture Notes in Geoinformation and Cartography*, Springer-Verlag, Berlin, 2008. – Pp. 367–384.

14. Stone A.J., Wales D.J. Theoretical Studies of Icosahedral C_{60} and Some Related Structures, *Chemical Physics Letters*, 1986, Vol. 128, Is. 5–6. – Pp. 501–503.

15. Aleksenko A.G. Graphene [Grafen], Moscow, BINOM. Knowledge Lab, 2014. – 168 p.

Автоматизированная система адаптивного управления насосами наливных терминалов на основе логического контроллера

Бакалавр В.И. Васнихин, д-р техн. наук А.Д. Хомоненко, С.И. Носова
Петербургский государственный университет путей сообщения Императора Александра I
Санкт-Петербург, Россия
vasnikhinvictor97@gmail.com, khomon@mail.ru, nosovasi@mail.ru

Аннотация. **Рассматривается разработанная автоматизированная система адаптивного управления насосами. Обоснован выбор математической модели, разработано программное обеспечение в среде разработки SoMachine, разработан графический интерфейс для панели оператора в среде разработки Vijeo Designer Basic, собран имитационный макет, моделирующий работу системы адаптивного управления насосами. Применение пропорционально-интегрально-дифференциального (ПИД) регулятора позволяет адаптивно настраивать систему управления насосами при изменении уставок и внешних воздействий. Плавные изменения параметров регулирования повышают срок службы механических объектов (насосов), гидравлических объектов (труб) благодаря отсутствию гидроударов.**

Ключевые слова: **программируемый логический контроллер, контроллер MODICON M241-TM241CEC24T, адаптивное управление, насос, ПИД-регулятор, автоматизация.**

Введение

В настоящее время существует тенденция автоматизации всех технологических процессов. Для выполнения этой задачи широкое применение получили системы, работающие на основе логических контроллеров. Достоинством таких систем является возможность оперативного проектирования систем автоматизации различных уровней, возможность быстрого изменения параметров технологических процессов, а также возможность наращивания систем без ущерба для функционирования. В соответствии с этим задача исследования логических контроллеров является актуальной для ОАО «РЖД», а также для возможной реализации в учебном процессе ПГУПС и других транспортных вузов. В рамках этой работы проводились исследования возможностей логических контроллеров для создания систем адаптивного управления насосами наливных терминалов.

Программируемый логический контроллер Modicon M241-TM241CEC24T

Программируемый логический контроллер (ПЛК) является вычислительной машиной, поэтому архитектурно практически не отличается от персонального компьютера. Отличие заключается главным образом в отсутствии видеоплаты, средств ручного ввода и дисковой системы. Вместо этого ПЛК имеет блок входов и выходов [1]. Структурная схема контроллера представлена на рисунке 1.

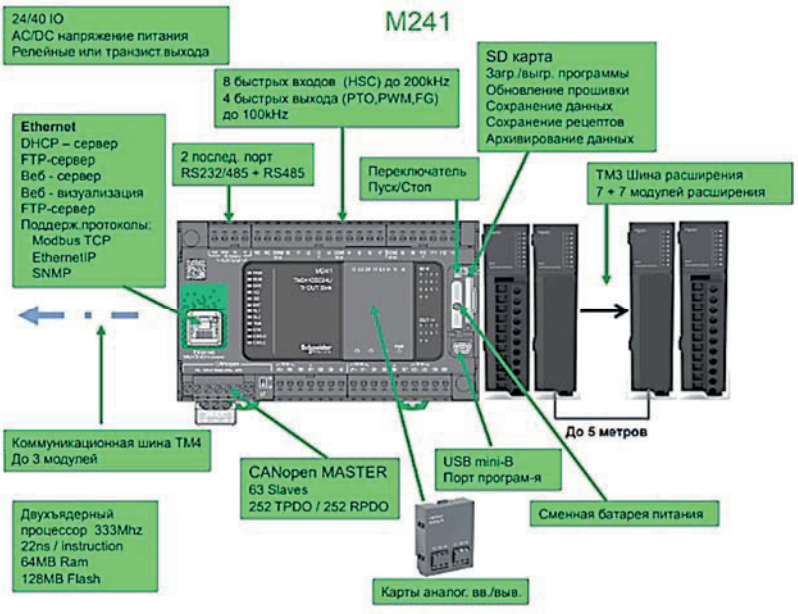

Рис. 1. Структурная схема контроллера

Основные характеристики контроллера MODICON M241-TM241CEC24T:

– *тип устройства:* программируемый логический контроллер;

– *номинальное напряжение сети:* 24 В, постоянный ток;

– *количество дискретных входов:* 14 (+24V), из них 8 быстродействующих входов (до 200 кГц) в соответствии с МЭК;

– *тип дискретного выхода:* транзисторный (источник);

– *количество дискретных выходов:* 10 транзисторных, из них 4 быстродействующих (до 100 кГц);

– *напряжение дискретного выхода:* 24 В, постоянный ток;

– *ток дискретного выхода:* Q0–Q9 – 0,5 А, Q0–Q3 –0,1 А в режиме быстродействующего выхода (режим PTO).

ПИД-РЕГУЛИРОВАНИЕ

Целью любого управления является изменение состояния объекта нужным образом (в соответствии с техническим заданием). Теория автоматического управления отвечает на вопрос построения правильного регулятора, который управляет объектом так, чтобы цель была достигнута. Поэтому разработчик сталкивается с необходимостью понять, как данная система управления реагирует на различные воздействия; ему необходимо знать модель системы: какой объект используется, типы датчиков, каналы связи, возмущения системы, а также шумы. Для поддержания постоянного давления (расхода) жидкости в промышленности используются ПИД-регуляторы [2].

ПИД-регулятор – это пропорционально-интегрально-дифференциальный регулятор. Операторно-структурная схема представлена на рисунке 2.

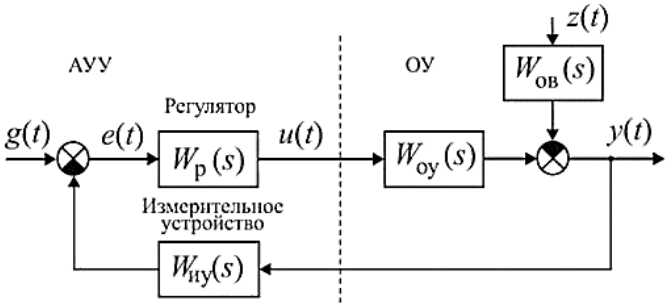

Рис. 2. Операторно-структурная схема АСУ

Проблема заключается в том, что на практике необходимо определять коэффициенты регулятора (П – пропорциональный, И – интегральный, Д – дифференциальный). Все коэффициенты должны учитывать особенности управляемой системы. В связи с этим на практике часто возникают проблемы, например некачественная настройка ПИД-регуляторов, так как каждая система должна иметь собственные коэффициенты в силу использования разного оборудования и предназначения самой системы. Выбранные коэффициенты, подходящие для одной системы, не обязательно будут учитывать особенности работы другой. Следовательно, управление становится неэффективным.

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ПИД-РЕГУЛЯТОРА

Для создания качественной адаптивной системы управления необходимо создание функционального блока, который будет находить оптимальные коэффициенты ПИД-регулятора в автоматическом режиме.

Выходной сигнал, формирующийся ПИД-регулятором, рассчитывается по формуле:

$$u(t) = P + I + D = K_p e(t) + K_i \int_0^t e(\tau)\,d\tau + K_d \frac{de}{dt}, \qquad (1)$$

где K_p – коэффициент пропорциональности;

K_i – интегральный коэффициент;

K_d – дифференциальный коэффициент;

$e(t)$ – функция ошибки регулирования.

Большая часть методов настройки ПИД-регуляторов используют другую формулу, определяющую значение выходного сигнала, в которой интегрирующая, а также дифференцирующая составляющие умножены на пропорциональный коэффициент:

$$u(t) = K_p \times (e(t) + K_{ip} \int_0^t e(\tau)\,d\tau + K_{dp} \frac{de}{dt}). \qquad (2)$$

Для реализации этого метода расчета в дискретной форме уравнение (3) имеет вид

$$U(n) = K_p E(n) + K_p K_{ip} T \sum_{k=0}^{n} E(k) + \frac{K_p K_{dp}}{T}\left(E(n) - E(n-1)\right), \qquad (3)$$

где T – время дискретизации.

Заменив $K^{discr} = K_p * K_{ip} * T$, $K^{discr}{}_d = (K_p * K_{dp}) / T$, получаем следующее выражение:

$$U(n) = K_p E(n) + K_i^{discr} \sum_{k=0}^{n} E(k) + K_d^{discr}\left(E(n) - E(n-1)\right). \qquad (4)$$

При программной реализации для оптимизации расчетов применяется формула вида

$$(n) = U(n-1) + K_p\left(E(n) - E(n-1)\right) + K_i^{discr} E(n) + K_d^{discr}\,(E(n) - 2E(n-1) + E(n-2). \qquad (5)$$

В таблице представлены этапы автоматического подбора коэффициентов ПИД-регулятора.

Этапы автоматического подбора коэффициентов ПИД-регулятора

Номер этапа	Описание этапа
1	Выход ПИД-регулятора устанавливается в значение «уставка + допустимое значение перерегулирования». После достижения этого значения процесс настройки переходит ко второму этапу
2	Этап состоит из двух шагов: 1) выход ПИД-регулятора устанавливается в значение «уставка – допустимое значение недорегулирования», пока не достигнет этого значения; 2) выход ПИД-регулятора устанавливается в значение «уставка + допустимое значение перерегулирования», пока не достигнет этого значения
3	Выход ПИД-регулятора устанавливается в значение «уставка – допустимое значение недорегулирования», пока не достигнет этого значения
4	Этап состоит из двух шагов: 1) выход ПИД-регулятора устанавливается в значение «уставка + допустимое значение перерегулирования», пока не достигнет этого значения; 2) выход ПИД-регулятора устанавливается в значение «уставка – допустимое значение недорегулирования», производится расчет коэффициентов и активируется работа ПИД-регулятора с вычисленными коэффициентами

Программная реализация ПИД-регулятора

В программе реализован блок ПИД-регулятора с автоматической настройкой для управления частотой вращения насосов в случае использования частотных преобразователей [3]. Функциональный блок представлен на рисунке 3.

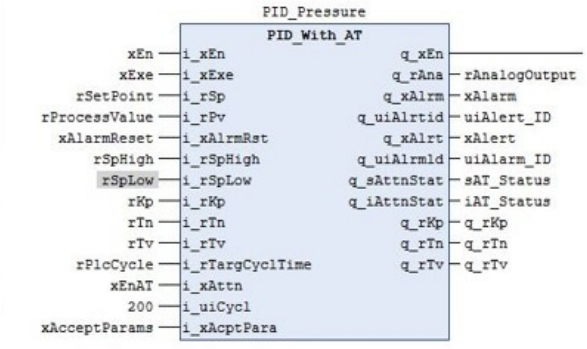

Рис. 3. Функциональный блок ПИД-регулятора

Подачей сигнала «TRUE» на вход i_xEn разрешается работа функционального блока. Затем запускается автоматическая настройка ПИД-регулятора положительным сигналом на входе i_xAttn. Введенные параметры инициализируются по входу i_xExe. Автоматическая настройка запущена, статус работы отслеживается на выходах q_sAttnStat и q_iAttnStat.

После этого происходит подсчет параметров ПИД-регулятора (q_iAttnStat = 6), которые необходимо принять (i_xAcptPara = TRUE), по положительному фронту сигнала на входе i_xExe. Далее необходимо сбросить значение параметров i_xAttn и i_xAcptPara. Процесс автоматической настройки ПИД-регулятора на этом завершен [4].

Таким образом, использование ПИД-регулятора с автоматическим подбором коэффициентов позволяет сократить количество испытаний системы и, следовательно, ускорить ввод в эксплуатацию объекта. Также это позволяет снизить требования к техническим знаниям персонала, так как используется простая технология оптимальной настройки.

Автоматический подбор коэффициентов создает устойчивую систему регулирования. Применение ПИД-регулятора позволяет адаптивно управлять системой при изменении исходных значений параметров и внешних воздействий. Плавные изменения параметров регулирования повышают срок службы механических объектов (насосов), гидравлических объектов (труб) благодаря отсутствию гидроударов.

Использование ПИД-регулирования в комбинации с примененным в программе методом выравнивания наработки насосов по времени и количеству пусков увеличивает наработку на отказ, следовательно, уменьшает стоимость эксплуатации системы, а также увеличивает интервал между этапами технического обслуживания оборудования. Автоматическое оповещение об авариях и возможность удаленного наблюдения за работой системы позволяет снизить стоимость обслуживания.

Программно-аппаратный комплекс

Для создания программно-аппаратного комплекса, помимо написания программного обеспечения, необходимо собрать макет (аппаратную часть).

Макет собирается на основании структурной и электрической схем. Каждый элемент соединен с другим определенным образом, соответствующим видом связи, как показано на рисунке 4.

Макет изготовлен на монтажной плате, на которой размещены контроллер, панель оператора, блок питания, элементы управления (переключатели, кнопки), элементы сигнализации (лампы), автоматические выключатели. К контроллеру подключены датчик температуры (Pt1000) и ультразвуковой датчик измерения расстояния.

Макет имитирует работу системы по наливу жидкости в железнодорожную цистерну с автоматическим отключением насосов по достижении уровня заполнения, а также включением системы обогрева в условиях низких температур окружающей среды.

Разработанное программное обеспечение для программируемого логического контроллера позволяет выбирать насос с наименьшим количеством пусков. В случае их равенства включается тот, у которого время работы наименьшее. При этом учитывается аварийное состояние насосов и состояние переключателей (ручной / автоматический).

Для панели оператора разработан графический интерфейс, отображающий параметры работы установки и визуализирующий наполнение цистерны жидкостью.

Макет автоматизированной насосной станции наливных терминалов собирается по правильно составленной электрической схеме. Для удобства в построении макета электрические схемы собираются поэтапно.

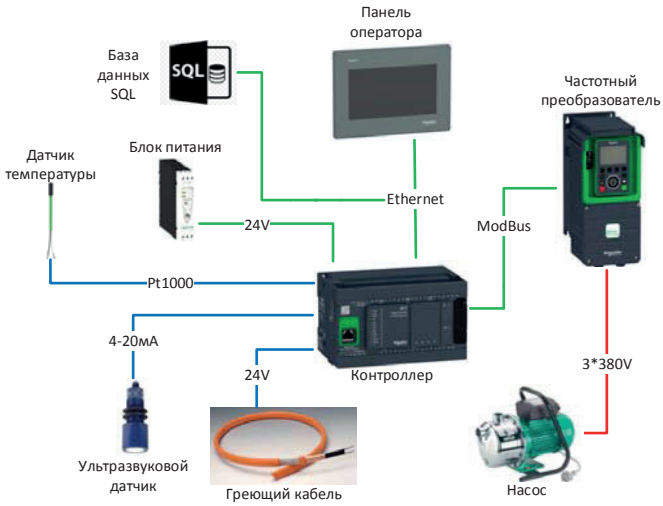

Рис. 4. Структурная схема системы

На рисунке 5 приведена схема подключения питания, а также связь панели оператора и контроллера по Ethernet.

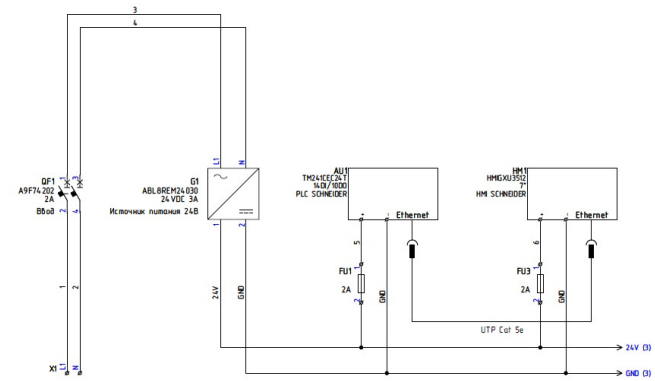

Рис. 5. Схема подключения питания к контроллеру и панели

На рисунке 6 приведена схема подключения переключателей и кнопки к дискретных входам контроллера (I0–I4).

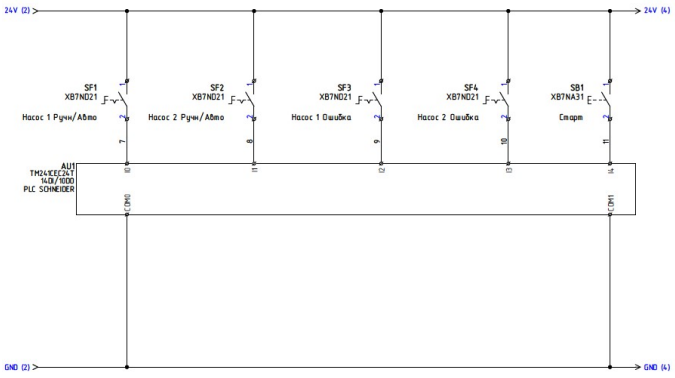

Рис. 6. Схема подключения переключателей и кнопки

На рисунке 7 приведена схема подключения аналоговых сигналов с датчика к модулю TM3TI4 контроллера.

На макете проверены и отработаны алгоритмы управления насосными станциями с разным числом насосов. Проверена функциональность и достаточность контроллера для решения подобных задач.

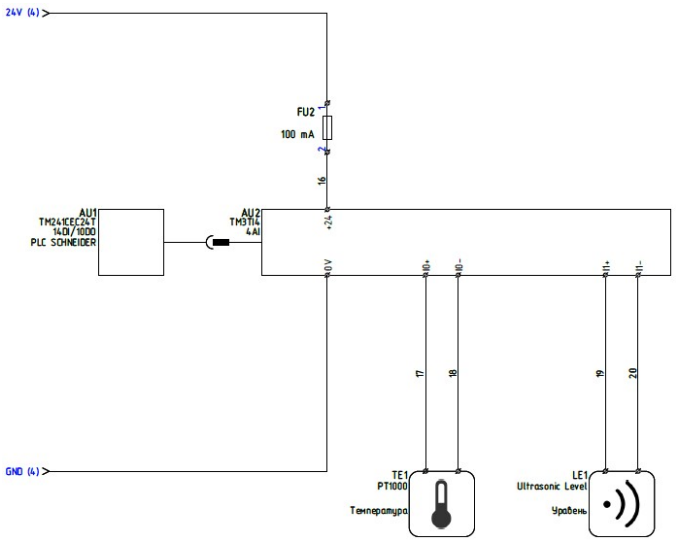

Рис. 7. Схема подключения аналоговых сигналов

ПРЕИМУЩЕСТВА РАЗРАБОТАННОЙ СИСТЕМЫ:

– *система управления насосами* позволяет насосам работать в эффективной зоне, сократить расход электроэнергии на метр кубический наливаемой жидкости (вода, нефтепродукты), предотвратить работу на холостом ходу;

– *система резервирования насосов* исключает простои в работе системы;

– *система выравнивания наработки насосов* обеспечивает равномерность износа насосов;

– *система подогрева* предотвращает замерзание трубопровода;

– *система измерения* позволяет точно учитывать расходы, вести точный учет перекаченной жидкости, предотвращает переливы;

– *система оповещения и связи с базой данных* позволяет удаленно вести учет отказов, а также учет всех результатов работы.

Разработанная программа допускает использование станций с частотным регулированием и без него, в зависимости от необходимости. Возможно использование разных типов датчиков для определения количества налитой жидкости, например гидростатических или, как в этой разработке, ультразвуковых, а также датчиков, имеющих выходы (0–10 В, 2–10 В, 0–20 мА, 4–20 мА).

Для панели оператора разработан графический интерфейс, при помощи которого оператор может отслеживать состояние системы, процесс перекачки жидкости, а также всю информацию о текущем состоянии каждого насоса.

В комплексе предусмотрено увеличение числа насосов, необходимых для работы. Это означает, что программа масштабируема и подходит для *n*-го числа насосов. Это достигнуто с помощью применения методов объектно-ориентированного программирования.

Заключение

Разработанный программно-аппаратный комплекс адаптивного управления насосами наливных терминалов решает ряд оптимизационных задач:

– внедрение автоматизированной системы управления, которая заменяет человеческий труд, уменьшает влияние человеческого фактора в процессе перекачки жидкостей;

– наличие резервирования насосов продлевает длительность работы устройств и механизмов;

– наличие системы ПИД-регулирования способствует работе насоса в эффективной зоне;

– наличие системы обогрева не позволяет замерзать подающим трубам в холодное время года;

– учет всех произведенных действий по перекачке различных жидкостей благодаря наличию системы контроля заполнения.

Говоря о направлении дальнейших исследований, отметим работу [5], в которой подчеркивается актуальность совершенствования энергосберегающих технологий на примере центробежных насосов с асинхронными двигателями.

Стремление к эффективности расходования энергии вызывает необходимость перехода от дроссельного регулирования насосных агрегатов к системам автоматического управления ими [6, 7]. Одной из целей оптимизации частотно-регулируемых электроприводов является достижение минимума функции потерь мощности [8].

На наш взгляд, в статье [5] справедливо подчеркивается необходимость рассмотрения динамических моделей установок центробежных насосов, входящих в состав электротехнических комплексов, что позволяет реализовать наиболее эффективные режимы работы с точки зрения энергосбережения, а также надежности.

Отметим также возможность применения систем адаптивного управления в контурах отбора тепла у трансформаторных маслоохладителей для последующего его использования для теплоснабжения помещений или горячего водоснабжения [9–11].

Тепловые потери трансформируемой электроэнергии в зависимости от мощности и нагрузки трансформаторов могут достигать сотен киловатт.

Принцип работы. Нагретое масло из верхней части бака трансформатора масляным насосом подается в теп-лообменник «масло–вода», где отдает теплоту другому теплоносителю, который циркулирует в промежуточном контуре между тепловым насосом и теплообменником «масло–вода». В испарителе ТН теплоноситель отдает теплоту фреоновому контуру. Фреон при поступлении теплоты начинает кипеть. После сжатия компрессором и конденсации в конденсаторе ТН температура фреона по-вышается и его теплота передается воде, подаваемой по-требителям.

При температуре масла 20–30 °С температура воды до-стигает значений 55–65 °С. Такую схему применяют на подстанциях мощностью 110–500 кВ.

Литература

1. Петров И.В. Программируемые контроллеры. Стандартные языки и приемы прикладного проектирования / И.В. Петров ; под ред. проф. В. П. Дьяконова. – М. : СОЛОН-Пресс, 2004. – 256 с. – (Библиотека инженера).

2. SoMachine. Programming Guide, Schneider Electric, February 2018. – 912 p. URL: http://www.schneider-electric.com/en/download/document/EIO0000000067.

3. Getting Started With SoMachine. Self Study Manual. SoMachine ver 4.1.1, Schneider Electric, March 2014. – 140 p.

4. Мирошник И.В. Теория автоматического управле-ния. Линейные системы / И.В. Мирошник. – СПб. : Питер, 2005. – 336 с. – (Учебное пособие).

5. Лысенко О.А. Режимы энергосбережения установок центробежных насосов с асинхронными двигателями // Известия Томского политехнического университета. Тех-ника и технологии в энергетике. – 2014. – Т. 325, № 4. – С. 133–141.

6. Ouchbel T. et al. Power Maximization of an Asynchronous Wind Turbine with a Variable Speed Feeding a Centrifugal Pump, *Energy Conversion and Management*, 2014, Vol. 78. – Pp. 976–984.

7. Belgacem B.Gh. Performance of Submersible PV Water Pumping Systems in Tunisia, *Energy for Sustainable Development*, 2012, Vol. 16, Is. 4. – Pp. 415–420.

8. Oguz Y., Dede M. Speed Estimation of Vector Controlled Squirrel Cage Asynchronous Motor with Artificial Neural Networks, *Energy Conversion and Management*, 2011, Vol. 52, Is. 1. – Pp. 675–686.

9. Трубаев П.А. Тепловые насосы : учебное пособие / П.А. Трубаев, Б.М. Гришко. – Белгород : БГТУ им. В.Г. Шухова, 2009. – 142 с.

10. Салова Т.Ю., Методология рационального использования природных ресурсов / Т.Ю. Салова, Е.А. Громова, Н.Ю. Громова // Международный журнал эксперимен-тального образования. – 2017. – № 3-1. – С. 55–57.

11. Sayegh M.A. et al. Heat Pump Placement, Connection and Operational Modes in European District Heating, *Energy and Buildings*, 2018, Vol. 166. – Pp. 122–144.

The Automated System of Adaptive Control of Pumps Filling Terminals on the Basis of the Logical Controller

Bachelor V.I. Vasnikhin, Doctor of Tech. Sci. A.D. Khomonenko, S.I. Nosova
Emperor Alexander I St. Petersburg State Transport University
St. Petersburg, Russia
vasnikhinvictor97@gmail.com, khomon@mail.ru, nosovasi@mail.ru

Abstract. **The developed automated system of adaptive pump control is considered. The choice of a mathematical model was substantiated, software was developed in the SoMachine development environment, a graphical interface for the operator panel in the Vijeo Designer Basic development environment was developed, and an imitation model was assembled that simulates the operation of an adaptive pump control system. The use of a proportional-integral-differential (PID) controller allows you to adaptively adjust the pump control system when changing settings and external influences. Smooth changes of regulation parameters increase the service life of mechanical objects (pumps), hydraulic objects (pipes) due to the lack of water hammer.**

Keywords: **programmable logic controller, MODICON M241-TM241CEC24T controller, adaptive control, pump, PID controller, automation.**

REFERENCES

1. Petrov I.V. Programmable controllers. Standart languages and methods of applied engineering [Programmiruemye kontrollery. Standartnye yazyki i priemy prikladnogo proektirovaniya], Moscow, SOLON-Press Publishers, 2004. – 256 p.

2. SoMachine. Programming Guide, Schneider Electric, February 2018. – 912 p. Available at: http://www.schneider-electric.com/en/download/document/EIO0000000067.

3. Getting Started With SoMachine. Self Study Manual. SoMachine ver 4.1.1, Schneider Electric, March 2014. – 140 p.

4. Miroshnik I.V. Automatic regulation theory. Linear systems [Teoriya avtomaticheskogo upravleniya. Lineynye sistemy], St. Petersburg, Piter, 2005. – 336 p.

5. Lysenko O.A. Energy Saving Modes of Centrifugal Pumps Installations with Asynchronous Engines [Rezhimy energosberezheniya ustanovok tsentrobezhnykh nasosov s asinkhronnymi dvigatelyami], *Bulletin of the Tomsk Polytechnic University. Engineering and Technologies in Power Engineering [Izvestiya Tomskogo politekhnicheskogo universiteta. Tekhnika i tekhnologii v energetike]*, 2014, Vol. 325. – No. 4, Pp. 133–141.

6. Ouchbel T. et al. Power Maximization of an Asynchronous Wind Turbine with a Variable Speed Feeding a Centrifugal Pump, *Energy Conversion and Management*, 2014, Vol. 78. – Pp. 976–984.

7. Belgacem B.Gh. Performance of Submersible PV Water Pumping Systems in Tunisia, *Energy for Sustainable Development*, 2012, Vol. 16, Is. 4. – Pp. 415–420.

8. Oguz Y., Dede M. Speed Estimation of Vector Controlled Squirrel Cage Asynchronous Motor with Artificial Neural Networks, *Energy Conversion and Management*, 2011, Vol. 52, Is. 1. – Pp. 675–686.

9. Trubaev P.A., Grishko B.M. Heat pumps: Study guide [Teplovye nasosy: Uchebnoe posobie], Belgorod, Belgorod State Technological University named after V.G. Shukhov, 2009. – 142 p.

10. Salova T.Yu., Gromova E.A., Gromova N.Yu. Methodology for the Rational Use of Natural Resources [Metodologiya ratsional'nogo ispol'zovaniya prirodnykh resursov], *International Journal of Experimental Education [Mezhdunarodnyy zhurnal eksperimental'nogo obrazovaniya]*, 2017, No. 3, Is. 1. – Pp. 55–57.

11. Sayegh M.A. et al. Heat Pump Placement, Connection and Operational Modes in European District Heating, *Energy and Buildings*, 2018, Vol. 166. – Pp. 122–144.

Модель определения периодичности контроля технического состояния сложных технических комплексов с использованием нечеткой логики

Канд. техн. наук И.В. Дорожко, А.Б. Петухов, О.А. Салун
Военно-космическая академия имени А.Ф. Можайского
Санкт-Петербург, Россия
Doroghko-Igor@yandex.ru

Аннотация. **Работа посвящена решению задачи определения и коррекции периодичности контроля исходя из текущего состояния объекта в целях обеспечения предъявляемых требований к коэффициенту готовности сложных технических комплексов. Предлагается имитационная модель, разработанная с помощью среды моделирования Stateflow программного продукта Matlab, содержащая нечеткий логический контроллер, в котором периодичность контроля и требования к коэффициенту готовности заданы как лингвистические переменные.**

Ключевые слова: **надежность, контроль, нечеткая логика, коэффициент готовности, периодичность контроля, марковский процесс, имитационная модель.**

Введение

Вопросы обеспечения надежности и контроля (диагностирования) сложных технических комплексов (СТК) имеют важное значение и являются взаимосвязанными [1–3]. Современные структурно сложные комплексы имеют в своем составе встроенные средства контроля и диагностирования [4].

В настоящее время наряду с календарной схемой технического обслуживания (ТО) внедряется и осуществляется ТО по состоянию. Своевременные и целесообразные по глубине и объему контроль и техническое обслуживание позволяют:

– поддерживать установленное техническое состояние и эффективное использование СТК;

– уменьшить число преждевременных вмешательств в работу СТК;

– предотвратить отказы в процессе применения СТК за счет проведения необходимых профилактических мероприятий при обнаружении предотказного состояния [5].

Современные СТК и составляющие их элементы чрезвычайно разнообразны, вследствие этого нельзя предложить какую-либо единую стратегию контроля, которая была бы пригодна для любого из них и для любых условий эксплуатации и при этом обеспечивала бы высокую готовность, экономичность и безопасность эксплуатации.

В связи с этим наиболее целесообразным подходом к выбору стратегий контроля является комплексный подход, ориентированный на рациональное сочетание различных видов контроля с различной периодичностью [5].

Следовательно, выбор и корректировка периодичности контроля СТК является актуальной задачей.

Модель марковского процесса, связывающая коэффициент готовности и достоверность диагностирования сложных технических комплексов

На рисунке 1 изображена схема графа состояний, представляющего собой описание марковского процесса перехода из одного состояния в другое [6–9]. Модель подробно рассмотрена в работах [10–12] и позволяет связать коэффициент готовности [13] с такими показателями и характеристиками контроля сложных технических комплексов, как достоверность и периодичность контроля [14–16].

В качестве видов состояния марковского процесса выступают работоспособное и неработоспособное техническое состояние объекта (S_0, $\overline{S_0}$), в которых контроль не проводится (т. е. «рабочий режим»); а также состояния, при которых производится контроль: R_1, R_3 – состояния, при которых проводится контроль с достоверным результатом (S_0^*/S_0 – система контроля фиксирует работоспособное состояние S_0^*, при этом объект действительно работоспособен S_0, $\overline{S_0^*}/\overline{S_0}$ – система контроля обнаруживает неработоспособное состояние $\overline{S_0^*}$, при этом объект действительно неработоспособен $\overline{S_0}$); R_2, R_4 – состояния, при которых проводится контроль с ошибочным результатом ($\overline{S_0^*}/S_0$ – система контроля обнаруживает неработоспособное состояние $\overline{S_0^*}$, при этом объект работоспособен S_0; $S_0^*/\overline{S_0}$ – система контроля фиксирует работоспособное состояние S_0^*, при этом объект неработоспособен $\overline{S_0}$). Кроме этого, на рисунке 1 также обозначены: α, β – вероятности ошибок контроля (ошибок 1-го и 2-го рода); $T_{\text{СР}}$ – средняя наработка на отказ; $T_{\text{д}}$ – периодичность контроля; $T_{\text{В1}}$ – среднее время восстановления без применения средств контроля (т. е. предполагается, что объект можно восстановить даже при отсутствии или отказе средств контроля, например, с помощью последовательной замены блоков до тех пор, пока система не станет работоспособной; очевидно, что на это может потребоваться значительное время и ресурсы; если без контроля объект не может быть восстановлен, то $T_{\text{В1}} \to \infty$); $T_{\text{В2}}$ – среднее время восстановления с учетом контроля; $T_{\text{п}}$ – средняя продолжительность перевода объекта из режима контроля в рабочий режим (если контроль происходит параллельно с работой объекта (функциональный контроль), то $T_{\text{п}} = 0$, но если производится тестовый контроль, при котором объект последовательно переводится из режима контроля в рабочий режим и обратно, то необходимо учитывать $T_{\text{п}}$).

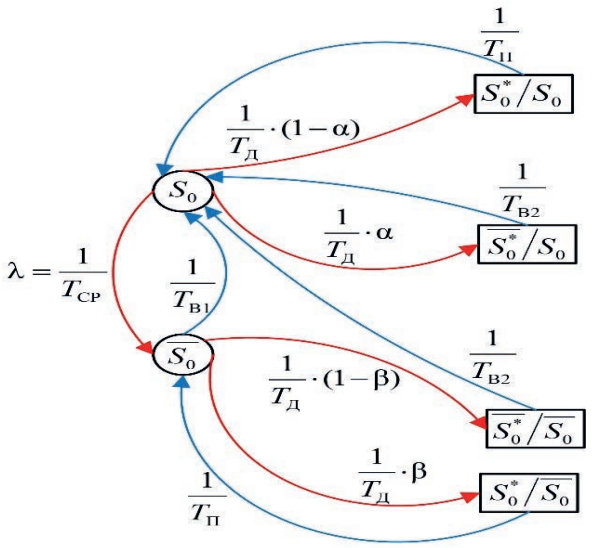

Рис. 1. Схема марковского процесса, связывающая показатели надежности и контроля

Матрица интенсивностей переходов данного марковского процесса будет иметь вид:

$$\boldsymbol{A}=\begin{bmatrix} -\left(\frac{1}{T_{\text{Д}}}+\frac{1}{T_{\text{СР}}}\right) & \frac{1}{T_{\text{СР}}} & \frac{1-\alpha}{T_{\text{Д}}} & \frac{\alpha}{T_{\text{Д}}} & 0 & 0 \\ \frac{1}{T_{\text{В1}}} & -\left(\frac{1}{T_{\text{Д}}}+\frac{1}{T_{\text{В1}}}\right) & 0 & 0 & \frac{1-\beta}{T_{\text{Д}}} & \frac{\beta}{T_{\text{Д}}} \\ \frac{1}{T_{\text{П}}} & 0 & -\frac{1}{T_{\text{П}}} & 0 & 0 & 0 \\ \frac{1}{T_{\text{В2}}} & 0 & 0 & -\frac{1}{T_{\text{В2}}} & 0 & 0 \\ \frac{1}{T_{\text{В2}}} & 0 & 0 & 0 & -\frac{1}{T_{\text{В2}}} & 0 \\ 0 & \frac{1}{T_{\text{П}}} & 0 & 0 & 0 & -\frac{1}{T_{\text{П}}} \end{bmatrix}. \quad (1)$$

Имитационная модель, связывающая коэффициент готовности с показателями и характеристиками контроля СТК

Опираясь на марковскую модель, изображенную на рисунке 1, в среде Stateflow [17] программного продукта Matlab [18] была разработана имитационная модель, связывающая показатели надежности с показателями и характеристиками контроля СТК (рис. 2).

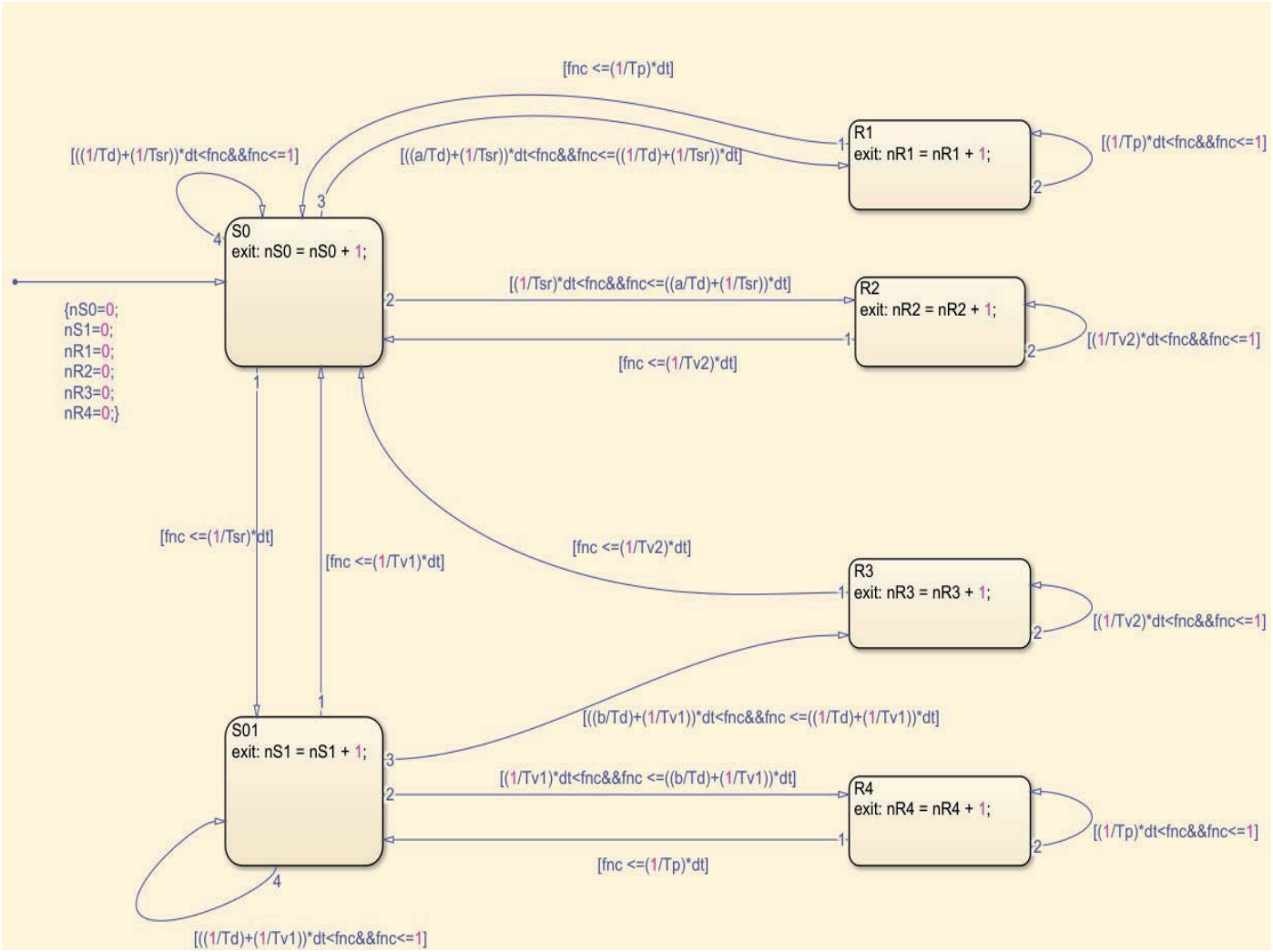

Рис. 2. Имитационная модель, связывающая показатели надежности и контроля СТК

Для построения имитационной модели был осуществлен переход от матрицы интенсивностей переходов к матрице вероятностей переходов с использованием уравнения Колмогорова–Чепмена в следующем виде:

$$P(t+\Delta t) = P(t) \cdot \boldsymbol{P},$$

где $\boldsymbol{P}$ – матрица вероятностей переходов.

Вычтем из правой и левой частей уравнения $P(t)$:

$$P(t+\Delta t) - P(t) = P(t) \cdot \boldsymbol{P} - P(t).$$

Разделим на Δt обе части:

$$\frac{P(t+\Delta t)-P(t)}{\Delta t} = \frac{P(t)\cdot(\boldsymbol{P}-\boldsymbol{E})}{\Delta t},$$

где $\boldsymbol{E}$ – единичная матрица.

При $\Delta t \to 0$ получим:

$$\dot{P}(t) = P(t) \cdot \boldsymbol{A},$$

где $\boldsymbol{A}$ – матрица интенсивностей переходов, $\boldsymbol{A} = \frac{\boldsymbol{P}-\boldsymbol{E}}{\Delta t}$.

Следовательно, матрица вероятностей переходов может быть получена следующим образом:

$$\boldsymbol{P} = \boldsymbol{A} \cdot \Delta t + \boldsymbol{E}. \quad (2)$$

От матрицы интенсивностей переходов (1) с помощью соотношения (2) перейдем к матрице вероятностей переходов:

$$\boldsymbol{P} = \begin{bmatrix} 1-\left(\frac{1}{T_д}+\frac{1}{T_{СР}}\right)\cdot\Delta t & \frac{1}{T_{СР}}\cdot\Delta t & \frac{1-\alpha}{T_д}\cdot\Delta t & \frac{\alpha}{T_д}\cdot\Delta t & 0 & 0 \\ \frac{1}{T_{В1}}\cdot\Delta t & 1-\left(\frac{1}{T_д}+\frac{1}{T_{В1}}\right)\cdot\Delta t & 0 & 0 & \frac{1-\beta}{T_д}\cdot\Delta t & \frac{\beta}{T_д}\cdot\Delta t \\ \frac{1}{T_Л}\cdot\Delta t & 0 & 1-\frac{1}{T_П}\cdot\Delta t & 0 & 0 & 0 \\ \frac{1}{T_{В2}}\cdot\Delta t & 0 & 0 & 1-\frac{1}{T_{В2}}\cdot\Delta t & 0 & 0 \\ \frac{1}{T_{В2}}\cdot\Delta t & 0 & 0 & 0 & 1-\frac{1}{T_{В2}}\cdot\Delta t & 0 \\ 0 & \frac{1}{T_П}\cdot\Delta t & 0 & 0 & 0 & 1-\frac{1}{T_П}\cdot\Delta t \end{bmatrix}. \quad (3)$$

В основу имитационной модели положены принципы моделирования непрерывных случайных величин с произвольными законами распределения. Символами «fnc» обозначено сгенерированное случайное число от 0 до 1 с равномерным законом распределения.

Применение нечеткой логики для определения периодичности контроля СТК

Основной сложностью при определении периодичности контроля является то, что коэффициент готовности имеет нелинейную зависимость с периодичностью контроля. Также при определенных показателях надежности СТК и характеристиках системы контроля и диагностирования возможны три типа решений:

– $T_д \to 0$ (контроль должен быть постоянным). Например, в случаях, когда отказавший СТК не может быть восстановлен без результатов контроля, при этом проводится функциональный контроль, не увеличивающий время простоя, и результаты контроля имеют высокую достоверность, т. е. «чем чаще контролируем, тем лучше»;

– существует оптимальное значение $T_д$. В случаях, когда очень частый контроль ведет к увеличению времени простоя, ошибки контроля ведут к ненужным заменам работоспособных элементов, однако очень редкий контроль также приводит к тому, что своевременно не проводятся профилактические мероприятия, что влечет за собой снижение готовности СТК;

– $T_д \to \infty$ (контроль из-за низкой достоверности снижает готовность СТК и должен проводиться как можно реже или вообще не проводиться).

Для определения периодичности контроля был выбран аппарат нечеткой логики, который эффективно применяется при обработке экспертных оценок. Периодичность контроля может быть представлена нечеткой лингвистической переменной, содержащей термы «очень часто», «часто», «редко», «очень редко», «никогда». Входными переменными являются:

– разница текущего коэффициента готовности и требуемого: $\left(K_г(t) - K_{треб}\right)$;

– скорость изменения коэффициента готовности: $\frac{K_г(t+\Delta t)-K_г(t)}{\Delta t}$;

– текущее значение периодичности диагностирования: $T_д(t)$;

– скорость изменения периодичности диагностирования: $\frac{T_д(t+\Delta t)-T_д(t)}{\Delta t}$.

Выходной переменной является периодичность диагностирования.

Входная переменная «текущая периодичность диагностирования» и выходная переменная «периодичность диагностирования» имеют одинаковые термы и параметры, которые задают эти термы.

На рисунке 3 приведено описание термов переменных (общий вид всех функций принадлежности термов входных и выходных параметров модели нечеткого регулятора). Изменяя параметры терма «удовлетворяет» лингвистической переменной «разница текущего коэффициента готовности и требуемого», можно задавать допуски (точность регулирования) коэффициента готовности. Из рисунка 3 видно, что для рассматриваемого примера точность настройки составит ±0,2, т. е. коэффициент готовности будет считаться удовлетворяющим требования при значениях $K_г = K_{треб} \pm 0{,}2$.

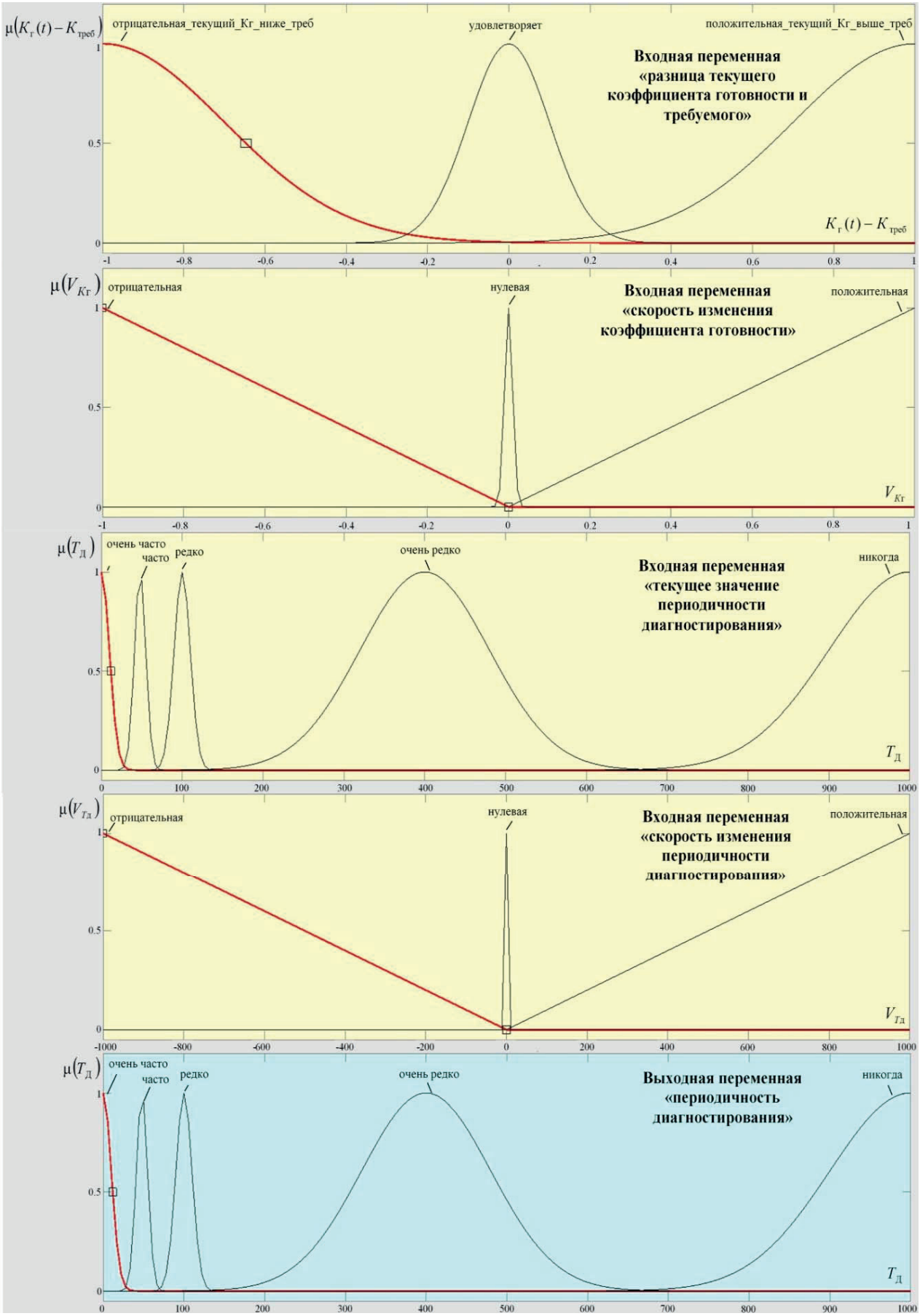

Рис. 3. Описание термов переменных

База знаний нечеткого логического контроллера была составлена по экспертным оценкам и представляет собой продукционные правила в виде «если условие выполнено, то выполнить действие». Приведем примеры продукционных правил.

1. «**Если** (*разница текущего коэффициента готовности и требуемого* – «**удовлетворяет**») и (скорость изменения коэффициента готовности – «**удовлетворяет**»), и (текущее значение периодичности диагностирования – «**редко**»), и (скорость изменения периодичности диагностирования – «**нулевая**»), **то** (периодичность диагностирования – «**редко**»)», т. е. периодичность контроля (диагностирования) не следует менять.

2. «**Если** (*разница текущего коэффициента готовности и требуемого* – «**отрицательная**»), и (скорость изменения коэффициента готовности – «**отрицательная**»), и (текущее значение периодичности диагностирования – «**редко**»), и (скорость изменения периодичности диагностирования – «**отрицательная**»), **то** (периодичность диагностирования – «**очень редко**»)», т. е. предпринимаемые попытки до этого повысить коэффициент готовности учащением контроля (диагностирования) приводят к еще большему уменьшению коэффициента готовности, следовательно, периодичность контроля (диагностирования) следует повысить.

3. «**Если** (*разница текущего коэффициента готовности и требуемого* – «**отрицательная**») и (скорость изменения коэффициента готовности – «**положительная**»), и (текущее значение периодичности диагностирования – «**редко**»), и (скорость изменения периодичности диагностирования – «**отрицательная**»), **то** (периодичность диагностирования – «**часто**»)», т. е. предпринимаемые попытки до этого повысить коэффициент готовности учащением контроля (диагностирования) приводят к увеличению коэффициента готовности, следовательно, периодичность контроля (диагностирования) следует продолжать уменьшать.

Для реализации базы знаний нечеткого логического контроллера были составлены 60 правил. Фрагмент базы правил (базы знаний) нечеткого логического контроллера представлен на рисунке 4.

37. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is часто) and (скорость__изменения__Тд is нулевая) then (Тд is редко) (1)
38. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is редко) and (скорость__изменения__Тд is нулевая) then (Тд is часто) (1)
39. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is очень__редко) and (скорость__изменения__Тд is нулевая) then (Тд is редко) (1)
40. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is никогда) and (скорость__изменения__Тд is нулевая) then (Тд is очень__редко) (1)
41. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is очень__часто) and (скорость__изменения__Тд is положительная) then (Тд is очень__часто) (1)
42. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is часто) and (скорость__изменения__Тд is положительная) then (Тд is очень__часто) (1)
43. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is очень__редко) and (скорость__изменения__Тд is положительная) then (Тд is редко) (1)
44. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is никогда) and (скорость__изменения__Тд is положительная) then (Тд is очень__редко) (1)
45. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is очень__часто) and (скорость__изменения__Тд is отрицательная) then (Тд is часто) (1)
46. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is часто) and (скорость__изменения__Тд is отрицательная) then (Тд is редко) (1)
47. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is редко) and (скорость__изменения__Тд is отрицательная) then (Тд is очень__редко) (1)
48. If (разница__текущего_требуемого_Kr is отрицательная_текущий_Kr__ниже_треб) and (VKr is отрицательная) and (Тд__текущее is очень__редко) and (скорость__изменения__Тд is отрицательная) then (Тд is никогда) (1)
49. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is никогда) and (скорость__изменения__Тд is нулевая) then (Тд is очень__редко) (1)
50. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is очень__редко) and (скорость__изменения__Тд is нулевая) then (Тд is редко) (1)
51. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is редко) and (скорость__изменения__Тд is нулевая) then (Тд is часто) (1)
52. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is часто) and (скорость__изменения__Тд is нулевая) then (Тд is очень__часто) (1)
53. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is никогда) and (скорость__изменения__Тд is положительная) then (Тд is очень__редко) (1)
54. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is очень__редко) and (скорость__изменения__Тд is положительная) then (Тд is редко) (1)
55. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is редко) and (скорость__изменения__Тд is положительная) then (Тд is часто) (1)
56. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is часто) and (скорость__изменения__Тд is положительная) then (Тд is очень__часто) (1)
57. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is никогда) and (скорость__изменения__Тд is отрицательная) then (Тд is очень__редко) (1)
58. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is очень__редко) and (скорость__изменения__Тд is отрицательная) then (Тд is редко) (1)
59. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is редко) and (скорость__изменения__Тд is отрицательная) then (Тд is часто) (1)
60. If (разница__текущего_требуемого_Kr is not отрицательная_текущий_Kr__ниже_треб) and (VKr is нулевая) and (Тд__текущее is часто) and (скорость__изменения__Тд is отрицательная) then (Тд is очень__часто) (1)

Рис. 4. Фрагмент базы правил нечеткого логического контроллера

На рисунке 5 представлена модель надежности и контроля СТК, содержащая контур обратной связи для корректировки периодичности контроля с помощью нечеткого логического контроллера.

На рисунке 6 представлен график $T_д(t)$, отображающий зависимость регулируемой (корректируемой) величины периодичности контроля (диагностирования) $T_д$ от времени эксплуатации t.

Анализ графика, изображенного на рисунке 6, показывает, что на начальном этапе эксплуатации периодичность контроля варьируется в пределах от 27 до 49 часов, а в дальнейшем составляет 5,8 ч, при этом будет обеспечено требуемое значение коэффициента готовности $K_{треб} = 0{,}9$ (см. рис. 5). Уменьшение периодичности контроля в процессе эксплуатации СТК до 5,8 ч (учащение контроля) связано с наступлением старения (износа) СТК, но на начальном этапе нет необходимости в таком частом контроле.

Для сравнения было проведено имитационное моделирование без использования нечеткого логического контроллера при строго заданной периодичности контроля $T_д$ = 5 ч. При этом коэффициент готовности СТК не удовлетворял требованиям и составил 0,771. Это объясняется тем, что при заданных показателях надежности СТК и характеристиках контроля частый перевод СТК в режим контроля приводит, во-первых, к уменьшению продолжительности его применения по назначению, во-вторых, к учащению

случаев, когда происходят ошибки контроля, по результатам которых производится восстановление (замена) работоспособных элементов, что также снижает коэффициент готовности СТК.

Результаты тестирования нечеткого логического контроллера позволяют сделать вывод о положительном эффекте от применения нечеткого логического контроллера при определении и коррекции периодичности контроля СТК.

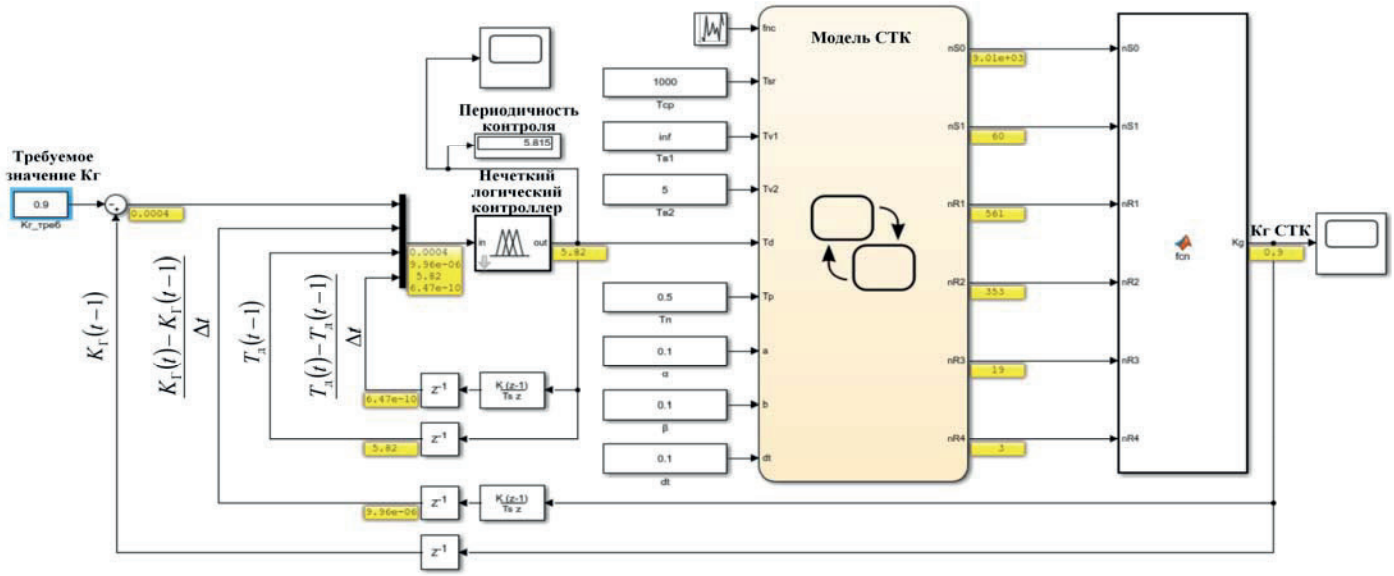

Рис. 5. Имитационная модель надежности и контроля СТК с использованием нечеткого логического контроллера

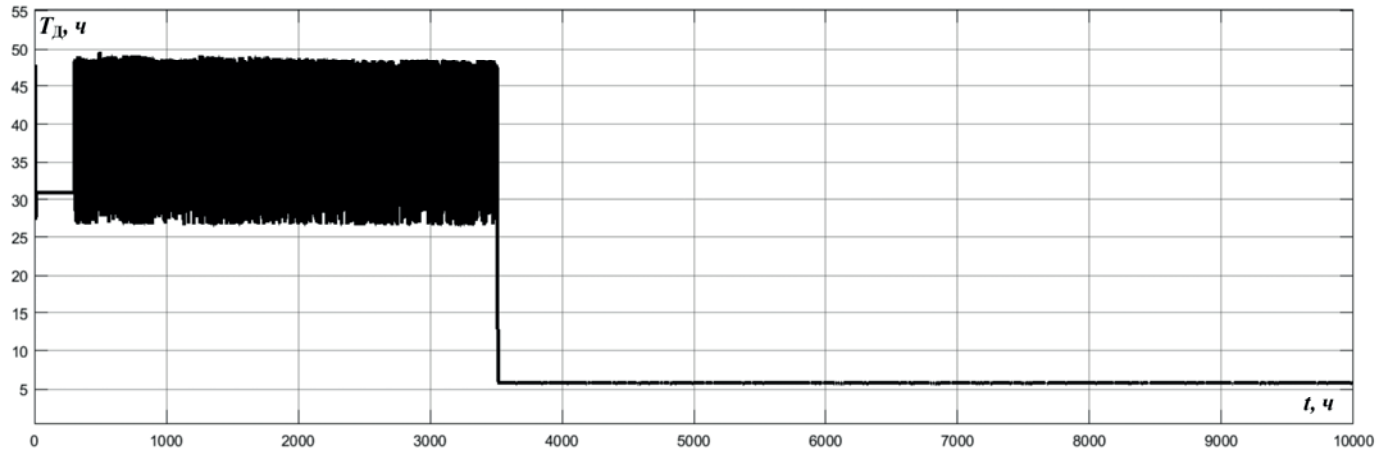

Рис. 6. График изменения $T_д$ в зависимости от времени эксплуатации

Заключение

Разработанная имитационная модель связывает показатели надежности и характеристики системы контроля (диагностирования) СТК. Применение нечеткой логики позволяет производить корректировку периодичности контроля для обеспечения требований по коэффициенту готовности.

Литература

1. Боровиков С.М. Расчет показателей надежности радиоэлектронных средств : учебно-метод. пособие / С.М. Боровиков, И.Н. Цырельчук, Ф.Д. Троян ; под ред. С.М. Боровикова. – Минск : БГУИР, 2010. – 68 с.

2. Острейковский В.А. Теория надежности : учебник для вузов / В.А. Острейковский. – М. : Высшая школа, 2003. – 463 с.

3. Половко А.М. Основы теории надежности / А.М. Половко, С.В. Гуров. – 2-е изд., перераб. и доп. – СПб. : БХВ-Петербург, 2006. – 704 с.

4. Дорожко И.В. Оценка надежности структурно сложных технических комплексов с помощью моделей байесовских сетей доверия в среде GeNIe / И.В. Дорожко, А.Г. Тарасов, А.М. Барановский // Интеллектуальные технологии на транспорте. – 2015. – № 3. – С. 36–45.

5. Эксплуатация космических средств : учебник / Г.Д. Петров [и др.]; под ред. А.П. Вышинского. – СПб. : ВКА им. А.Ф. Можайского, 2015. – 454 с.

6. IEC 61165:2006. Application of Markov techniques, British Standards Institution, 29 February 2008. – 40 p.

7. IEC 61070:1991. Compliance test procedures for steady-state availability, International Electrotechnical Commission, 25 November 1991. – 52 p.

8. Mechatronic Systems Techniques and Applications, Taylor & Francis Group, 2000, XVI, Volume 5. – 362 p.

9. Shubinsky I. B., Zamyshlyaev A. M. Topological Semi-Markov Method for Calculation of Stationary Parameters of Reliability and Functional Safety of Technical Systems, *Reliability: Theory & Applications*, 2012, Vol. 1, No. 2(25). – Pp. 12–22.

10. Научно-методический подход к оцениванию готовности сложных технических комплексов с учетом метрологического обеспечения / Я.Н. Гусеница, И.В. Дорожко, И.А. Кочанов, А.Б. Петухов // Труды МАИ. – 2018. – № 98. – URL: http://trudymai.ru/published.php?ID=90383.

11. Дорожко И.В. Исследование коэффициента готовности сложных технических комплексов с помощью имитационной модели, разработанной в среде Stateflow пакета MatLab / И.В. Дорожко, А.Л. Копейка // Интеллектуальные технологии на транспорте. – 2018. – № 3(15). – С.18–26.

12. Дорожко И.В. Комплексная модель надежности и диагностирования сложных технических систем / И.В. Дорожко, Н.А. Осипов, И.А. Кочанов, А.В. Бутырин // Труды Военно-космической академии им. А.Ф. Можайского. – 2016. – № 652. – С. 137–146.

13. ГОСТ 27.002-15. Надежность в технике. Термины и определения. – М. : Стандартинформ, 2016. – 24 с.

14. ГОСТ 20.911-89. Техническая диагностика. Термины и определения. – М. : Издательство стандартов, 1989. – 12 с.

15. Технические средства диагностирования : справочник / В.В. Клюев [и др.] ; под общ. ред. В.В. Клюева. – М. : Машиностроение, 1989. – 671 с.

16. Bubnov V.P., Khomonenko A.D., Tyrva A.V. Software Reliability Model with Coxian Distribution of Length of Intervals Between Errors Detection and Fixing Moments, *Proceedings of 35th Annual IEEE Computer Software and Applications Conference Workshops (COMPSACW 2011)*, Munich, 18–22 July 2011. – Pp. 310–314.

17. Stateflow® and Stateflow® Coder™ User's Guide, The MathWorks Inc., 2017. – 3290 p.

18. Дьяконов В.П. Simulink 5/6/7 : самоучитель. – М. : ДМК-Пресс, 2008. – 784 с.

Model for Determining the Frequency of Monitoring the Technical Condition of Complex Technical Systems Using Fuzzy Logic

PhD I.V. Dorozhko, A.B. Petukhov, O.A. Salun
Military Space Academy named after A.F. Mozhaisky
Saint Petersburg, Russia
Doroghko-Igor@yandex.ru

Abstract. **The work is devoted to the problem of determining and correlating the periodicity of control, based on the current state of the object, in order to ensure the requirements for the coefficient of readiness of complex technical complexes. The simulation model developed using the modeling environment Stateflow software Matlab that contains fuzzy logic controller, in which the frequency of control and the requirements for availability are defined as linguistic variables.**

Keywords: **reliability, technical state inspection, fuzzy logic, readiness coefficient, periodicity of control, Markov process, simulation model.**

REFERENCES

1. Borovikov S.M., Tsyrelchuk I.N., Troyan F.D. Calculation of indicators of reliability of radio-electronic means: Study guide [Raschet pokazateley nadezhnosti radioelektronnykh sredstv: uchebno-metodicheskoe posobie], Minsk, Belarusian State University of Informatics and Radioelectronics, 2010. – 68 p.
2. Ostreikovsky V.A. Reliability theory: Textbook for high schools [Teoriya nadezhnosti: Uchebnik dlya vuzov], Moscow, Vysshaya Shkola Publishers, 2003. – 463 p.
3. Polovko A.M., Gurov S.V. Fundamentals of theory of reliability [Osnovy teorii nadezhnosti], Saint Petersbirg, BHV-Peterburg, 2006. – 704 p.
4. Dorozhko I.V., Tarasov A.G., Baranovsky A.M. Estimation to Reliability of Structural Complex Technical Systems by Using Bayesian Networks Belief Models in the Environment of GeNIe [Otsenka nadezhnosti strukturno slozhnykh tekhnicheskikh kompleksov s pomoshch`yu modeley bayesovskikh setey doveriya v srede GeNIe], *Intellectual Technologies on Transport [Intellektual`nye tekhnologii na transporte]*, 2015, No. 3. – Pp. 36–45.
5. Petrov G.D. et al. Operation of space vehicles: Textbook [Ekspluatatsiya kosmicheskikh sredstv: Uchebnik], Saint Petersburg, Military Space Academy named after A.F. Mozhaisky, 2015. – 454 p.
6. IEC 61165:2006. Application of Markov techniques, British Standards Institution, 29 February 2008. – 40 p.
7. IEC 61070:1991. Compliance test procedures for steady-state availability, International Electrotechnical Commission, 25 November 1991. – 52 p.
8. Mechatronic Systems Techniques and Applications, Taylor & Francis Group, 2000, XVI, Volume 5. – 362 p.
9. Shubinsky I. B., Zamyshlyaev A. M. Topological Semi-Markov Method for Calculation of Stationary Parameters of Reliability and Functional Safety of Technical Systems, *Reliability: Theory & Applications*, 2012, Vol. 1, No. 2 (25). – Pp. 12–22.
10. Gusenitsa Ya.N., Dorozhko I.V., Kochanov I.A., Petukhov A.B. Scientific-Methods Approach to Evaluating Readiness of Complex Technical Systems with Account for Metrological Assurance [Nauchno-metodicheskiy podkhod k otsenivaniyu gotovnosti slozhnykh tekhnicheskikh kompleksov s uchetom metrologicheskogo obespecheniya], Trudy MAI *[Trudy MAI]*, 2018, No. 98. Available at: http://trudymai.ru/published.php?ID=90383.
11. Dorozhko I.V., Kopeyka A.L. A Study of The Coefficient of Readiness of Complex Technical Systems Using the Simulation Models Developed in the Stateflow Environment of MatLab Package [Issledovanie koeffitsienta gotovnosti slozhnykh tekhnicheskikh kompleksov s pomoshch`yu imitatsionnoy modeli, razrabotannoy v srede Stateflow paketa Matlab], *Intellectual Technologies on Transport [Intellektual`nye tekhnologii na transporte]*, 2018, No. 3 (15). – Pp.18–26.
12. Dorozhko I.V., Osipov N.A., Kochanov I.A., Butyrin A.V. Integrated model of reliability and diagnostics of complex technical systems [Kompleksnaya model` nadezhnosti i diagnostirovaniya slozhnykh tekhnicheskikh sistem], *Proceedings of the Military Space Academy named after A.F. Mozhaisky [Trudy Voenno-kosmicheskoy akademii imeni A.F. Mozhayskogo]*, 2016, No. 652. – Pp. 137–146.
13. GOST 27.002-2015. Dependability in technics. Terms and definitions [GOST 27.002-2015. Nadezhnost` v tekhnike. Terminy i opredeleniya], Moscow, StandardInform, 2016, 24 p.
14. GOST 20.911-89. Technical diagnostics. Terms and definitions [GOST 20.911-89. Tekhnicheskaya diagnostika. Terminy i opredeleniya], Moscow, Standards Publishing, 1989. – 12 p.
15. Klyuev V.V. et al. Technical Diagnostic Tools: Reference Guide [Tekhnicheskie sredstva diagnostirovaniya: Spravochnik], Engineering Publishing House, 1989. – 671 p.
16. Bubnov V.P., Khomonenko A.D., Tyrva A.V. Software Reliability Model with Coxian Distribution of Length of Intervals Between Errors Detection and Fixing Moments, *Proceedings of 35th Annual IEEE Computer Software and Applications Conference Workshops (COMPSACW 2011)*, Munich, 18–22 July 2011. – Pp. 310–314.
17. Stateflow® and Stateflow® Coder™ User's Guide 2017 by The MathWorks Inc. – 3290 p.
18. Dyakonov V.P. Simulink 5/6/7: Tutorial [Simulink 5/6/7: Samouchitel`], Moscow, DMK-Press, 2008. – 784 p.

Концептуальная и графовая модели взаимодействия беспилотного летательного аппарата с наземной робототехнической сервисной платформой

Аспирант Т.К. Нго
Санкт-Петербургский государственный университет
аэрокосмического приборостроения,
Санкт-Петербург, Россия
quoctienbn@gmail.com

Аннотация. **Рассматривается задача увеличения времени работы беспилотных летательных аппаратов при решении сельскохозяйственных задач. Проанализированы подходы заряда или замены бортовых аккумуляторов на сопровождающей роботизированной платформе. Имеющиеся прототипы сервисных роботизированных платформ отличаются сложностью внутренних механизмов, скоростью обслуживания, алгоритмами совместной работы платформы и летательного аппарата при посадке и обслуживании аккумулятора. Предложены концептуальная и графовая модели взаимодействия беспилотного летательного аппарата с наземной платформой, осуществляющей функции их транспортировки и технического обслуживания. Выполнены предварительные расчеты, создана программа для моделирования и визуализации процесса взаимодействия нескольких платформ и беспилотных летательных аппаратов при обработке аграрных объектов.**

Ключевые слова: **робототехнические сервисные платформы, беспилотные летательные аппараты, системы заряда аккумуляторов, системы замены аккумуляторов, аграрная робототехника.**

Введение

Роботизированные средства с разной степенью автономности функционирования все чаще применяются в сельскохозяйственном секторе, в том числе для посева зерновых, внесения удобрений, сбора урожая, распыления пестицидов. Совместное использование гетерогенных наземных и воздушных средств расширяет функциональные и сенсорные возможности роботизированной обработки сельскохозяйственных угодий и применения в других предметных областях [1–3]. В ряде случаев, например при обслуживании систем энергопитания и транспортировке воздушных средств, возникает задача физического соединения между летательным аппаратом и наземной платформой, функционирующих автономно [4, 5]. Сложность решения данной задачи связана с выполнением этапов посадки, фиксации и механизированной обработки аккумуляторов летательного аппарата на роботизированной наземной платформе [6].

В связи с изложенным исследование моделей и алгоритмов управления и взаимодействия гетерогенных сельскохозяйственных роботизированных комплексов является актуальным научным направлением, ориентированным на решение задачи увеличения времени функционирования беспилотных летательных аппаратов (БЛА) при длительных автономных работах на сельскохозяйственных полях, что в конечном итоге будет способствовать сокращению сроков обработки сельскохозяйственных объектов за счет автоматизации и применения роботизированных комплексов [7]. Далее рассмотрим несколько работ последних лет, описывающих существующие подходы к решению задачи сервисного обслуживания БЛА на наземных робототехнических станциях.

В работе [8] обосновывается, что поиск оптимальной траектории движения группы БЛА с учетом периодического обслуживания систем энергопитания на сопровождающей группе наземных мобильных заряжающих станций при выполнении некоторой миссии является NP-полной задачей. Для ее решения предлагается ряд модификаций методов линейного программирования и эвристических подходов.

В работе [9] БЛА рассматриваются как транспортные средства, оценивается стоимость их массового использования с учетом обслуживания на стационарных дозаправочных станциях, очередности обработки БЛА и динамических задержек в их прилете.

В работе [10] рассмотрены основные подходы к увеличению продолжительности полета БЛА за счет снижения энергозатрат, в частности: 1) использование новых легких материалов; 2) снижение энергопотребления бортовых устройств; 3) улучшение аэродинамических характеристик БЛА; 4) использование гибридных схем построения БЛА, в том числе с использованием аэростатической разгрузки БЛА самолетного и вертолетного типа; 5) динамическая маршрутизация полетов группы БЛА; 6) нетривиальное решение со сбрасыванием разряженных источников питания и снижения за счет этого массы БЛА.

При разработке модели управления многофункциональным механизмом соединения гетерогенных сельскохозяйственных роботизированных комплексов, а именно систем обслуживания аккумуляторов БЛА, необходимо учесть следующие функциональные и технические требования [11]:

– обеспечение идентифицируемого пространства для посадки на зарядной платформе: 1) наличие достаточной площади для посадки (более чем в 1,5 раза больше погрешности навигационной системы) пропорционально допустимому количеству и размеру проекции обслуживаемых БЛА; 2) наличие визуальной посадочной метки достаточного размера и сложности для распознавания системой навигации БЛА; 3) наличие коммуникационной системы для передачи текущего положения БЛА в навигационную систему;

– обеспечение зарядки батареи: 1) обеспечение безопасного электрического интерфейса между батареей на БЛА, электроникой БЛА, зарядным устройством на платформе и системой обнаружения БЛА; 2) идентификация факта приземления БЛА в правильном положении; 3) обеспечение заряда батареи; 4) контроль расхода электроэнергии и состояния внутренних источников питания;

– обеспечение питания зарядной платформы: 1) пополнение энергосистемы платформы; 2) преобразование энергии для использования при зарядке БЛА;

– обеспечение мобильности платформы: 1) легкая настройка при разворачивании системы на новом месте; 2) возможность транспортировки.

Перечисленные выше функциональные и технические требования будут учтены при разработке наземной роботизированной платформы, оснащенной посадочными площадками и многофункциональными механизмами захвата беспилотных летательных аппаратов и обслуживания их встроенных энергосистем, с учетом разработанных ранее технологических решений [12, 13].

Моделирование взаимодействия БЛА и сервисной платформы

Для формализации задачи взаимодействия гетерогенных сельскохозяйственных роботизированных комплексов и диспетчеризации процесса обслуживания беспилотных летательных аппаратов на наземных сервисных платформах была предложена концептуальная модель, включающая следующие основные сущности: сельскохозяйственное пространство, беспилотный летальный аппарат, наземная сервисная платформа, наземный центр управления. Предложенная модель служит основой для анализа способов управления и взаимодействия наземных сервисных платформ и беспилотных летательных аппаратов при решении аграрных задач. Основные элементы и связи предложенной концептуальной модели показаны на рисунке 1.

Сельскохозяйственное пространство упрощенно описывается основными параметрами по геометрической проходимости наземной техники по обрабатываемым угодьям, параметрами выращиваемых культур, необходимым перечнем аграрных задач и расписанием их выполнения, а также текущими погодными условиями.

Наземный центр управления осуществляет связь с гетерогенными роботизированными комплексами, расчет маршрутов их перемещения и планируемые сельскохозяйственные задачи. Также наземный центр управления включает в себя автоматизированное рабочее место оператора, но его функции с развитием и увеличением степени автономности роботизированных средств будет сокращаться от телеуправления к супервизорному управлению и мониторингу непредусмотренных ситуаций.

При описании наземной сервисной платформы необходимо знать текущий режим ее функционирования, координаты, скорость, состояние системы энергопитания, степень заполненности контейнера с ресурсами, а также текущее число БЛА, находящихся на платформе. Среди рассматриваемых параметров БЛА следует отметить также текущий режим его функционирования, координаты, скорость, степень разряженности аккумулятора, степень заполненности контейнера с ресурсами.

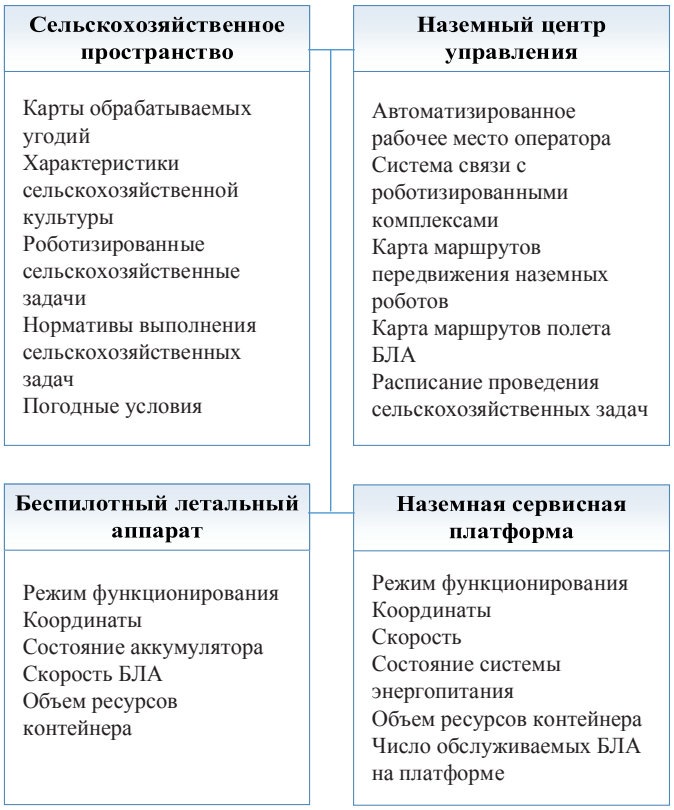

Рис. 1. Концептуальная модель системы управления гетерогенными сельскохозяйственными роботизированными комплексами

Далее проанализируем функционирование автоматической системы обслуживания аккумулятора БЛА на наземной сервисной платформе. Возможно применение двух вариантов обслуживания: с зарядкой аккумулятора или с заменой на новый. Системы с заменой аккумулятора значительно сокращают время подготовки БЛА для нового полета и увеличивают общее число БЛА, находящихся одновременно в воздухе, но стоимость подобных сервисных станций высока из-за сложного механизма манипулирования аккумуляторами. Система подзарядки аккумуляторов имеет более низкую стоимость по сравнению с системой замены аккумулятора за счет минимизации механических узлов конструкции. Далее рассмотрим основные этапы.

На рисунке 2 показана графовая модель процесса взаимодействия гетерогенных робототехнических комплексов, где T_F обозначает суммарное время, которое все БЛА находятся в полете; T_R – суммарное время нахождения всех БЛА на платформе сервисного обслуживания; T_C – суммарное время зарядки всех аккумуляторов от разряженного до

полного; T_I – суммарное время простоя всех БЛА, при котором аккумулятор не расходуется. Также следует выделить четыре основных цикла: цикл работы БЛА (L_{UAV}), цикл работы сервисной платформы (L_{PLAT}), цикл обслуживания аккумулятора (L_{BATT}) и цикл работы зарядного устройства (L_{CGR}). Определим длительность всех циклов:

$$T_{L_{UAV}} = \frac{T_F + T_R + T_I}{N_{UAV}};$$

$$T_{L_{PLAT}} = \frac{T_R}{N_{PLAT}};$$

$$T_{L_{BATT}} = \frac{T_C + T_R}{N_{BATT} - N_{UAV}};$$

$$T_{L_{CGR}} = \frac{T_R}{N_{CGR}},$$

где $N_{UAV}, N_{PLAT}, N_{BATT}, N_{CGR}$ – количество БЛА, сервисных платформ, аккумуляторов (всего, в том числе на БЛА и на платформах) и зарядных устройств соответственно.

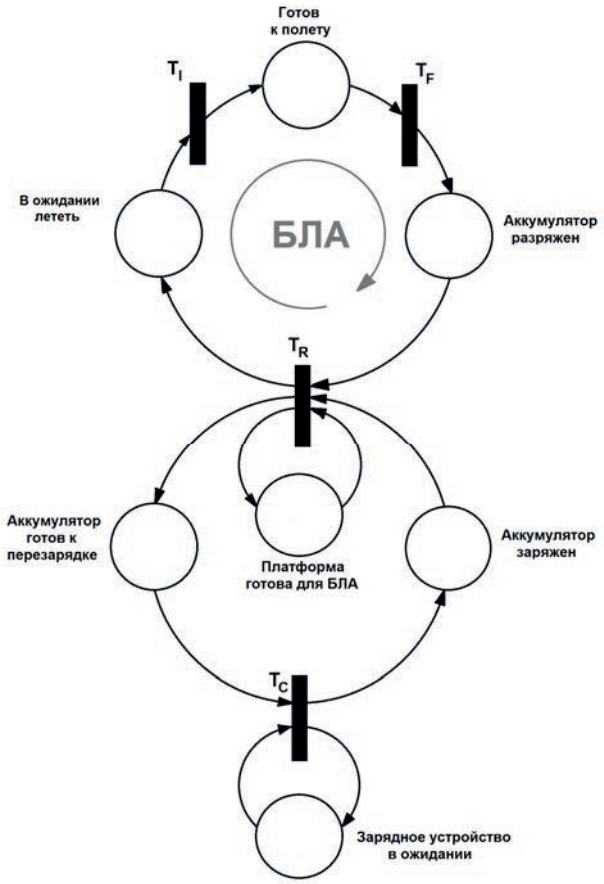

Рис. 2. Графовая модель взаимодействия БЛА с платформой, осуществляющей замену аккумулятора

В работе [14] обосновывается, что минимальное время цикла (максимальная производительность) системы составляет:

$$T_{CYC} = max\{T_{L_{UAV}}, T_{L_{PLAT}}, T_{L_{BATT}}, T_{L_{CGR}}\}.$$

Таким образом, каждый раз обслуживание аккумулятора требует времени T_{CYC}. Коэффициент эффективности сервисного обслуживания БЛА определяется отношением времени его нахождения в воздухе ко времени зарядки на сервисной платформе:

$$C_{CYC} = \frac{T_F}{T_{CYC}}.$$

Рассмотрим пример системы обслуживания БЛА, состоящей из 2 БЛА и одной платформы со следующими параметрами: $T_F = 20$ мин, $T_R = 3$ мин и $T_C = 0$ мин (поскольку производится замена аккумулятора, а не зарядка).

Установим, что $T_I = 0; N_{CGR} = 2; N_{UAV} = 2; N_{BATT} = 100$, тогда длительности циклов будут равны:

$$T_{L_{UAV}} = \frac{T_F + T_R + T_I}{N_{UAV}} = \frac{20 + 3 + 0}{2} = 11{,}5;$$

$$T_{L_{PLAT}} = \frac{T_R}{N_{PLAT}} = \frac{3}{1} = 3;$$

$$T_{L_{BATT}} = \frac{T_C + T_R}{N_{BATT} - N_{UAV}} = \frac{0 + 3}{100 - 2} = 0{,}03;$$

$$T_{L_{CGR}} = \frac{T_R}{N_{CGR}} = 1{,}5.$$

Минимальное время цикла (максимальная производительность) системы составляет:

$$T_{CYC} = T_{L_{UAV}} = 11{,}5.$$

Коэффициент эффективности сервисного обслуживания БЛА

$$C_{CYC} = \frac{T_F}{T_{CYC}} = \frac{20}{11{,}5} = 1{,}74.$$

Рассмотрим пример системы обслуживания БЛА, состоящей из 4 БЛА и одной платформы, со следующими параметрами: $T_F = 20$ мин, $T_R = 3$ мин и $T_C = 0$ мин.

Установим, что $T_I = 0; N_{CGR} = 4;\ N_{UAV} = 4;\ N_{BATT} = 100$, тогда длительности циклов будут равны:

$$T_{L_{UAV}} = \frac{T_F + T_R + T_I}{N_{UAV}} = \frac{20 + 3 + 0}{4} = 5{,}75;$$

$$T_{L_{PLAT}} = \frac{T_R}{N_{PLAT}} = \frac{3}{1} = 3;$$

$$T_{L_{BATT}} = \frac{T_C + T_R}{N_{BATT} - N_{UAV}} = \frac{0 + 3}{100 - 4} = 0{,}03;$$

$$T_{L_{CGR}} = \frac{T_R}{N_{CGR}} = 1.$$

Минимальное время цикла системы составляет:

$$T_{CYC} = T_{L_{UAV}} = 5{,}75.$$

Тогда коэффициент эффективности сервисного обслуживания БЛА будет

$$C_{CYC} = \frac{T_F}{T_{CYC}} = \frac{20}{5{,}75} = 3{,}48.$$

Таким образом, с увеличением количества активных БЛА возрастает коэффициент эффективности сервисного обслуживания БЛА. Также с увеличением числа единиц привлекаемой техники возрастает скорость обработки

сельскохозяйственного угодья, однако это сказывается и на себестоимости конечной продукции. Поэтому здесь следует находить компромисс между стоимостью и скоростью обработки в зависимости от прогнозируемых погодных условий и других аспектов аграрной деятельности.

На рисунке 3 показан пользовательский интерфейс разработанной программы AgroBotModeling, визуализирующей процесс моделирования обработки сельскохозяйственного угодья с различным числом платформ и БЛА. Интерфейс содержит четыре основных блока, отражающих параметры угодья, платформы и БЛА, а также элементы управления и индикации прошедшего времени.

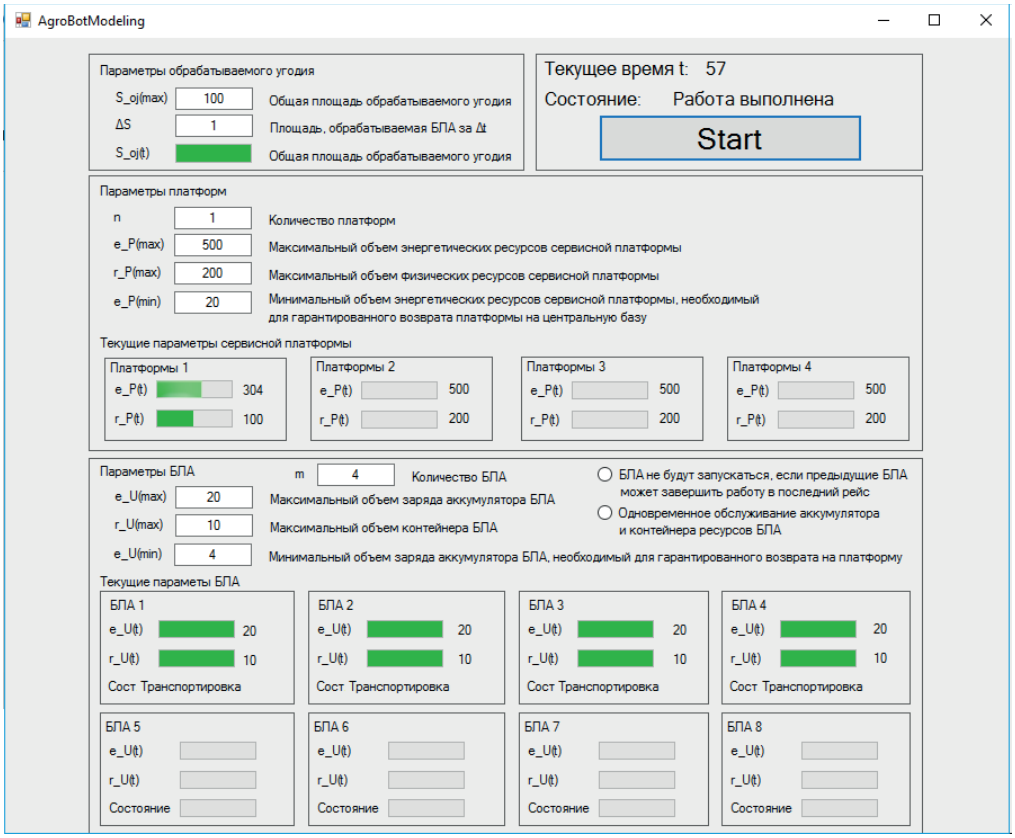

Рис. 3. Пользовательский интерфейс программы AgroBotModeling

Блок сельскохозяйственного угодья позволяет ввести начальные значения следующих параметров: 1) общая площадь обрабатываемого угодья; 2) площадь, обрабатываемая одним БЛА за время Δt. Также в блоке отображается текущее значение обработанной площади угодья.

В блоке сервисных наземных платформ задаются начальные значения следующих параметров: 1) количество платформ, используемых в обработке угодья; 2) максимальный объем энергетических ресурсов, размещаемых на одной сервисной платформе; 2) максимальный объем физических ресурсов, размещаемых на одной сервисной платформе. Также отображаются текущие значения следующих параметров: 1) объем оставшихся энергетических ресурсов на сервисной платформе; 2) объем оставшихся физических ресурсов на сервисной платформе.

В блоке БЛА задаются начальные значения следующих параметров: 1) количество БЛА, транспортируемых на одной сервисной платформе; 2) максимальный объем заряда аккумулятора БЛА; 3) максимальный объем контейнера с физическими ресурсами на БЛА; 4) набор опций, влияющих на порядок функционирования БЛА. Также отображаются текущие значения следующих параметров: 1) оставшийся заряд аккумулятора БЛА; 2) оставшийся объем физических ресурсов на БЛА; 3) текущий режим функционирования БЛА.

Число блоков платформ и БЛА может меняться в зависимости от первоначально заданных значений. На рисунке 4 представлены примеры диалоговых окон, где производится упрощенная визуализация процесса движения платформ и БЛА по обрабатываемому сельскохозяйственному угодью.

БЛА представлен в виде синего круга, сервисная платформа обозначена красным кругом, обрабатываемый аграрный объект – зеленым кругом. Кроме того, отображается текущий заряд аккумулятора БЛА и направление движения БЛА.

В дальнейшем планируется произвести интеграцию расчетной и визуализационной частей программы, а также, используя стоимостные критерии, предложенные в работе [15], модификацию разработанных программных средств для многокритериальной оценки количества необходимой техники для обработки сельскохозяйственного угодья.

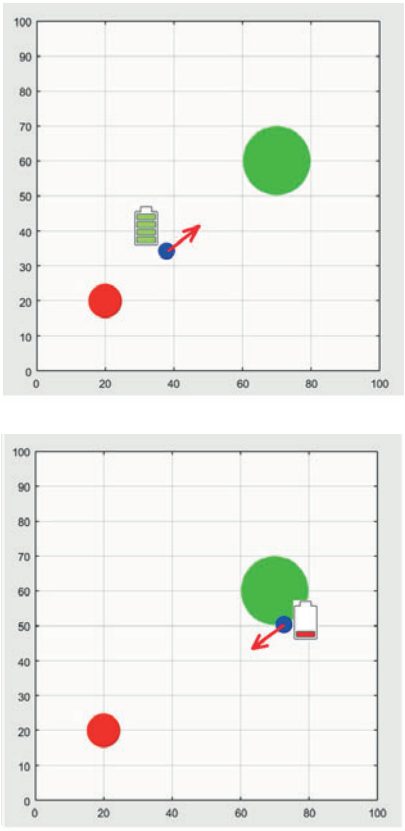

Рис. 4. Визуализация работы БЛА и платформы

Заключение

Представленное модельно-алгоритмическое обеспечение управления взаимодействием гетерогенных сельскохозяйственных роботизированных комплексов разработано с целью увеличения времени функционирования беспилотных летательных аппаратов в длительных автономных работах на сельскохозяйственных угодьях. Разработанная программа позволяет провести моделирование и визуализацию процесса взаимодействия разного количества платформ и беспилотных летательных аппаратов при решении аграрных задач на сельскохозяйственных угодьях различной площади. Моделирование количества используемой техники для обработки заданной площади позволяет выбрать оптимальное значение в зависимости от прогнозируемых погодных условий и других аспектов аграрной деятельности, влияющих на расчет себестоимости конечной продукции.

Библиографический список

1. Андреев В.П. Метод информационного взаимодействия для систем распределённого управления в роботах с модульной архитектурой / В.П. Андреев, П.Ф. Плетенев // Труды СПИИРАН. – 2018. – № 2 (57). – С. 134–160. – http://doi.org/10.15622/sp.57.6.

2. Лавренов Р.О. Разработка и имплементация сплайн-алгоритма планирования пути в среде ROS/Gazebo / Р.О. Лавренов, Е.А. Магид, Ф. Мацуно, М.М. Свинин, Д. Сутакорн // Труды СПИИРАН. – 2019. – Т. 18, № 1. – С. 57–84. – http://doi.org/10.15622/sp.18.1.57-84.

3. Ватаманюк И.В. Реконфигурация пространственного положения роя роботов / И.В. Ватаманюк, Г.Ю. Панина, А.Л. Ронжин // Управление Большими Системами. – 2015. – Вып. 58. – С. 285–305.

4. By Д.К. Обзор робототехнических захватов для физических манипуляций с аграрной продукцией / Д.К. By, О.Я. Соленая, А.Л. Ронжин // Тракторы и сельхозмашины. – 2017. – № 12. – С. 14–24.

5. Ронжин А.Л. Анализ проблем разработки беспилотных летательных манипуляторов и физического взаимодействия БЛА с наземными объектами / А.Л. Ронжин, В.В. Нгуен, О.Я. Соленая // Труды МАИ. – 2018. – № 98. – URL: http://trudymai.ru/published.php?ID=90439.

6. Нго Т.К. Анализ подвижных роботизированных платформ для обслуживания аккумуляторов беспилотных летательных аппаратов / Т.К. Нго, О.Я. Соленая, А.Л. Ронжин // Труды МАИ. – 2017. – № 95. – URL: http://trudymai.ru/published.php?ID=84444.

7. By Д.К. Анализ задач аграрной робототехники, решаемых посредством беспилотных летательных аппаратов / Д.К. By, В.В. Нгуен, О.Я. Соленая, А.Л. Ронжин // Агрофизика. – 2017. – № 3. – С. 57–65.

8. Maini P., Sujit P.B. On Cooperation Between a Fuel Constrained UAV and a Refueling UGV for Large Scale Mapping Applications, *In: 2015 International Conference on Unmanned Aircraft Systems (ICUAS), 9–12 June 2015, Denver, CO*. – Pp. 1370–1377. – http://doi.org/10.1109/ICUAS.2015.7152432.

9. Zhang K., Lu L., Lei C., Zhu H., Ouyang Y. Dynamic Operations and Pricing of Electric Unmanned Aerial Vehicle Systems and Power Networks, *Transportation Research Part C: Emerging Technologies*, 2018, Vol. 92. – Pp. 472–485. http://doi.org/10.1016/j.trc.2018.05.011.

10. Фетисов В. С. Автоматические сервисные станции для обслуживания электрических беспилотных летательных аппаратов : монография / В.С. Фетисов, А.Е. Артемьев, Д.Ф. Муфаззалов / Уфимский гос. авиац. техн. ун-т. – М. : Инновационное машиностроение, 2017. – 252 с.

11. Солёная О.Я. Анализ требований и ограничений мобильных электромеханических систем обслуживания аккумуляторов беспилотных летательных аппаратов / О.Я. Солёная, Т.К. Нго, А.Л. Ронжин // Вопросы электромеханики. Труды ВНИИЭМ. – 2017. – Т. 159. – № 4. – С. 42–48.

12. Яковлев Р.Н. Параметры и алгоритмы управления динамическим механическим движением многозвенных робототехнических систем / Р.Н. Яковлев, П.М. Черноусова, К.Д. Крестовников, А.В. Денисов // Известия Кабардино-Балкарского научного центра РАН. – 2018. – № 6–3 (86). – С. 227–242.

13. Нго Т.К. Функциональная модель взаимодействия БЛА с наземной роботизированной платформой при решении сельскохозяйственных задач / Т.К. Нго, В.В. Нгуен, И.Ю. Харьков, Е.Е. Усина, О.О. Шумская // Известия Кабардино-Балкарского научного центра РАН. – 2018. – № 6–3 (86). – С. 41–50.

14. Suzuki K. A. O., Filho P.K., Morrison J.R. Automatic Battery Replacement System for UAVs: Analysis and Design, *Journal of Intelligent & Robotic Systems*, 2012, Vol. 65, Is. 1-4. – Pp. 563–586. – http://doi.org/10.1007/s10846-011-9616-y.

Фетисов В.С., Ахмеров Ш.Р. Сравнительная характеристика автоматических зарядных и обменно-зарядных станций для обслуживания малых электрических беспилотных летательных аппаратов // Авиакосмическое приборостроение. – 2019. – № 2. С. 3–10.

Conceptual and Graph Models of Interaction of Unmanned Aerial Vehicle with Ground Robotic Service Platform

PhD student, T.Q. Ngo
St. Petersburg State University of Aerospace Instrumentation,
St. Petersburg, Russia
quoctienbn@gmail.com

Abstract. **The task of increasing the operating time of unmanned aerial vehicles in solving agricultural operations is considered. Several approaches to charge or replace onboard batteries on the accompanying robotic platform are analyzed. The existing prototypes of service robotic platforms are distinguished by the complexity of the internal mechanisms, the speed of service, the algorithms of the platform and unmanned aerial vehicles working together during landing and battery maintenance. Conceptual and graph models of the interaction of an unmanned aerial vehicle with a ground platform that performs the functions of their transportation and maintenance are proposed. Preliminary calculations have been performed and a program has been created for modeling and visualizing the process of interaction between several platforms and unmanned aerial vehicles during the processing of agricultural objects.**

Keywords: **robotic service platforms, unmanned aerial vehicles, battery charging systems, battery replacement systems, agricultural robotics.**

References

1. Andreev V.P., Pletenev P.F. Method of Information Interaction for Distributed Control Systems of Robots with Modular Architecture [Metod informatsionnogo vzaimodeystviya dlya sistem raspredelennogo upravleniya v robotakh s modul'noy arkhitekturoy], *SPIIRAS Proceeding [Trudy SPIIRAN]*, 2018, No. 2 (57). – Pp. 134–160. http://doi.org/10.15622/sp.57.6.

2. Lavrenov R.O., Magid E.A., Matsuno F., Svinin M.M., Suthakorn J. Development and Implementation of Spline Based Path Planning Algorithm in ROS/Gazebo Environment [Razrabotka i implementatsiya splayn-algoritma planirovaniya puti v srede ROS/Gazebo], *SPIIRAS Proceeding [Trudy SPIIRAN]*, 2019, Vol. 18, No. 1. – Pp. 57–84. http://doi.org/10.15622/sp.18.1.57-84.

3. Vatamaniuk I.V., Panina G.Yu., Ronzhin A.L. Reconfiguration of Robot Swarm Formation [Rekonfiguratsiya prostranstvennogo polozheniya roya robotov], *Large System Management [Upravlenie Bol'shimi Sistemami*], 2015, Is. 58. – Pp. 285–305.

4. Vu D.Q., Solenaya O.Ya., Ronzhin A.L. Over-View of Robotic Grippers for Physical Manipulation with Agricultural Products [Obzor robototekhnicheskikh zakhvatov dlya fizicheskikh manipulyatsiy s agrarnoy produktsiey], *Tractors and Agricultural Machinery [Traktory i sel'khozmashiny]*, 2017, No. 12. – Pp. 14–24.

5. Ronzhin A.L., Nguen V.V., Solenaya O.Ya. Analysis of the Problems of Unmanned Flying Manipulators Development and UAV Physical Interaction with Ground Objects [Analiz problem razrabotki bespilotnykh letatel'nykh manipulyatorov i fizicheskogo vzaimodeystviya BLA s nazemnymi ob`ektami], *Trudy MAI [Trudy MAI]*, 2018, No. 98. Available at: http://trudymai.ru/published.php?ID=90439.

6. Ngo T.Q., Solenaya O.Ya., Ronzhin A.L. Analysis of Mobile Robotic Platforms Used to Batteries Service of Unmanned Aerial Vehicles in Autonomous Missions [Analiz podvizhnykh robotizirovannykh platform dlya obsluzhivaniya akkumulyatorov bespilotnykh letatel'nykh apparatov], *Trudy MAI [Trudy MAI]*, 2017, No. 95. Available at: http://trudymai.ru/published.php?ID=84444.

7. Vu D.Q., Nguyen V.V., Solenaya O.Ya., Ronzhin A.L. Analysis of Agricultury Robotics Tasks Solved by Using Unmanned Aerial Vehicles [Analiz zadach agrarnoy robototekhniki, reshaemykh posredstvom bespilotnykh letatel'nykh apparatov], *Agrophysics [Agrofizika]*, 2017, No. 3. – Pp. 57–65.

8. Maini P., Sujit P.B. On Cooperation Between a Fuel Constrained UAV and a Refueling UGV for Large Scale Mapping Applications, *In: 2015 International Conference on Unmanned Aircraft Systems (ICUAS), 9–12 June 2015, Denver, CO*. – Pp. 1370–1377. – http://doi.org/10.1109/ICUAS.2015.7152432.

9. Zhang K., Lu L., Lei C., Zhu H., Ouyang Y. Dynamic Operations and Pricing of Electric Unmanned Aerial Vehicle Systems and Power Networks, *Transportation Research Part C: Emerging Technologies*, 2018, Vol. 92, Pp. 472–485. – http://doi.org/10.1016/j.trc.2018.05.011.

10. Fetisov V.S., Artemyev A.E., Mufazzalov D.F. Automatic service stations for maintenance of electric unmanned aerial vehicles: Monograph [Avtomaticheskie servisnye stantsii dlya obsluzhivaniya elektricheskikh bespilotnykh letatel'nykh apparatov: Monografiya], Moscow, Innovative Engineering Publishing House, 2017. – 252 p.

11. Solenaya O.Ya., Ngo T.Q., Ronzhin A.L. Analysis of Requirements and Limitations for Mobile Electromechanical Service Systems of Storage Batteries of Unmanned Aerial Vehicles [Analiz trebovaniy i ogranicheniy mobil'nykh elektromekhanicheskikh sistem obsluzhivaniya akkumulyatorov bespilotnykh letatel'nykh apparatov], *Electromechanical Matters. VNIIEM Studies [Voprosy elektromekhaniki. Trudy VNIIEM]*, 2017, Vol. 159, No. 4. – Pp. 42–48.

12. Yakovlev R.N., Chernousova P.M., Krestovnikov K.D., Denisov A.V. Parameters and Control Algorithms for Dynamic Mechanical Motion of The Multi-Link Robotic Systems [Parametry i algoritmy upravleniya dinamicheskim mekhanicheskim dvizheniem mnogozvennykh robototekhnicheskikh sistem], *Proceedings of the Kabardino-Balkarian Research Center of the RAS [Izvestiya Kabardino-Balkarskogo nauchnogo tsentra RAN]*, 2018, No. 6-3 (86). – Pp. 227–242.

13. Ngo T.Q. Nguyen V.V., Kharkov I.Yu., Usina E.E., Shumskaya O.O. Functional Model of Interaction Between an UAV and a Land-Based Robotic Platform in Agricultural Tasks [Funktsional'naya model' vzaimodeystviya BLA s nazemnoy robotizirovannoy platformoy pri reshenii sel'skohozyaystvennykh zadach], *Proceedings of the Kabardino-Balkarian Research Center of the RAS [Izvestiya Kabardino-Balkarskogo nauchnogo tsentra RAN]*, 2018, No. 6-3 (86). – Pp. 41–50.

14. Suzuki K. A. O., Filho P.K., Morrison J.R. Automatic Battery Replacement System for UAVs: Analysis and Design, *Journal of Intelligent & Robotic Systems*, 2012, Vol. 65, Is. 1-4. – Pp. 563–586. – http://doi.org/10.1007/s10846-011-9616-y.

15. Fetisov V.S., Akhmerov Sh.R. Comparative Estimation of Automatic Charging and Exchange-Charging Stations for Maintenance of Small Electric Unmanned Aerial Vehicles [Sravnitel'naya kharakteristika avtomaticheskikh zaryadnykh i obmenno-zaryadnyh stantsiy dlya obsluzhivaniya malykh elektricheskikh bespilotnykh letatel'nykh apparatov], *Aerospace Instrument-Making [Aviakosmicheskoe priborostroenie]*, 2019, No. 2. – Pp. 3–10.

Комплекс программных средств разработки и верификации требований и проектных решений АСУ объектами транспортной инфраструктуры

А.В. Баев, канд. техн. наук А.В. Самонов
Военно-космическая академия имени А.Ф.Можайского
Санкт-Петербург, Россия
a.baih@mail.ru, a.samonov@mail.ru

Аннотация. **Представлены состав, структура и способы применения комплекса программных средств, обеспечивающего разработчиков автоматизированных систем управления организационно-техническими комплексами и объектами транспортной инфраструктуры интеллектуальными средствами поддержки при обосновании требований и проектировании систем. Комплекс средств разработан с целью устранения логических и семантических разрывов между представлениями о разрабатываемой системе у участников данного процесса: заказчика и пользователей, системных архитекторов и проектировщиков, инженеров и программистов. Это достигается посредством реализации программно-управляемого процесса разработки и верификации формальных моделей требований и проектных решений в единой для всех его участников модельно-языковой и информационно-программной среде. Построение моделей осуществляется автоматизированным способом на основе шаблонов формального описания требований к функциональным и эксплуатационным характеристикам создаваемых систем, разработанных с использованием сетей Петри, временных автоматов, языков моделирования и проектирования SysML, FUML, OCL. Для валидации и верификации комплекса требований и проектных решений разработаны метамодель и алгоритм построения и анализа трассы выполнения модели в среде виртуальной машины VM FUML. Предложены способы интеграции и использования для автоматизированного тестирования моделей комплекса требований и проектных решений специализированных средств верификации CPN Tools, Rodin, SPIN и Modelica. Данный комплекс обеспечивает более эффективное взаимодействие заказчика и исполнителя как при разработке требований, так и при проектировании системы, раннее обнаружение и устранение дефектов посредством реализации автоматизированных процедур верификации, валидации и коррекции. Это позволит повысить качество требований и проектных решений, а также улучшить экономические показатели в части снижения финансовых и временных затрат, связанных с выполнением дополнительных работ как в случае обнаружения дефектов, так и при изменении требований или условий эксплуатации.**

Ключевые слова: **автоматизированные системы управления, валидация и верификация, временные автоматы, проектирование и моделирование, сети Петри, функциональные и эксплуатационные требования.**

Введение

Разработка автоматизированных систем управления организационно-техническими комплексами и объектами транспортной инфраструктуры (АСУ КОТИ) представляет собой сложную и ресурсоемкую задачу, результат решения которой не всегда удовлетворяет заданным требованиям и укладывается в рамки выделенных временных и финансовых ресурсов. Одной из основных причин этого является наличие логических и семантических разрывов между представлениями о разрабатываемой системе у участников данного процесса: заказчика и пользователей, конструкторов и проектировщиков, инженеров и программистов. Это приводит к тому, что связи между тремя основными артефактами жизненного цикла разработки АСУ КОТИ являются несистемными, фрагментарными, неточными и несогласованными. Данное обстоятельство существенно затрудняет контроль их качества и проверку соответствия друг другу.

Одним из основных способов устранения этих разрывов является создание и применение комплексных программных средств (КПС), построенных и функционирующих на основе принципов и технологий модельно-ориентированной системной инженерии (МОСИ, Model-based systems engineering, MBSE). Примерами таких систем являются: IBM Rational Rhapsody Developer [1], Sparx Enterprise Arhitect [2], MASIW [3]. Данные системы в настоящее время используются в авиационной и космической отрасли, автомобилестроении, энергетике, в военно-промышленном комплексе. Для того чтобы эти средства приносили максимальную пользу, в них должна выдерживаться «золотая середина» между универсальностью и ориентированностью на определенную предметную область. При решении данной задачи необходимо, с одной стороны, использовать и опираться на общепринятые международные и отечественные стандарты, с другой стороны, разработать и применять предметно-ориентированные онтологии, описывающие объекты контроля и способы управления ими.

В данной статье описан комплекс программных средств, обеспечивающий единую модельно-языковую и

информационно-программную среду для разработки и контроля качества основных артефактов жизненного цикла АСУ КОТИ: комплекса требований и проектных решений.

Анализ технологий и средств промышленной разработки автоматизированных систем управления сложными комплексами и объектами

Методологическую и технологическую базу МОСИ образуют методические документы и спецификации, разработанные под эгидой таких организаций, как Международная организация по стандартизации (International Organization for Standardization, ISO), Международная электротехническая комиссия (International Electrotechnical Commission, IEC), Компьютерная ассоциация Института инженеров электротехники и электроники (Institute of Electrical and Electronics Engineers Computer Society, IEEE CS), Международный совет по системной инженерии (International Council on Systems Engineering, INCOSE), Международная ассоциация по управлению проектами (International Project Management Association, IPMA), Международный союз электросвязи (International Telecommunication Union, ITU), некоммерческое объединение OMG (Object Management Group) и многие другие. В разработке этих документов активное участие принимают научно-исследовательские организации (DISA, INCOSE, NIST и др.) и промышленные компании (AT&T, IBM, Oracle, Microsoft, Cisco Systems, NASA и др.). В нашей стране активные исследования в этой области ведутся в таких организациях, как ИСП РАН, ВМК МГУ им. М.В. Ломоносова, СПбГУ, Новосибирском ГТУ, ВКА им. А.Ф. Можайского и др.

МОСИ включает следующие основные технологии: разработка на основе моделей (Model driven development, MDD), архитектура на основе моделей (Model driven architecture, MDA), тестирование на основе моделей (Model based testing, MBT) [4, 5]. Реализация данных технологий предполагает, что создаваемые и используемые на всех этапах жизненного цикла (ЖЦ) системы артефакты (комплекс требований, технический проект, программные и технические реализации) представляются в виде формальных моделей, которые подвергаются процедурам верификации и валидации с помощью специальных автоматизированных методов и средств. По результатам проверки качества этих артефактов принимается решение об их пригодности для использования на следующих этапах ЖЦ системы или направлении на доработку.

В настоящее время на сайте OMG [6] опубликовано более 230 методических документов и спецификаций. С точки зрения рассматриваемых в настоящей статье вопросов наиболее важными из них являются следующие спецификации: UML (Unified Modeling Language), XMI (XML Metadata Interchange), SysML (System Modeling Language), OCL (Object Constraint Language), UTP (UML Testing Profile), ALF (Action Language for Foundational UML), FUML (Semantics of a Foundational Subset for Executable UML Models), ReqIF (Requirements Interchange Format).

В настоящее время наиболее перспективными и активно используемыми языками проектирования сложных автоматизированных информационных систем являются SysML, FUML, OCL. С их помощью можно решать задачи анализа, специфицирования, проектирования и тестирования систем, состоящих из аппаратных средств, программного обеспечения, данных, персонала, процедур, средств и других искусственных и природных систем. Эти языки используются в целом ряде систем автоматизированного проектирования – как промышленных (IBM Rational Rhapsody Developer, Sparx Enterprise Arhitect), так и исследовательских (Modelio [7], Eclipse Papyrus [8], GEMOC Studio [9]). Анализ результатов практического применения этих систем, проблемные вопросы, предложения по их разрешению и совершенствованию методов и средств представлены в целом ряде научно-технических публикаций и отчетов. Ниже представлен обзор наиболее интересных из них.

В первую очередь следует отметить монографии и практические руководства ряда известных специалистов в данной области. В фундаментальном труде известного сербского ученого и эксперта в области MBSE, профессора Белградского университета Д. Милисева (Dragan Milicev) [10] изложены методы, рекомендации и примеры использования для промышленной разработки сложных программно-технических систем (СПТС) исполняемого языка моделирования FUML (Foundational UML). В монографии [11], разработанной коллективом авторов в составе S. Friedenthal, A. Moore, R. Steiner, являющихся активными участниками разработки целого ряда методических документов и спецификаций OMG, а также применяющих свои знания на практике в таких компаниях, как Lockheed Martin и Raytheon Company, в сжатой и хорошо иллюстрированной практическими примерами форме не только изложены приемы и способы применения конструкций и механизмов языка SysML, но и дано объяснение его философии и принципов. Много полезной информации о способах применения для разработки СПТС языка системного моделирования SysML представлено в монографии Lenny Delligatti, имеющего богатый опыт разработки СПТС в компании Lockheed Martin [12].

В статье [13] представлена методика формирования UML-профиля, с помощью которого разрабатывается формальное описание требований к таким характеристикам качества ПО, как конфиденциальность, целостность, доступность и надежность. Профиль построен на основе языка OCL, реализован с помощью средств фреймворка Eclipse EMF. Разработанный с помощью данного профиля комплекс требований может быть верифицирован на предмет его полноты, корректности и согласованности. В случае обнаружения нарушения этих свойств определяется тип дефекта и локализуются содержащие его части модели требований. Прошедший проверку комплекс требований может быть использован для автоматической генерации элементов проекта модели: например, он генерирует наблюдаемые и управляемые интерфейсы из имен ассоциаций.

Как показал проведенный анализ, в настоящее время ведутся активные исследования и работы по созданию предметно-ориентированных языков моделирования (eXecutable Domain-Specific Modeling Languages, xDSML), использующие и развивающие ключевую концепцию МОСИ – метамоделирование. Результатом таких работ является, в частности, разработанный группой исследователей из Франции и Германии программный комплекс

GEMOC Studio [9]. Данный комплекс предназначен для создания xDSML на основе языков моделирования SysML/FUML, разработки исполняемых моделей на этом языке, их валидации и верификации с помощью графического аниматора и интеллектуального механизма трассировки. С помощью комплекса можно осуществлять мониторинг состояния анализируемых моделей (переходов, событий, значений переменных) во время их выполнения в виртуальной машине. В работе [14] дано описание используемой в данном средстве многомерной предметно-ориентированной метамодели трассировки, обеспечивающей более высокую производительность по сравнению со стандартной метамоделью UML за счет исключения из обработки избыточных данных.

При разработке и верификации распределенных динамических дискретных систем, какими являются АСУ КОТИ, полезно использовать методы и средства имитационного моделирования. Одним из наиболее эффективных и адекватных средств формального описания поведения подобных систем, анализа причинно-следственных связей и отслеживания происходящих при этом событий является математический аппарат и язык сетей Петри (СП). Язык СП по своей мощности не уступает контекстно-свободным языкам, класс контекстно-зависимых языков включает класс языков СП, а язык регулярных выражений и автоматных языков – суть подкласс языков СП. Сети Петри являются теоретической основой для формального представления диаграмм активности языка UML 2.5, их автоматической трансляция в сети Петри и обратно. Примеры использования сетей Петри для разработки представлены в целом ряде публикаций [15–18].

В частности, в работе [15] дано описание методики проектирования программного обеспечения с использованием UML-диаграмм последовательности в формате *.xmi* и представлен способ их автоматической трансляции в формат *.cpn*, используемый для описания сетей Петри. Результатом применения данного способа являются иерархические сети Петри, с помощью анализа которых осуществляется верификация проекта программного обеспечения (ПО), представленного в виде UML-диаграмм.

Для описания поведения динамических систем, кроме диаграмм активности, в языках моделирования SysML и FUML также активно используются диаграммы состояний и последовательностей. Математической основой для них являются автоматные модели и темпоральные логики.

В статье [19] дано описание двух методов реализации автоматической проверки нагруженных систем реального времени с использованием сценариев. В первом случае система моделируется как сеть временных автоматов (ВА). Во втором – с помощью набора диаграмм динамических последовательностей (LSCs) и требований в виде отдельной анализируемой диаграммы LSC.

Авторы статьи разработали временные (темпоральные) расширения для подмножества ядра языка LSC и определили его семантику на основе трассировки. Анализируемая диаграмма LSC транслируется в ее поведенческий эквивалент в нотации диаграммы ВА. Верификация корректности модели осуществляется посредством моделирования диаграммы ВА в режиме реального времени с использованием аппарата темпоральной логики CTL (Computational Tree Logic) с последующим сравнением полученного результата с эталоном. Оба метода реализованы с помощью средств инструмента UPPAAL.

Проведенный анализ нормативно-методических документов, научно-технических публикаций и систем позволил сделать вывод о том, что для формального описания комплекса требований и проектных решений при разработке АСУ КОТИ целесообразно использовать следующие средства и возможности языков SysML, FUML и OCL:

- для описания состава и структуры системы – диаграммы пакетов (Package), внешнего и внутреннего представления блоков (Block Definition, Internal Block), активных и пассивных классов (Active Class, Passive Class);
- для описания поведения и способов взаимодействия ее компонентов – диаграммы вариантов использования (UseCase), деятельности (Activity), состояний (State Machine) и взаимодействия (Sequience);
- для описания требований к эксплуатационным характеристикам (производительности, надежности, защищенности и др.) – средства языка OCL;
- для описания связей между структурными, поведенческими, ограничительными и другими аспектами и характеристиками – отношения обобщения (generalization), агрегации (aggregation), композиции (composition), ассоциации (association), зависимости (dependency), представленные с помощью средств языка OCL.

Для валидации и верификации формальных моделей комплекса требований и проектных решений целесообразно использовать виртуальную машину VM FUML, а также такие инструменты, как CPN Tools [20], Rodin [21], SPIN [22, 23].

Таким образом, для разработки моделей и алгоритмов формального описания основных артефактов процесса создания АСУ КОТИ предлагается использовать следующие математические модели и языки моделирования:

- сети Петри, временные автоматы;
- языки визуального моделирования и проектирования SysML, FUML, OCL, ALF.

В качестве автоматизированных средств верификации формальных моделей комплекса требований и проектных решений были использованы: VM FUML, CPN Tools, SPIN, Rodin. На основе представленных выше моделей, языков и средств создается единая для всех участников разработки АСУ КОТИ модельно-языковая и информационно-программная среда и реализуется последовательно-итерационный, программно-управляемый процесс проектирования и разработки.

Программные средства разработки и верификации формальной модели комплекса требований к АСУ КОТИ

Комплекс требований к АСУ КОТИ включает две основные группы требований: требования к ее функциональным возможностям и требования к качеству и условиям их реализации.

Для обеспечения автоматизированного построения формальных описаний этих требований в среде визуального моделирования были разработаны специальные шаблоны (стереотипы), представляющие собой расширения

диаграмм вариантов использования (UseCase), деятельности (Activity), состояний (State Machine), взаимодействия (Sequence), внешнего (block definition diagram) и внутреннего (internal block diagram) представления компонентов системы, активных (Active Class) и пассивных (Passive Class) классов. В частности, для формального описания требований к функциональным возможностям самого верхнего уровня их представления разработан шаблон «function requirement», который является расширением SysML-диаграммы вариантов использования – UseCase diagram (см. таблицу).

Шаблон представления требований
к функциональным возможностям АСУ КОТИ

Поле шаблона	Тип данных	Описание поля шаблона	Обязательность задания
Nameuc	String	Имя (идентификатор требования)	Обязательный
description	String	Краткое описание функции	Обязательный
Actor	actor (actors)	Пользователь функции	Обязательный
Subject	class (block)	Класс (блок, модуль) системы, реализующий функцию	Обязательный
basic_scenarios	scenarios (control &object_flow)	Основной сценарий выполнения функции	Опциональный
alter_scenarios	scenarios (control &object_flow)	Альтернативный сценарий выполнения функции	Опциональный

Каждый *i*-й вариант использования представляет собой функциональное требование и описывается следующим образом:

$$uc_i = (nameuc_i,\ description_i,\ actor_k, subject_j,\ basic_scenario_i,\ alter_scenario_i).$$

Функциональные блоки системы (*subject*) описываются посредством диаграмм пакетов (*Package*), внешнего и внутреннего представления блоков (*Block Definition*, *Internal Block*).

Основной и альтернативные сценарии реализации функций представляются в форме активных и пассивных классов (*Active Class*, *Passive Class*). Для описания методов классов используются диаграммы деятельности (*Activity*), состояний (*State Machine)* и взаимодействия (*Sequence*).

Следующим шагом построения модели комплекса требований является разработка требований к качеству реализации каждой функции, т. е. к эксплуатационным характеристикам АСУ КОТИ:

$$d_reqs = \left(r_1^{f_i},\ r_2^{f_i}, r_3^{f_i}, r_4^{f_i} \ldots\right),$$

где $r_1^{f_i}$ – требования к оперативности исполнения функции f_i;

$r_2^{f_i}$ – требования к производительности (например, объемы хранимых, обрабатываемых и передаваемых данных, количество пользователей, количество и размеры запросов в единицу времени и др.);

$r_3^{f_i}$ – требования к надежности (коэффициент готовности, время безотказной работы, время восстановления и др.);

$r_4^{f_i}$ – требования к защищенности.

Требования к эксплуатационным характеристикам (производительности, надежности, защищенности, совместимости и др.) описываются с помощью средств языка OCL. Для каждой из этих характеристик разработаны соответствующие шаблоны, которые используются в подключаемых к инструменту Eclipse Papyrus модулях (plugins) для реализации программно-управляемого процесса их формального описания и верификации. Затем с помощью отношений обобщения (*generalization*), агрегации (*aggregation*), композиции (*composition*), ассоциации (*association*), зависимости (*dependency*) устанавливаются связи между структурными, поведенческими, ограничительными и другими аспектами и характеристиками модели, которые также представляются с помощью средств языка OCL.

Все АСУ КОТИ с точки зрения особенностей алгоритма их функционирования можно разделить на три группы. Первую группу образуют системы, выполняющие свои функции в соответствии с определенным алгоритмом, имеющим только вход и выход. Во вторую группу входят реагирующие системы, выполняющие определенные операции и изменяющие свое состояние в зависимости от внешних воздействий. Третью группу образуют системы, взаимодействующие посредством обмена сообщениями.

Для формального описания функционирования систем первой группы наиболее адекватным средством являются диаграммы деятельности (activity), основанные на математическом аппарате сетей Петри. СП представляется четверкой

$$N = (P, T, F, m_I),$$

где $P = \{p_1, p_2, \ldots, p_n\}$ – множество мест;
$T = \{t_1, t_2, \ldots, t_n\}$ – множество переходов, таких, что $P \cap T = \emptyset$;

$F \subseteq P \times T \cup T \times P$ – матрица инцидентности;

$m_I : P \to N$ – начальная маркировка.

Основными свойствами сетей Петри, которые следует использовать для верификации формальных моделей требований и архитектуры, являются: живость (отсутствие блокировок и зацикливаний) и ограниченность. СП обладает свойствами живости, если она обладает свойствами достижимости и покрываемости. *Достижимость* – это наличие путей перехода в заданные состояния из начального состояния $m \in [m_I\rangle$. *Покрываемость* – возможность перехода в заданные состояния из других состояний $m'' \in [m'\rangle$. Сеть Петри является *k*-ограниченной, если и только если все маркировки $m \in [m_I\rangle$ и для всех $p \in P: m(p) \leq k$.

Обладающая данным свойством СП и, как следствие, описываемая с ее помощью система будут функционировать в определенных границах. Частным случаем свойства ограниченности является *безопасность*. Сеть Петри

(P, T, F, m_I) является безопасной, если все маркировки $m \in [\, m_I\rangle$ и для всех $p \in P$: $m(\, p) \leq\ 1$.

Важным достоинством использования сетей Петри является наличие целого ряда автоматизированных средств верификации построенных с их помощью моделей. Одним из наиболее развитых и эффективных является CPN Tools, который и предлагается использовать в нашем случае для анализа формальных моделей требований и архитектуры АСУ КОТИ.

Формальная модель комплекса требований подвергается процедурам валидации и верификации. Процедура валидации заключается в оценивании комплекса требований на предмет его полноты и корректности и выполняется как программными средствами, так и неформальной экспертизой специалистов в данной предметной области.

При верификации проверяются такие свойства комплекса требований, как непротиворечивость, системность, неизбыточность, безопасность, живость, отсутствие взаимоблокировок, невыполнимых операций, зацикливаний.

Состав и структура комплекса моделей и программных средств, с помощью которых реализуются процедуры верификации и валидации формальной модели требований к АСУ КОТИ, представлены на рисунке 1.

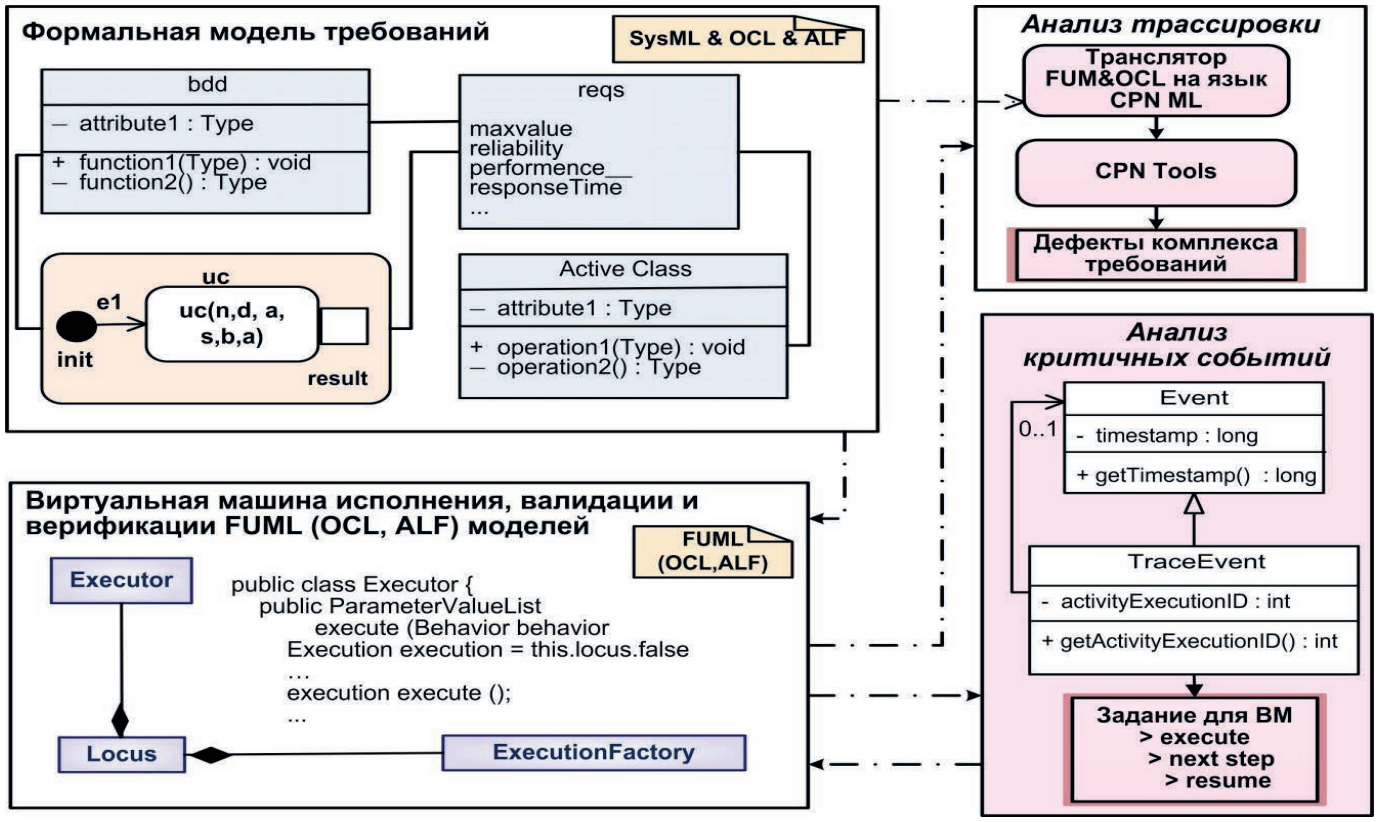

Рис. 1. Состав и структура комплекса моделей и программных средств реализации процедур верификации и валидации формальной модели требований к АСУ КОТИ

Процедура тестирования комплекса требований включает:

- валидацию комплекса требований посредством исполнения его формальной модели в среде виртуальной машины VM FUML и анализа ее состояния в контрольных точках и при обработке критичных событий;
- анализ поведенческой части модели, представленной диаграммами активности, с помощью CPN Tools на предмет выявления тупиковых и мертвых состояний, стабильности, ограниченности и безопасности;
- анализ поведенческой части модели, представленной диаграммами состояний, с помощью инструмента Rodin на предмет выявления тупиковых и мертвых состояний, необрабатываемых стимулов и др.;
- верификацию качества реализации в диаграмме классов базовых механизмов объектно-ориентированной парадигмы (инкапсуляции, наследования, полиморфизма, абстракции, обмена сообщениями) посредством расчета и оценивания метрик, характеризующих эти механизмы.

Для реализации процедуры валидации модели требований посредством анализа ее состояния в контрольных точках была разработана представленная на рисунке 2 метамодель построения трассы ее исполнения в среде VM FUML. Трасса (класс *Trace*) формируется из конструкций, представленных в трех пакетах: Steps, States и PNnet, описывающих контрольные точки, состояния модели в них и абстрактный синтаксис СП соответственно. На рисунке 3 приведен фрагмент разработанного на основе данной метамодели

алгоритма построения трассы выполнения и анализа модели комплекса требований в среде VM FUML. В отличие от стандартного способа, в данном алгоритме исключены повторные проверки неизменяемых данных и обеспечена возможность его многократного использования.

Для того чтобы требования, представленные в виде диаграмм активностей и состояний, можно было верифицировать средствами CPN Tools и Rodin, они предварительно транслируются на язык CPN ML и EventB соответственно с помощью разработанных для этого модулей Eclipse Papyrus.

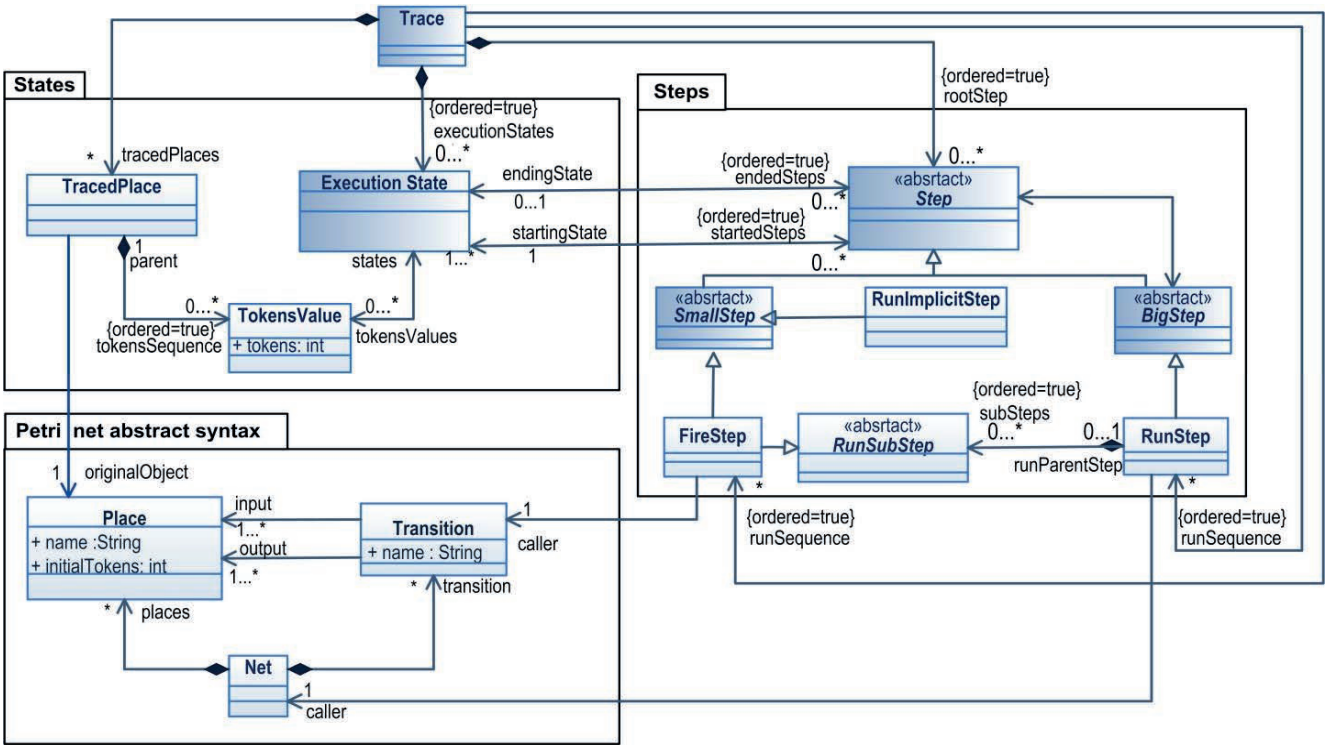

Рис. 2. Метамодель построения трассы исполнения формальной модели комплекса требований в среде VM FUML

```
begin
   c_trace, c_exeState, c_step, c_smallStep, c_bigStep ← createBaseGeneric-
Classes()
  mm_trace ← {c_trace, c_exeState, c_step, c_smallStep, c_bigStep}
  foreach c_exe ∈ {c ∈ mm_exe | containsDynamicProperties(c)}
    do
     c_traced ← createClass()
     mm_trace ← mm_trace ∪ {c_traced}
     c_trace.createReferenceTo(c_traced, [0..*], unordered)
     if containsStaticProperties(c_exe) then
        c_orig ← getClassFromAbstractSyntax(c_exe)
      c_traced.createReferenceTo(c_orig, [1..1])
   foreach p ∈ getDynamicPropertiesOf(c_exe) do
      c_value ← createClass()
      mm_trace ← mmtrace ∪ {c_value}
     c_value.properties ← {copyProperty(p)}
     c_traced.createReferenceTo(c_value, [0..*], ordered)
     c_value.createReferenceTo(c_traced, [1..1])
     c_exeState.createReferenceTo(c_value, [0..*], unordered)
     c_value.createReferenceTo(c_exeState, [1..1])
  foreach r ∈ exeTransf do
    map_steps ← createMap()
    createStepClass(r, map_steps,mm_trace, c_smallStep, c_bigStep)
    replaceReferencesToExecutionMM(mm_trace,mm_as,mm_exe)
end
```

Рис. 3. Фрагмент алгоритма построения трассы выполнения модели комплекса требований в среде VM FUML

В случае обнаружения каких-либо дефектов в комплекс требований вносятся необходимые исправления, после чего он подвергается повторной процедуре валидации и верификации. После прохождения описанной в данном разделе процедуры тестирования комплекс функциональных и эксплуатационных требований будет обладать следующими необходимыми для его использования на этапе проектирования АСУ КОТИ свойствами: полнотой, корректностью, непротиворечивостью, системностью, однозначностью, согласованностью. Такие свойства комплекса требований, как прослеживаемость, верифицируемость и модифицируемость обеспечиваются средствами единой модельно-языковой и информационно-программной среды, в рамках которой осуществляется разработка всех артефактов жизненного цикла АСУ КОТИ.

Алгоритмы и программные средства разработки и верификации архитектуры и проектных решений

Проектирование архитектуры и проектных решений АСУ КОТИ осуществляется системным архитектором на основе разработанного и верифицированного на предыдущем этапе комплекса требований в той же модельно-языковой и информационно-программной среде. Для этого ему предоставляются графические и табличные шаблоны проектирования структурных и поведенческих аспектов системы:

- для описания состава и структуры аппаратных и программных компонентов системы – диаграммы внешнего (bdd, block definition diagram) и внутреннего (ibd, internal block diagram) блоков;
- для представления реализующих определенные функции компонентов системы – диаграммы активных классов (active_class);
- для описания используемых или создаваемых в процессе функционирования системы объектов и артефактов – диаграммы пассивных классов (passive_class);
- для представления алгоритмов реализации методов классов (функций) – диаграммы деятельности (*Activity*), состояний (*State Machine)* и взаимодействия (*Sequience*).

Для представления отношений между ними и описания различного рода ограничений и условий используются средства языка OCL и библиотека Eclipse OCL Standard Library [16].

При разработке шаблона для проектирования архитектуры и проектных решений АСУ КОТИ с помощью диаграмм активности были использованы раскрашенные временные сети Петри (РВСП), формальное описание которых представляется следующим выражением:

$$N = (P, T, C, F, m_I, S, Z),$$

где P, T, F и m_I имеют то же значение, что и для простых сетей Петри;

$C = \{c_1, c_2, \ldots, c_d\}$ – цвета или типы маркеров;

$S = (s_1, s_2, \ldots, s_n)$ – вектор временных задержек маркеров в позициях;

$Z = \{z_1, z_2, \ldots, z_r\}$ – вектор времени срабатывания разрешенных переходов.

При разработке шаблона для проектирования архитектуры и проектных решений АСУ КОТИ с помощью диаграмм состояний были использованы математические модели временных автоматов (ВА) [24, 25], описываемые следующим выражением:

$$TA \equiv < S, C, D, R, f, s_0 >,$$

где S – конечное множество позиций автомата;

C – конечное множество локальных часов, значения которых возрастают синхронно с реальным временем и могут принимать значения из множества действительных чисел $\Re$;

$f: S \to P$ – функция, которая ставит в соответствие каждой позиции $s \in S$ некоторый предикат $p \in P$; в общем случае p есть логическое выражение из ограничений вида $c \le k$, $c < k$, $c_i - c_j \le k$, $c_i - c_j < k$, где $c_i, c_j \in C$, $k \in \Re$. Переход из позиции $s \in S$ должен быть выполнен до того, как предикат $p = f(s)$ станет ложным, в противном случае автомат прекращает работу;

D – конечное множество действий;

$R \subseteq S \times P \times D \times C \times S$ – множество переходов автомата;

$s_0 \in S$ – начальная позиция автомата.

Для моделирования АСУ КОТИ, которые состоят из нескольких взаимодействующих компонент, целесообразно использовать сети временных автоматов (СВА).

Формальная модель архитектуры и проектных решений должна обладать следующими характеристиками качества:

- полнотой и корректностью реализации функциональных требований;
- полнотой и корректностью реализации эксплуатационных требований;
- согласованностью и непротиворечивостью всех диаграмм модели;
- отсутствием избыточности диаграмм и их атрибутов;
- отсутствием взаимоблокировок, невыполнимых операций и зацикливаний, которые могут привести к нарушению безопасности и живости системы.

Состав и структура комплекса программных средств, обеспечивающих проверку этих характеристик, представлены на рисунке 4. Основными из них являются: виртуальная машина VM FUML, CPN Tools, SPIN, Rodin, Modelica. Проверка полноты и корректности реализации функциональных требований осуществляется посредством анализа трасс выполнения функций в среде VM FUML, включая полноту учета и правильность обработки критичных событий. Данный вид проверки выполняется при непосредственном участии экспертов и системных проектировщиков. Кроме того, модели, представленные в виде диаграмм активностей (*d_act*), транслируются на язык CPN ML и затем верифицируются средствами CPN Tools. Диаграммы последовательностей (*d_seq*) транслируются на язык Promela для последующей верификации в среде SPIN. Диаграммы состояний (*d_sm*) транслируются на язык EventB и проверяются с помощью средств Rodin.

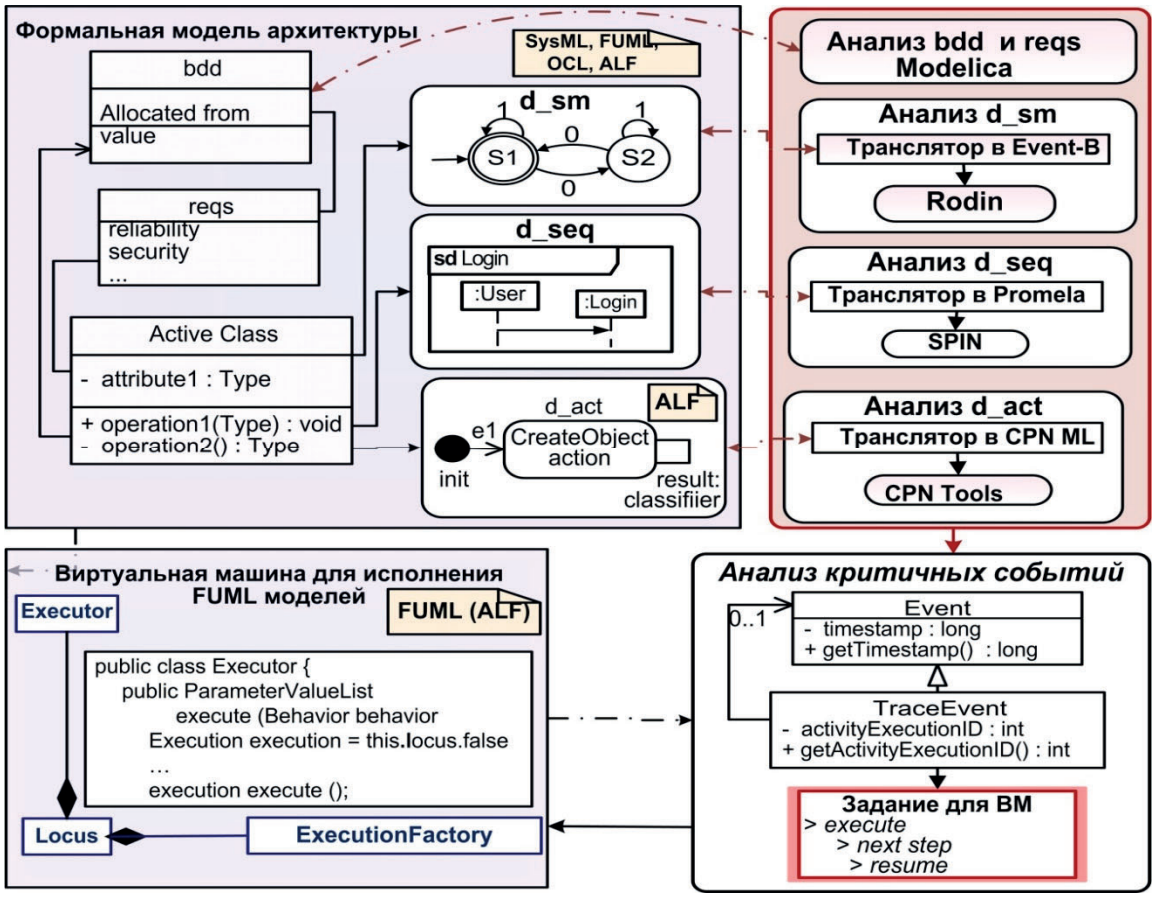

Рис. 4. Состав и структура комплекса программных средств, обеспечивающих валидацию и верификацию проектных решений

Процедура проверки непротиворечивости, согласованности и неизбыточности входящих в проект АСУ КОТИ диаграмм заключается в проверке следующих условий:

- каждый класс представлен не менее чем в одной поведенческой диаграмме;
- для каждого метода класса в проекте имеется единственная диаграмма поведения (активности, состояний, последовательности);
- сигнатуры методов класса и описывающих их реализацию диаграмм поведения совпадают;
- каждая поведенческая диаграмма имеет связь (отношение *realization*) с методом класса, который она реализует;
- все действующие лица (*actor*) и информационные объекты из поведенческих диаграмм должны быть представлены в виде классов (*class*);
- типы данных элементов диаграмм, связанных отношениями агрегации и композиции, должны совпадать.

Для описания и верификации приведенных выше условий используется язык OCL и библиотека Eclipse OCL Standard Library [26].

Для верификации технической части проектных решений предлагается использовать средство моделирования киберфизических систем OpenModelica [27]. Проектные решения, представленные в виде SysML-диаграмм блоков и классов, с помощью специальных программ транслируются в соответствующие конструкции объектно-ориентированного языка Modelica.

В случае обнаружения каких-либо дефектов проектные решения возвращаются архитекторам системы на доработку. Исправленный проект подвергается повторной процедуре валидации и верификации. В результате реализации данного последовательно-итерационного и программно-управляемого процесса будут разработаны проектные решения, в полной мере соответствующие предъявленным к АСУ КОТИ требованиям пользователей, условиям применения и нормативным документам.

Заключение

Представленный в статье комплекс программных средств предназначен для реализации программно-управляемого процесса разработки АСУ КОТИ в единой для всех его участников модельно-языковой и информационно-программной среде, построенной на основе технологий объектно-ориентированного анализа и проектирования, языков и средств визуального моделирования SysML, FUML, OCL. Его применение позволит существенно сократить логические и семантические разрывы между представлениями о разрабатываемой системе у участников процесса создания АСУ КОТИ, повысит эффективность их

взаимодействия и приведет к повышению качества всех артефактов жизненного цикла разработки систем.

Реализация автоматизированных процедур верификации, валидации и коррекции комплекса требований и проектных решений с помощью таких средств, как VM FUML, CPN Tools, SPIN и Rodin, позволит улучшить экономические показатели в части снижения финансовых и временных затрат, связанных с выполнением дополнительных работ как в случае обнаружения каких-либо дефектов, так и при изменении самих требований или условий эксплуатации АСУ КОТИ.

Литература

1. IBM Engineering Systems Design Rhapsody – Developer. – URL: http://www.ibm.com/ru-ru/marketplace/uml-tools (дата обращения 20.06.2019).

2. Sparx Systems Enterprise Architect. – URL: http://sparxsystems.com/products/ea (дата обращения 21.05.2019).

3. Буздалов Д.В. Инструментальные средства проектирования систем интегрированной модульной авионики / Д.В. Буздалов, С.В. Зеленов, Е.В. Корныхин, А.К. Петренко, А.В. Страх, А.А. Угненко, А.В. Хорошилов // Труды ИСП РАН. – 2014. Т. 26. – Вып. 1. – С. 201–230.

4. Systems Engineering and Software Engineering // SEBoK. – URL: http://www.sebokwiki.org/wiki/Systems_Engineering_and_Software_Engineering (дата обращения 25.05.2019).

5. Hart L.E. Introduction to Model-Based System Engineering (MBSE) and SysML // International Council on Systems Engineering Website. – URL: http://www.incose.org/docs/default-source/delaware-valley/mbse-overview-incose-30-july-2015.pdf (дата обращения 21.05.2019).

6. The OMG Specifications Catalog // Object Management Group. – URL: http://www.omg.org/spec (дата обращения 20.06.2019).

7. Modelio Open Source – UML and BPMN free modeling tool. – URL: http://www.modelio.org (дата обращения 20.06.2019).

8. Eclipse Papyrus. Modeling environment. – URL: http://www.eclipse.org/papyrus (дата обращения 20.06.2019).

9. Eclipse GEMOC Studio. – URL: http://projects.eclipse.org/projects/modeling.gemoc (дата обращения: 20.06.2019).

10. Milicev D. Model-Driven Development with Executable UML, Indianapolis, Wiley Publishing, Inc., Wrox, 2009. – 816 p.

11. Friedenthal S., Moore A., Steiner R. A Practical Guide to SysML: The Systems Modeling Language. 2nd edition, Morgan Kaufmann, 2011. – 640 p.

12. Delligatti L. SysML Distilled: A Brief Guide to the Systems Modeling Language, Addison-Wesley Professional, 2013. – 304 p.

13. Hatebur D., Heisel M. (2010) A UML Profile for Requirements Analysis of Dependable Software. *In: Schoitsch E. (eds) Proceedings of the 29th International Conference on Computer Safety, Reliability, and Security (SAFECOMP 2010), Vienna, Austria, 14–17 September 2010. Lecture Notes in Computer Science, Vol. 6351*, Berlin, Springer-Verlag, 2010. – Pp. 317–331. – URL: http://www.researchgate.net/publication/221147725_A_UML_Profile_for_Requirements_Analysis_of_Dependable_Software (дата обращения 20.06.2019).

14. Bousse E., Mayerhofer T., Combemale B., Baudry B. Advanced and Efficient Execution Trace Management for Executable Domain-Specific Modeling Languages, *Software and Systems Modeling*, 2019, Vol. 18, No. 1. – Pp. 385–421. – URL:http://link.springer.com/article/10.1007/s10270-017-0598-5 (дата обращения 27.04.2019).

15. Марков А.В. Автоматизация проектирования и анализа программного обеспечения с использованием языка UML и сетей Петри : дисс. … канд. техн. наук: 05.13.11. – Новосибирск, 2015. – 176 с.

16. Anand S. et al. An Orchestrated Survey on Automated Software Test Case Generation, *Journal of Systems and Software*, 2013, Vol. 86, Is. 8. – Pp. 1978–2001.

17. Andre E., Choppy C., and Reggio G. Activity Diagrams Patterns for Modeling Business Processes. – URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.650.6505&rank=1 (дата обращения 19.06.2019).

18. Rahim M., Boukala-ioualalen M., Hammad A. Petri Nets Based Approach for Modular Verification of SysML Requirements on Activity Diagrams. – URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.664.7517&rank=1 (дата обращения 27.05.2019).

19. Li S., Balaguer S., David A. et al. Scenario-Based Verification of Real-Time Systems Using Uppaal, Formal Methods in System Design, 2010, Vol. 37, Is. 2–3. – Pp. 200–264.

20. CPN Tools – A tool for editing, simulating, and analyzing Colored Petri nets. – URL: http://cpntools.org (дата обращения 27.05.2019).

21. Home of Event-B and the Rodin Platform. – URL: http://www.event-b.org (дата обращения 11.05.2019).

22. SPIN – Formal Verification. – URL: http://spinroot.com/spin/whatispin.html (дата обращения 11.06.2019)

23. Введение в язык PROMELA и систему комплексной верификации SPIN : учебное пособие / И.В. Шошмина, Ю.Г. Карпов. – СПб. : Изд-во Политехнического ун-та, 2010. – 110 с.

24. Bengtsson J., Yi W. Timed Automata: Semantics, Algorithms and Tools. Uppsala University. *In: Desel J., Reisig W., Rozenberg G. (eds) Lectures on Concurrency and Petri Nets. ACPN 2003. Lecture Notes in Computer Science, Vol. 3098*, Berlin, Springer-Verlag, 2004. – pp. 87–124. – URL: http://www.fi.muni.cz/~xpelanek/IA158/TA-intro.pdf. (дата обращения 27.05.2019).

25. Твардовский А.С., Лапутенко А.В. О возможностях автоматного описания параллельной композиции временных автоматов // Труды ИСП РАН. – 2018. – Том 30, Вып. 1. – С. 25–40.

26. Eclipse OCL (Object Constraint Language). – URL: http://projects.eclipse.org/projects/modeling.mdt.ocl (дата обращения 27.05.2019).

27. OpenModelica. – URL: http://www.openmodelica.org (дата обращения 27.05.2019).

Complex of Software for Development and Verification of Requirements and Design Solutions Automated Control System by Transport Infrastructure Objects

A.V. Baev, PhD A.V. Samonov
Military Space Academy named after A.F. Mozhaisky
Saint Petersburg, Russia
a.baih@mail.ru, a.samonov@mail.ru

Abstract. **The composition, structure and methods of application of a complex of software that provides developers of automated control systems of organizational and technical complexes and objects of transport infrastructure with intelligent means of support in justifying the requirements and designing systems are presented. The complex of means is developed for the purpose of elimination of logical and semantic gaps between representations about the developed system at participants of this process: the customer and users, system architects and designers, engineers and programmers. This is achieved through the implementation of a program-driven process of development and verification of formal models of requirements and design solutions in a common model-language and information-software environment for all its participants. Construction of models is carried out by the automated method on the basis of templates of the formal description of requirements to functional and operational characteristics of the created systems developed with use of Petri nets, time automata, languages of modeling and design SysML, FUML, OCL.**

For validation and verification of complex requirements and design solutions developed meta-model and algorithm for design and analysis of execution routes of model that run in a virtual machine environment FUML VM. The methods of integration and use for automated testing of models of requirements and design solutions of specialized verification tools CPN Tools, Rodin, SPIN and Modelica presents. This complex provides more effective interaction between the customer and the contractor both in the development of requirements and in the design of the system, early detection and elimination of defects through the implementation of automated verification, validation and correction procedures. This will improve the quality of requirements and design solutions, as well as improve economic performance in terms of reducing the financial and time costs associated with the implementation of additional work in the case of defects, and when changing requirements or operating conditions.

Keywords: **automated control systems, validation and verification, time machines, design and modeling, Petri nets, functional and operational requirements.**

References

1. IBM Engineering Systems Design Rhapsody – Developer. Available at: http://www.ibm.com/ru-ru/marketplace/uml-tools (accessed 20.06.2019).

2. Sparx Systems Enterprise Architect. Available at: http://sparxsystems.com/products/ea (accessed 21.05.2019).

3. Buzdalov D.V. et al. Tools for System Design of Integrated Modular Avionics [Instrumental'nye sredstva proektirovaniya sistem integrirovannoy modul'noy avioniki], *Proc. of the Institute for System Programming of the RAS [Trudy ISP RAN]*, 2014, Vol. 26, Is. 1. – Pp. 201–230.

4. SEBoK: Systems Engineering and Software Engineering. Available at: http://www.sebokwiki.org/wiki/Systems_Engineering_and_Software_Engineering (accessed 25.05.2019).

5. Laura E. Hart. Introduction To Model-Based System Engineering (MBSE) and SysML. Available at: http://www.incose.org/docs/default-source/delaware-valley/mbse-overview-incose-30-july-2015.pdf (accessed 21.05.2019).

6. Object Management Group: The OMG Specifications Catalog. Available at: http://www.omg.org/spec (accessed 20.06 2019).

7. Modelio Open Source – UML and BPMN free modeling tool. Available at: http://www.modelio.org (accessed 20.06 2019).

8. Eclipse Papyrus. Modeling environment. Available at: http://www.eclipse.org/papyrus (accessed 20.06 2019).

9. Eclipse GEMOC Studio. Available at: http://projects.eclipse.org/projects/modeling.gemoc (accessed 20.06 2019).

10. Milicev D. Model-Driven Development with Executable UML, Indianapolis, Wiley Publishing, Inc., Wrox, 2009. – 816 p.

11. Friedenthal S., Moore A., Steiner R. A Practical Guide to SysML: The Systems Modeling Language. 2nd edition, Morgan Kaufmann, 2011. – 640 p.

12. Delligatti L. SysML Distilled: A Brief Guide to the Systems Modeling Language, Addison-Wesley Professional, 2013. – 304 p.

13. Hatebur D., Heisel M. (2010) A UML Profile for Requirements Analysis of Dependable Software. *In: Schoitsch E. (eds) Proceedings of the 29th International Conference on Computer Safety, Reliability, and Security (SAFECOMP 2010), Vienna, Austria, 14–17 September 2010. Lecture Notes in Computer Science, Vol. 6351*, Berlin, Springer-Verlag, 2010. – Pp. 317–331. Available at: http://www.researchgate.net/publication/221147725_A_UML_Profile_for_Requirements_Analysis_of_Dependable_Software (accessed 20.06 2019).

14. Bousse E., Mayerhofer T., Combemale B., Baudry B. Advanced and Efficient Execution Trace Management for Executable Domain-Specific Modeling Languages, *Software and Systems Modeling*, 2019, Vol. 18, No. 1. – Pp. 385–421. Available at: http://link.springer.com/article/10.1007/s10270-017-0598-5 (accessed 27.04.2019).

15. Markov A.V. Automation of Software Design and Analysis Using UML and Petri Nets [Avtomatizatsiya proyektirovaniya i analiza programmnogo obespecheniya s ispol'zovaniyem yazyka UML i setey Petri]: diss. on competition of a scientific degree Ph.D. (Engin.), Novosibirsk, 2015. – 176 p.

16. Anand S. et al. An Orchestrated Survey on Automated Software Test Case Generation, *Journal of Systems and Software*, 2013, Vol. 86, Is. 8. – Pp. 1978–2001.

17. Andre E., Choppy C., and Reggio G. Activity Diagrams Patterns for Modeling Business Processes. Available at:http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.650.6505&rank=1 (accessed 19.06.2019).

18. Rahim M., Boukala-ioualalen M., Hammad A. Petri Nets Based Approach for Modular Verification of SysML Requirements on Activity Diagrams. Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.664.7517&rank=1 (accessed 27.05.2019).

19. Li S., Balaguer S., David A. et al. Scenario-Based Verification of Real-Time Systems Using Uppaal, Formal Methods in System Design, 2010, Vol. 37, Is. 2–3. – Pp. 200–264.

20. CPN Tools – A tool for editing, simulating, and analyzing Colored Petri nets. Available at: http://cpntools.org (accessed 27.05.2019).

21. Home of Event-B and the Rodin Platform. Available at: http://www.event-b.org (accessed 11.05.2019).

22. SPIN – Formal Verification. Available at: http://spinroot.com/spin/whatispin.html (accessed 11.06.2019)

23. Shoshmina I.V., Karpov Yu.G. Introduction to PROMELA language and SPIN Integrated Verification System: Study guide [Vvedeniye v yazyk PROMELA i sistemu kompleksnoy verifikatsii SPIN: Uchebnoe posobie], Saint Petesburg, Peter the Great St. Petersburg Polytechnic University, 2010. – 110 p.

24. Bengtsson J., Yi W. Timed Automata: Semantics, Algorithms and Tools. Uppsala University. *In: Desel J., Reisig W., Rozenberg G. (eds) Lectures on Concurrency and Petri Nets. ACPN 2003. Lecture Notes in Computer Science, Vol. 3098*, Berlin, Springer-Verlag, 2004, pp. 87–124. Available at: http://www.fi.muni.cz/~xpelanek/IA158/TA-intro.pdf. (accessed 27.05.2019).

25. Tvardovskii A.S., Laputenko A.V. On the possibilities of FSM description of parallel composition of timed Finite State Machines [O vozmozhnostyakh avtomatnogo opisaniya parallel'noy kompozitsii vremennykh avtomatov], *Proc. of the Institute for System Programming of the RAS [Trudy ISP RAN]*, 2018, Vol. 30, Is. 1. – Pp. 25–40.

26. Eclipse OCL (Object Constraint Language). Available at: http://projects.eclipse.org/projects/modeling.mdt.ocl. (accessed: 27.05.2019).

27. OpenModelica. Available at: http://www.openmodelica.org (accessed: 27.05.2019).

Программный продукт автоматизации технологических процессов моделирования конструкций вагонов на предприятиях АО «Узбекистон темир йуллари»

Ш.Х. Султонов
Петербургский государственный университет
путей сообщения Императора Александра I
Санкт-Петербург, Россия
Sultonovsh@yandex.ru

З.Р. Султонова
Навоийский государственный горный институт
Навои, Узбекистан
Sultonovazr@yandex.ru

Аннотация. **Целью работы является автоматизация вычислений нагрузок, действующих на конструкцию кузова грузового вагона при прочностных расчетах, с дальнейшим трехмерным моделированием с применением информационных технологий. Разработанная на языке C# с применением интегрированной среды Visual Studio программа позволяет повысить точность и сократить сроки вычисления нагрузок, действующих на кузов грузового вагона, при прочностных расчетах с различными исходными технико-экономическими характеристиками и конструкционными особенностями грузовых вагонов.**

Ключевые слова: **вагоностроение, грузовой вагон, металлоконструкция, нагрузка, прочность, автоматизация, программирование.**

Введение

С приобретением государственной независимости в Республике Узбекистан большое внимание уделяется развитию железнодорожного транспорта [1]. На сегодняшний день этот вид транспорта является неотъемлемой частью и играет ключевую роль в формировании экономики страны [2].

Для освоения возрастающих объемов перевозок по сети железных дорог Узбекистана и пополнения парка подвижного состава современными вагонами было принято решение освоить производство новых грузовых и пассажирских вагонов на базе существующих вагоноремонтных предприятий страны [3, 4].

В связи с этим на базе ОАО «Ташкентский завод по ремонту пассажирских вагонов» было освоено серийное производство современных пассажирских вагонов купейного типа с установкой кондиционирования воздуха модели 68-907 и тележки пассажирского вагона модели 68-908 (68-909) [5]. На базе ДП «Андижанский механический завод» и ДП «Литейно-механический завод» было освоено производство новых моделей грузовых вагонов различного типа [6].

В процессе проектирования и разработки технической документации нового и модернизированного подвижного состава необходимо проверить соответствие прочностных характеристик их конструкций установленным требованиям [7].

Основными документами при этом являются «Нормы для расчета и проектирования новых и модернизируемых вагонов» [8] и ГОСТ 33211 [9], которые содержат критерии оценки прочности конструкции кузова грузового вагона.

Для повышения качества проектирования, сокращения сроков выполнения теоретических и экспериментальных исследований и освоения новых видов выпускаемой продукции явным преимуществом является автоматизация процесса моделирования новых конструкций грузовых вагонов и внедрение цифрового прототипа с применением информационных технологий на вагоностроительных предприятиях [10, 11].

Требования к программе

С целью ускорить и упростить проведение инженерного анализа проектируемого вагона с дальнейшим трехмерным моделированием и расчетом на прочность требуется разработка программы для вычисления величин ожидаемых нагрузок, действующих на вагон. Была разработана специальная программа, которая удовлетворяет и соответствует перечисленным ниже требованиям [12, 13]:

- отображение полей для ввода исходных данных;
- функция проверки корректности введенных значений;
- функция для выполнения расчета и получения результатов;
- блок для результирующих параметров;
- функция очистки полей;
- блок сочетания нагрузок, действующих на кузов вагона;
- справочный блок;
- программа не должна требовать установки дополнительных библиотек и плагинов;
- программа должна обладать интуитивно понятным графическим интерфейсом и возможностью использования на любых компьютерах с предустановленной операционной системой Windows.

При разработке программы для определения нагрузок, действующих на кузов грузовых вагонов, на начальном этапе была построена функциональная схема программы, которая создана по модульному принципу и позволяет в будущем расширить его функциональность за счет присоединения дополнительных функций для определения других технико-экономических параметров вагонов. Подробная функциональная схема программы для определения нагрузок, действующих на кузов грузовых вагонов, представлена на рисунке 1.

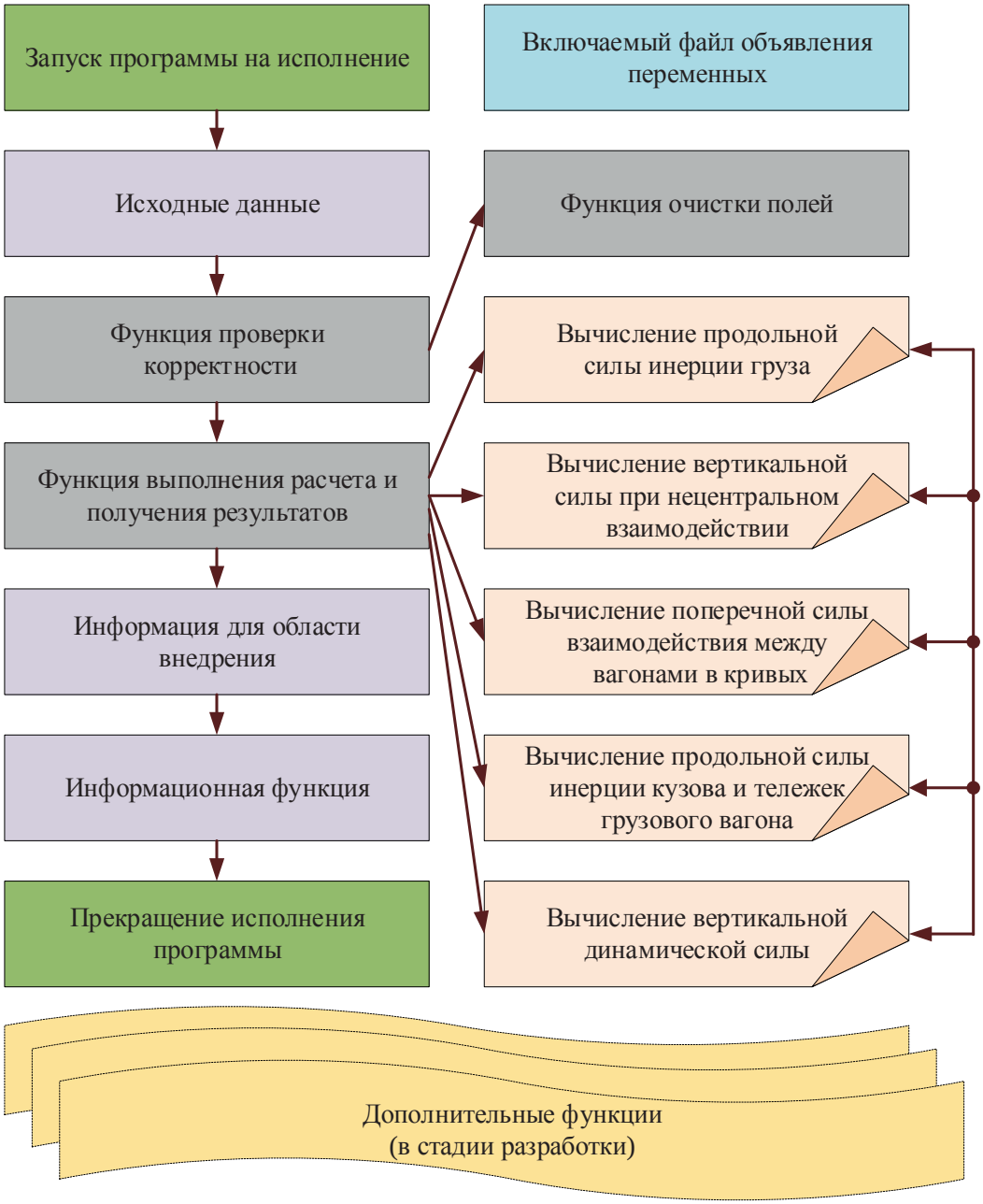

Рис. 1. Функциональная схема программы

Программа состоит из функций, каждая функция соответствует определенной форме и содержит процедуры. Функция начинает работать сразу же по окончании работы предыдущей функции либо по вызову пользователем щелчком мыши по соответствующей вкладке.

Алгоритм программы можно разделить на четыре основных блока: определение исходных данных, проверка правильности исходных данных, выполнение задачи и получение результатов (рис. 2).

В блоке *1* осуществляется выбор типа вагона из представленного списка; в блоке *2* – ввод исходных данных; в блоке *3* осуществляется проверка введенных исходных данных. При выявлении некорректных данных необходимо проверить введенные данные заново и ввести их корректные значения в блок *2*. При необходимости можно обратиться в справочный блок (блок *4*). В случае принятия решения о корректности значения процесс передается в блок *5*. В блоке *6* отображаются результаты расчетных значений.

В результате анализа инструментальных средств и языка программирования, предназначенных для реализации программного обеспечения для вычисления величин нагрузок, действующих на кузов грузового вагона, было принято решение использовать язык программирования C# [14] с помощью интегрированной среды разработки Visual Studio [15].

Для запуска разработанной программы [16] необходимо запустить файл *FactCar.exe*. После запуска программы внешний вид процесса загрузки будет выглядеть, как показано на рисунке 3.

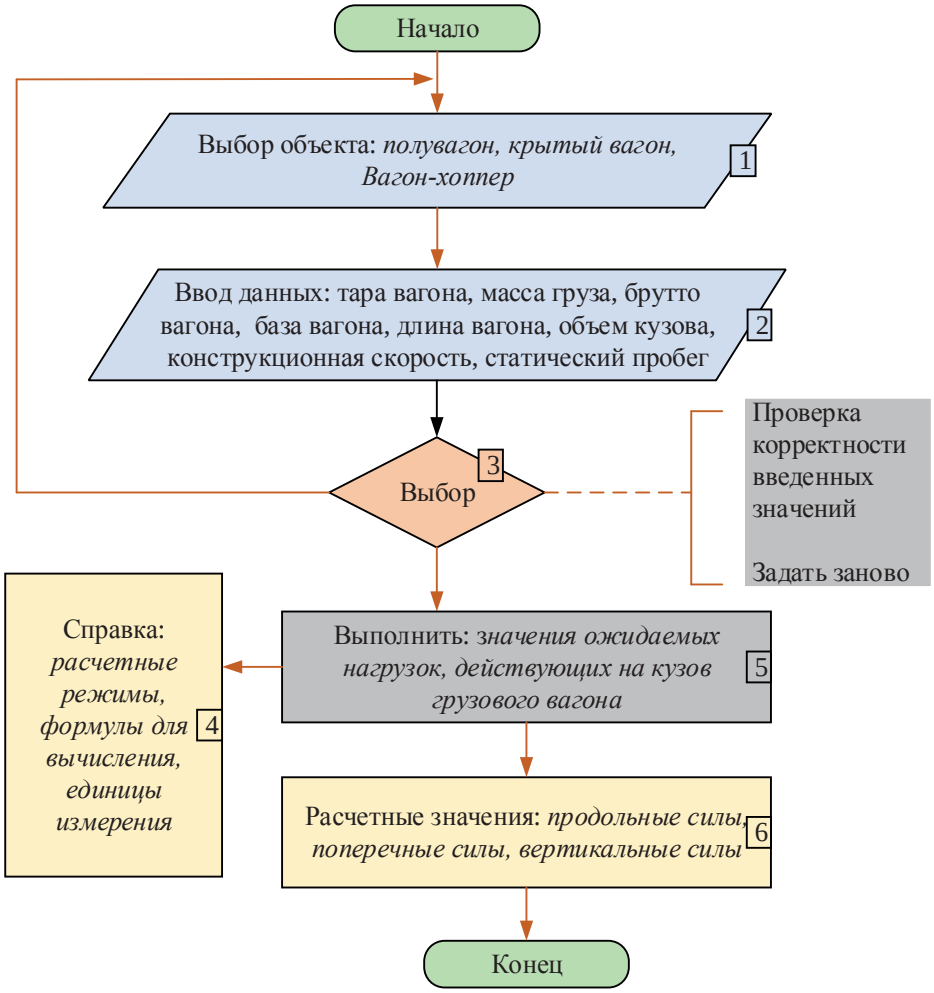

Рис. 2. Блок-схема алгоритма программы

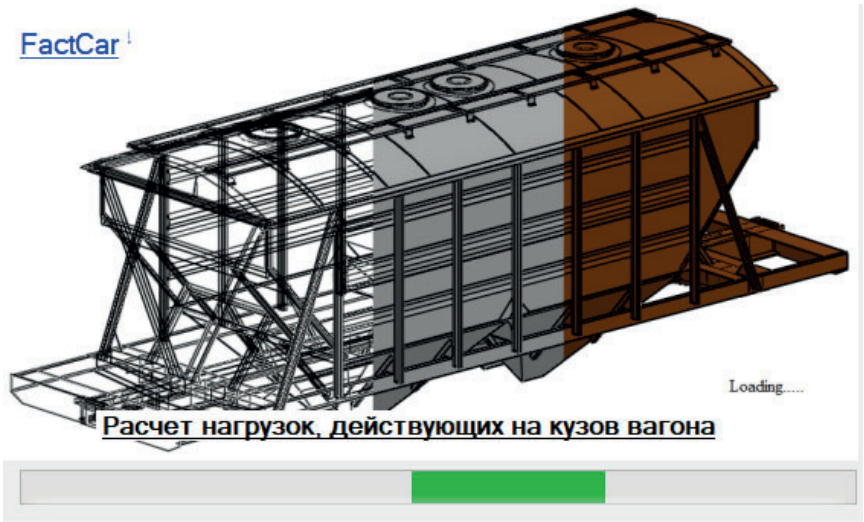

Рис. 3. Внешний вид процесса загрузки программы

После нажатия кнопкой мыши по этому интерфейсу на экране компьютера появится главное окно программного продукта. В главном окне располагаются несколько вкладок, на которых имеются все необходимые ячейки (поля) для введения исходных данных. Основное окно программы представлено на рисунке 4.

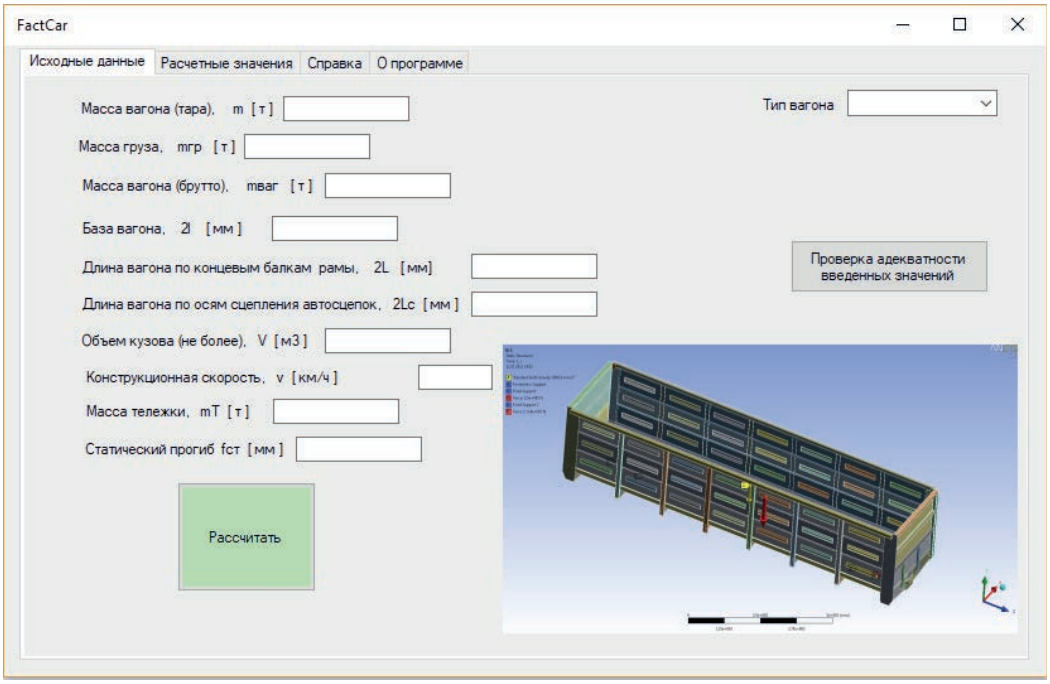

Рис. 4. Основное окно программы (до ввода исходных данных)

В основном окне программы на вкладке «Исходные данные» отображены:

– поля для заполнения значений с описаниями;
– кнопка для выбора типа вагона;
– кнопка для проверки на адекватность введенных исходных данных;
– поле для сведений о вагоне;
– кнопка для выполнения задачи.

При наведении указателя мыши на элемент «Тип вагона» пользователь имеет возможность в меню выбрать тип исследуемого вагона: полувагон, крытый вагон или вагон-хоппер.

Перед выполнением расчета целесообразно проверить корректность введенных значений, нажав на кнопку «Проверка адекватности введенных значений». Если хотя бы в одно из полей введены нечисловые данные (или не введены данные вовсе) программа выдаст пользователю соответствующее сообщение: «Пустые поля или некорректные значения!!!» (рис. 5).

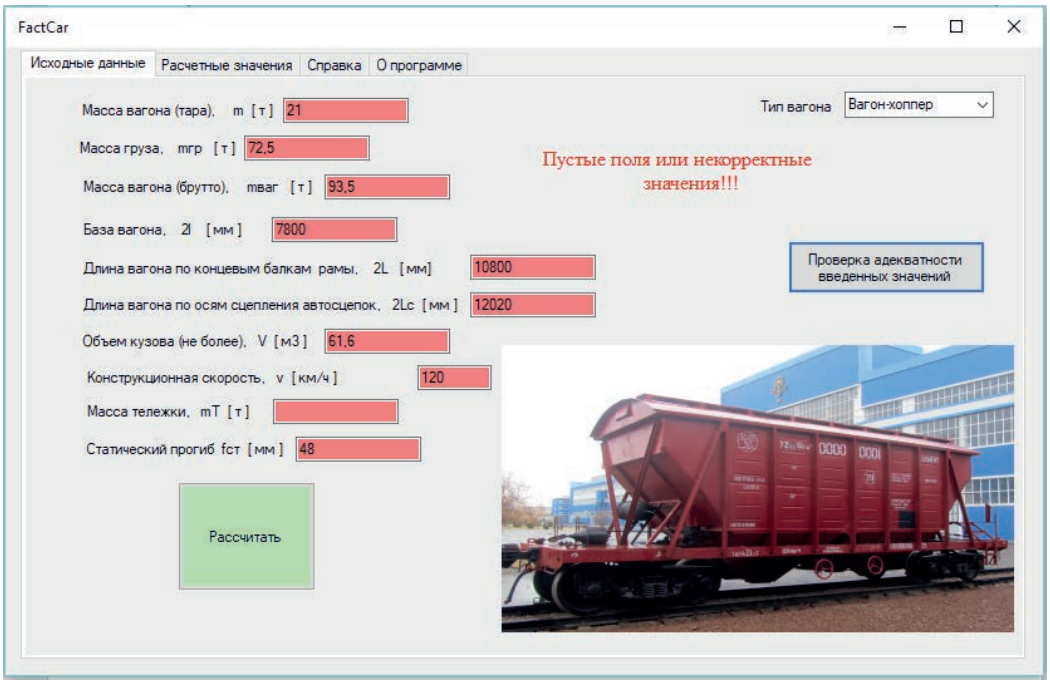

Рис. 5. Проверка корректности введенных данных

Для редактирования несоответствующих исходных данных следует выбрать и выделить область, в которой находятся редактируемые данные. После выделения области следует вписать корректные значения исходных данных.

Если во все поля введены правильные значения исходных данных, после нажатия на кнопку «Рассчитать» выполняется расчет и появляется сообщение: «Успешно выполнен!» (рис. 6). Расчетные значения становятся доступны на соответствующей вкладке.

Во вкладке «Расчетные значения» отображаются получаемые результаты действующих на кузов грузового вагона нагрузок (рис. 7). Результаты выполненных расчетов при визуализации округляются до двух десятичных знаков после запятой.

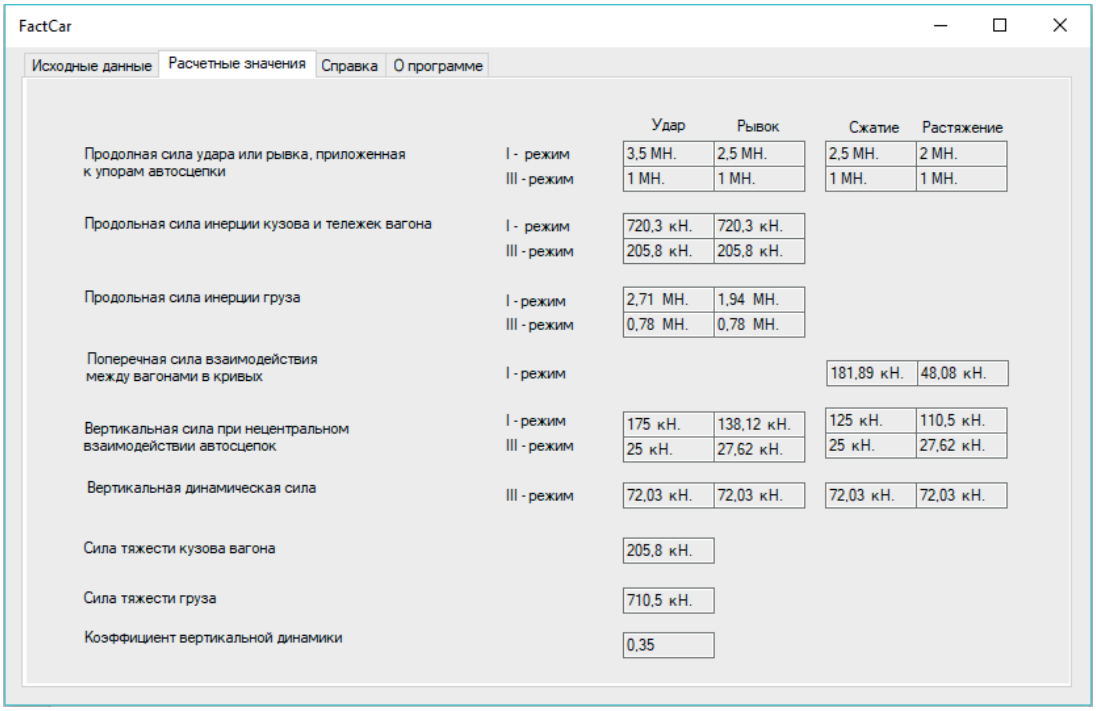

Рис. 6. Процесс выполнения расчета

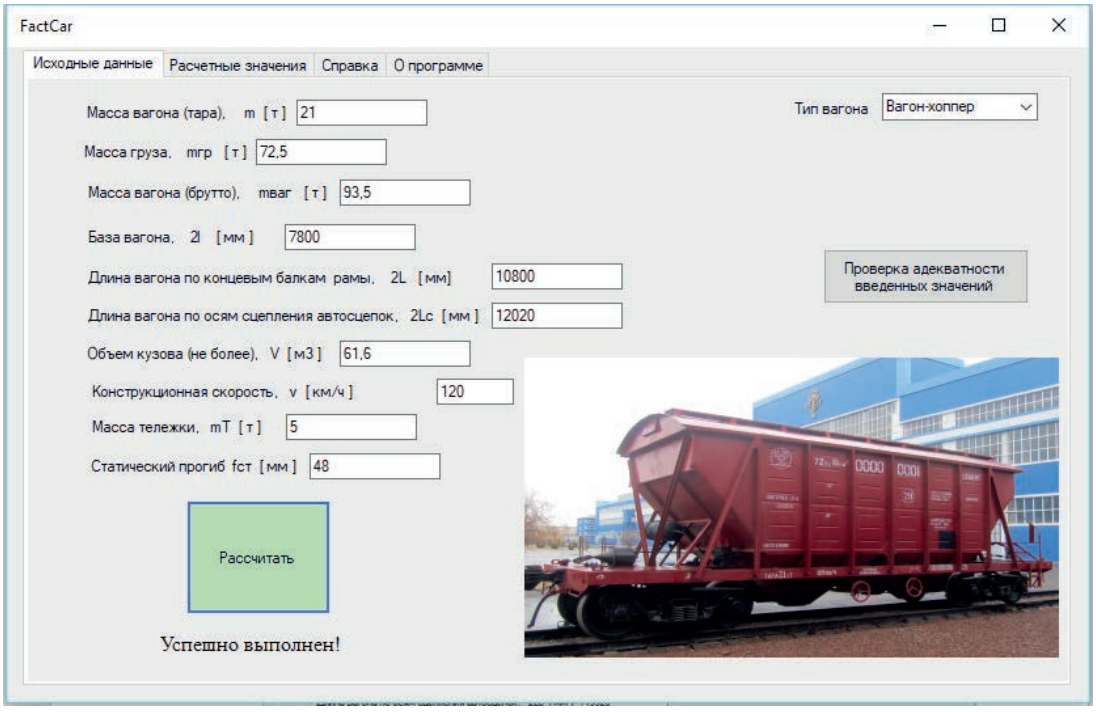

Рис. 7. Окно отображения результатов расчета

Вкладка «Справка» в главной форме программы предназначена для оказания конкретной помощи пользователю. Здесь приведены все необходимые формулы и их значения для выполнения расчета.

Во вкладке «О программе» приведено описание назначения программы, возможности данной программы, а также ее основные характеристики и ограничения, накладываемые на область применения программы в соответствии с нормативными документами.

Корректный выход из программы возможен стандартным способом: при нажатии на «крестик» в правом верхнем углу рабочей формы.

Заключение

Разработанная на языке C# с применением интегрированной среды Visual Studio программа обладает удобным пользовательским интерфейсом и позволяет в кратчайшие сроки вычислять нагрузки, действующие на кузов грузового вагона, при прочностных расчетах с дальнейшим трехмерным моделированием с различными исходными технико-экономическими характеристиками и конструкционными особенностями грузовых вагонов.

Литература

1. О мерах по реализации инвестиционного проекта «Развитие ремонтной базы подвижного состава, организация вагоностроения и реконструкция литейного производства на ДП «Quyuv mexanika zavodi» : постановление Президента Республики Узбекистан от 28.06.2012 № ПП-1780.

2. О стратегии действий по дальнейшему развитию Республики Узбекистан : указ Президента Республики Узбекистан от 07.02.2017 № УП-4947.

3. Рахимов Р.В. Анализ состояния и перспективы развития парка грузовых вагонов АО «Узбекистон темир йуллари» // Научные труды республиканской научно-технической Конференции с участием зарубежных ученых «Ресурсосберегающие технологии на железнодорожном транспорте» (20–21 декабря 2016, Ташкент). – Ташкент : ТашИИТ, 2017. – С. 91–94.

4. Rahimov R.V., Ruzmetov Ya.O. Analysis of the State and Prospects of the Development of the Freight Wagon Fleet of the Republic of Uzbekistan // Non-Ferrous Metals. – 2018. – No. 1 (44). – Pp. 7–11.

5. Rahimov R.V. Researches of the stressed – deformed state of the open wagon body model 12-9922, produced in Uzbekistan, *Proceeding VI International Scientific Conference «Transport Problems 2014» (25-27 June 2014, Katowice)*, Katowice, Silesian University of Technology, 2014. – Pp. 614–621.

6. Rahimov R.V., Khatamov S.A., Rakhmatov Z.X. Scientific Substantiation of Technical Solutions for the Improvement of the Construction of the Body of a Hopper Car for Transportation of Cement Produced in the Republic of Uzbekistan, *European Science Review*, 2017, No. 3–4. – Pp. 13–17.

7. Rahimov R.V., Ruzmetov Ya. O. Development of Maintenance and Repairing Tank Wagon for the Transport of Concentrated Sulfuric Acid // Вестник ТашГТУ. – 2017. – № 2. – С. 86–91.

8. Нормы для расчета и проектирования новых и модернизируемых вагонов железных дорог МПС колеи 1520 мм (несамоходных). – М. : ГосНИИВ ; ВНИИЖТ, 1996. – 317 с.

9. ГОСТ 33211-2014. Вагоны грузовые. Требования к прочности и динамическим качествам (с поправкой). – М. : Стандартинформ, 2016. – 57 с.

10. Бубнов В.П. Алгоритм автоматизации проектно-конструкторских работ и его применение в вагоностроении / В.П. Бубнов, Ш.Х. Султонов // Сб. трудов IV международной научно-методической конференции «Проблемы математической и естественно-научной подготовки в инженерном образовании» (3 ноября 2016 г., Санкт-Петербург). – Т. 2. – СПб. : ПГУПС, 2017. – С. 16–21.

11. Бубнов В.П. Применение систем автоматизированного проектирования в машиностроении / В.П. Бубнов, Ш.Х. Султонов // Интеллектуальные технологии на транспорте. – 2017. – № 1 (9). – С. 48–51.

12. ISO/IEC/IEEE 29148:2018. Systems and software engineering – Life cycle processes – Requirements engineering.

13. ГОСТ Р 57193-2016. Системная и программная инженерия. Процессы жизненного цикла систем. – М. : Стандартинформ, 2016. – 97 с.

14. Троелсен Э. C# и платформа .NET / пер. с англ. Р. Михеева. – СПб. : Питер, 2004. – 796 с. – (Библиотека программиста).

15. Freeman A., Sanderson S. Pro ASP.NET MVC 4. 4th edition, Apress, 2012. – 756 p.

16. Программное обеспечение «ВагонСила» : свидетельство об официальной регистрации программы для ЭВМ № DGU 05998 от 24.01.2019 Республика Узбекистан / Р.В. Рахимов, Ш.Х. Султонов ; заявл. DGU 2018 0873 от 07.11.2018.

Software Product of Automation of Technological Processes of Modeling Constructions of Wagons at the Enterprises of JSC «Uzbekiston Temir Yullari»

Sh.Kh. Sultonov
Emperor Alexander I Petersburg State Transport University
St. Petersburg, Russia
Sultonovsh@yandex.ru

Z.R. Sultonova
Navoi State Mining Institute
Navoiy, Uzbekistan
Sultonovazr@yandex.ru

Abstract. **The aim of the work is to automate the calculation of loads acting on the structure of the body of a freight wagon in strength calculations with further three-dimensional modeling with the use of information technology. The developed program in C# using the integrated environment of Visual Studio allows increasing the accuracy and reducing the time of calculation of loads acting on the body of a freight wagon in strength calculations with different initial technical and economic characteristics and structural features of freight wagons.**

Keywords: **wagon building, freight wagon, metalwork, load, strength, automation, programming.**

REFERENCES

1. About Measures for Implementation of Investment, Project «Development of Repair Base of Rolling Stock, Organization of Car-Building and Reconstruction of Foundry Production at DP «Quyuv Mexanika Zavodi»[O merakh po realizatsii investitsionnogo proekta «Razvitie remontnoy bazy podvizhnogo sostava, organizatsiya vagonostroyeniya i rekonstruktsiya liteynogo proizvodstva na DP «Quyuv mexanika zavodi»]: Resolution of the President of the Republic of Uzbekistan from 28.06.2012 No. PP-1780.

2. On the Strategy of Action for the Further Development of the Republic of Uzbekistan [O Strategii deystviy po dal'neyshemu razvitiyu Respubliki Uzbekistan]: Decree of the President of the Republic of Uzbekistan from 07.02.2017 No. UP-4947.

3. Rahimov R. V. Analysis of the status and development prospects of the fleet of freight cars of JSC «Uzbekiston Temir Yullari» [Analiz sostoyaniya i perspektivy razvitiya parka gruzovykh vagonov AO «Uzbekiston temir yullari»], *Proceedings of the Republican Scientific and Technical Conference with the Participation of Foreign Scientists «Resource-Saving Technologies in Railway Transport» [Nauchnye trudy respublikanskoy nauchno-tekhnicheskoy Konferentsii s uchastiyem zarubezhnykh uchenykh «Resurso-sberegayushchie tekhnologii na zheleznodorozhnom transporte»](20-21 December 2016, Tashkent)*, Tashkent, Tashkent Railway Engineering Institute, 2017. – Pp. 91–94.

4. Rahimov R.V., Ruzmetov Ya.O. Analysis of the State and Prospects of the Development of the Freight Wagon Fleet of the Republic of Uzbekistan, *Non-Ferrous Metals*, 2018, No. 1 (44). – Pp. 7–11.

5. Rahimov R.V. Researches of the stressed – deformed state of the open wagon body model 12-9922, produced in Uzbekistan, *Proceeding VI International Scientific Conference «Transport Problems 2014» (25–27 June 2014, Katowice)*, Katowice, Silesian University of Technology, 2014. – Pp. 614–621.

6. Rahimov R.V., Khatamov S.A., Rakhmatov Z.X. Scientific Substantiation of Technical Solutions for the Improvement of the Construction of the Body of a Hopper Car for Transportation of Cement Produced in the Republic of Uzbekistan, *European Science Review*, 2017, No. 3–4. – Pp. 13–17.

7. Rahimov R.V., Ruzmetov Ya. O. Development of Maintenance and Repairing Tank Wagon for the Transport of Concentrated Sulfuric Acid, *Bulletin of Tashkent State Technical University [Vestnik TashGTU]*, 2017, No. 2. – Pp. 86–91.

8. Standards for the Calculation and Design of New and Modernized Railroad Cars of the Ministry of Railways of 1520 mm Gauge (Non-Self-Propelled) [Normy dlya rascheta i proektirovaniya novykh i moderniziruemykh vagonov zheleznykh dorog MPS kolei 1520 mm (nesamokhodnykh)], Moscow, Research Institute of Carriage Engineering, Railway Research Institute, 1996. – 317 p.

9. GOST 33211-2014. Freight wagons. Requirements to structural strength and dynamic qualities (as amended) [GOST 33211-2014. Vagony gruzovye. Trebovaniya k prochnosti i dinamicheskim kachestvam (s popravkoy)], Moscow, StandardInform, 2016. – 57 p.

10. Bubnov V.P., Sultonov Sh.Kh. Algorithm of Automation of Design and Development Work and its Application in Railroad Cars Building [Algoritm avtomatizatsii proektno-konstruktorskikh rabot i ego primenenie v vagonostroenii], *Proceedings of the IV International Scientific and Methodological Conference «Problems of Mathematical and Natural Science Training in Engineering Education» [Sbornik trudov IV Mezhdunarodnoy nauchno-metodicheskoy konferentsii «Problemy matematicheskoy i estestvennonauchnoy podgotovki v inzhenernom obrazovanii»] (03 November 2016, St. Petersburg)*, St. Petersburg, PSTU, 2017, Vol. 2. – Pp. 16–21.

11. Bubnov V.P., Sultonov Sh. Kh. Application of Computer-Aided Design Systems in Engineering [Primenenie sistem avtomatizirovannogo proektirovaniya v mashinostroenii], *Intellectual Technologies on Transport [Intellektual`nye tekhnologii na transporte]*, 2017, No. 1. – Pp. 48–51.

12. ISO/IEC/IEEE 29148:2018. Systems and software engineering – Life cycle processes – Requirements engineering.

13. GOST R 57193-2016 System and software engineering. Systems life cycle processes. [GOST R 57193-2016. Sistemnaya i programmnaya inzheneriya. Protsessy zhiznennogo tsikla sistem], Moscow, StandardInform, 2016. – 97 p.

14. Troelsen A. C# and the .NET [C# i platforma .NET], St. Petersburg, Piter Publish. House, 2006. – 795 p.

15. Freeman A., Sanderson S. Pro ASP.NET MVC 4. 4th edition, Apress, 2012. – 756 p.

16. Rakhimov R.V., Sultonov Sh.Kh. «VagonSila» Software: Certificate of official registration of computer program No. DGU 05998 from 24.01.2019 Republic of Uzbekistan; Declare DGU 2018 0873 dd. 07.11.2018.

Calculation of Quality of ISP Services in Central Region of Iraq Using Statistics

M.Sc. W.A. Al-Nuaami, PhD. M.M. Khalil, PhD A.Kh. Abbas
College of Education for Pure Science of University of Diyala
Baqubah, Republic of Iraq
Uby.waleed2006@gmail.com, maadalomar@gmail.com, Ahmed.k.abbas@outlook.com

Abstract. **The paper investigates the characteristics of Internet usage and quality of service in central region of Iraq, It shows the potential of the Iraqi market and how eager the Iraqi people are for better services. We also point out the main difficulties and obstacles that prevent the development of good internet infrastructure. Many statistical and mathematical methods were used to show the orientation of internet usage, and a survey of 300 internet users was held and the results are shown and discussed.**

Keywords: **Customer Service Quality, Internet Service Provider (ISP), Network Quality.**

Introduction

Internet entered Iraq before the war in 2003 but it was very limited and controlled by the government and not affordable to the majority because it was considered as a threat for the government. Now Iraq's internet markets are expanding significantly but still with many obstacles in its way, the largest obstacle to the expansion and the main reason behind the lack of good internet services in Iraq is the poor and damaged infrastructure, which was damaged during the first and second gulf wars and the years of international sanctions.

After three wars and many internal conflicts in Iraq many people who have left the country use internet to keep in contact with their family, friends and relatives in Iraq mainly through chatting and social media. Also after many years of sanctions the education sector is thirsty for anything new, so the internet serves as a resource for research and software especially with the poor equipped libraries in Iraq. It's also used as an archive where people store many old documents and photos as the chaos threatens the entire country heritage. The political parties use the social media and western news media as a broker for news and election campaigns.

The government is steadily developing most of its facilities and upgrading from paperwork to digital work and automation, but still has a long road to go with many obstacles [1].

Iraq compared with Arab neighbours and worldwide

Iraq ranks Number (38) worldwide in number of individuals using internet and ranks number (123) worldwide in percentage of population using the internet. Which means that a large number of its population (almost 50%) does not have access to the internet, which is a very large number as the population of Iraq exceeded 40 million. This not only shows the size of the potential market, but also exposes the size of difficulties and problems the government is facing to make internet accessible and reachable to every citizen.

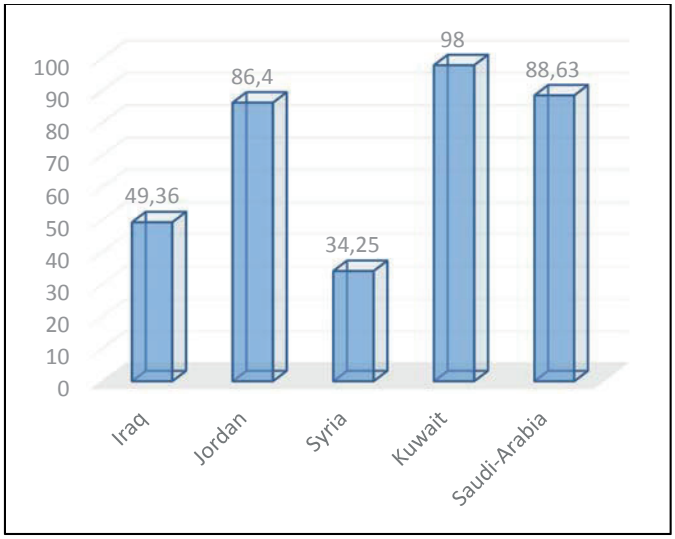

Fig. 1. Individuals using the Internet (% of population)

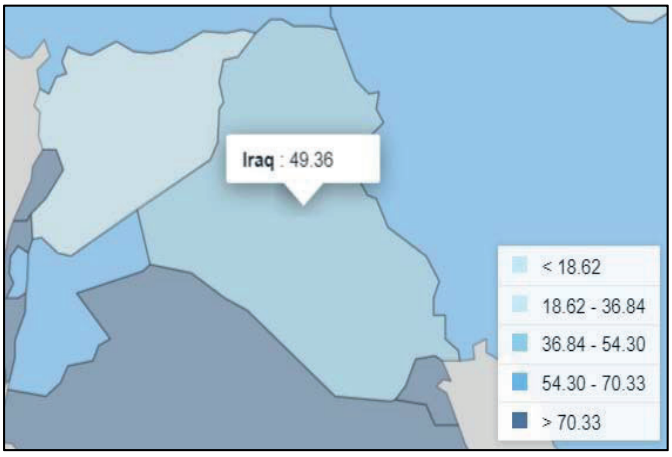

Fig. 2. Map of Iraq compared to its neighbours in % of population using internet

Figures 1 and 2 show the percentage of population that have access to internet compared to its Arab neighbours, clearly Iraq is way behind most of its Arab neighbours, as the graph shows that half of the Iraqi population is still not connected.

On the other hand, the half that does have access to the internet is very actively using it. Although the week service and infrastructure, Iraq's high population compared to its Arab neighbours makes it one of the highest users by number of penetration. This can only show that the Iraqi market is a thirsty and high demanding market for more internet services. The graph below (fig. 3) shows where Iraq lies in number of penetration compared to its Arab neighbours.

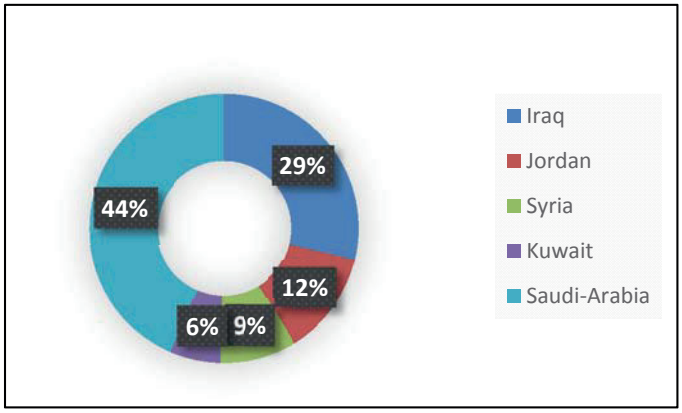

Fig. 3. Total use of Internet by number of penetration

In other words, if all of these countries were connected by one cable, then Iraq would have been using 29% of the traffic in that cable. Which shows the potential of the Iraqi market if Iraq's other half of the population had access to the internet.

IRAQ`S INFRASTRUCTURE

There were significant developments in Internet activity, with the optical cable network covering most of Iraq's territory (approximately 8,500 km) and totally controlled by the government (ITPC (Iraq Telecommunications & Post Co.)), The network was connected to submarine cables (GBI, FALCON) and (Terrestrial) cable from Kuwait from the south. Also the mobile phone companies that are the backbone of the e-government project play a large role in providing internet access, with the number of mobile Internet lines by operators in Iraq in 2015 amounting to 5,7 million, as well as 226,100 wireless Internet lines by operators in Iraq. The number of Internet subscribers increased from 130,200 in 2013 to 2,209 million in 2016, with an annual growth rate of 157 % due to the fact that the Internet is a modern and necessary means of communication. The figure 4 below shows Iraq's fibre optic nodes and connectivity to the rest of the world.

IRAQ`S CENTRAL REGION

Iraq is a federal parliamentary republic consisting of 18 governorates (provinces) including one autonomous region (Iraqi Kurdistan) in the north. However, the most populated area in Iraq is the central region, having Baghdad with a population of over 8 million and Hilla province above 2 million, adding to them Diyala, Salah Ad Din and Al-Anbar provinces the overall population exceeds 15 million, from that number, 73% are urban population [1]. The internet infrastructure of the region is relatively new but poor, as the area witnessed many violence and political issues through the past years.

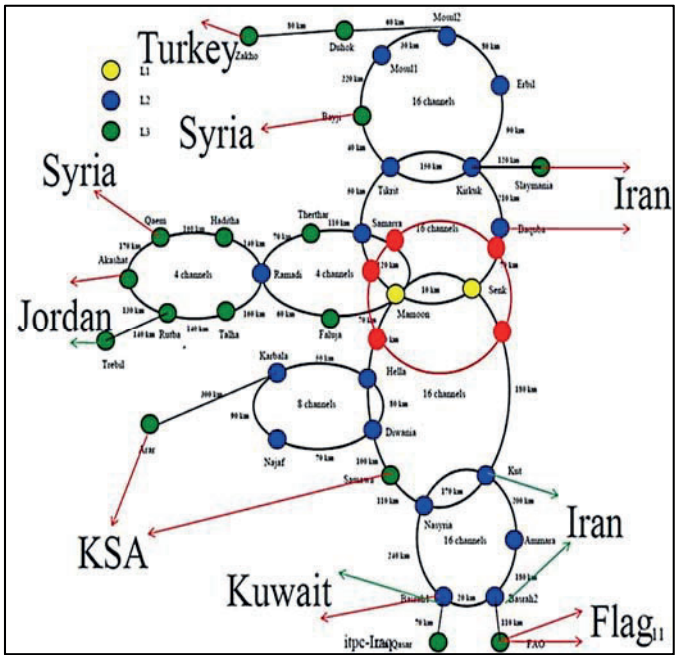

Fig. 4. Iraq's fibre optic nodes and connectivity

Where $Flag_{11}$ represents the connection with the submarine cables (GBI, FALCON).

RELATED WORK

Due to the fact that Iraq is relatively new to the world of Internet, not much can be found investigating the quality of service in that troubled region of the world and if found then not so accurate. However, there is some good work that should be considered like the investigation that Bilal [2] did by also using a survey and analysing the results statistically but they only managed to study the relatively secure and safe area of Iraq Kurdistan covering a population of just over 5 million. Again covering only the Kurdistan area the study of Muhammed Anwar [3] showed that security and privacy have a considerable positive impact on Trust. When it comes to Customer Value and that only Network Quality has a significant positive impact and Information Quality is the only factor which is directly correlated to Customer Commitment. An older study by Firas H. Al-Hammadany [4] exposed the lack of adequate resources, insufficient incentives and encouragement from the government, and social inhibitions.

METHODOLOGY

A quantitative research method is used to investigate and evaluate Internet usage in central region of Iraq, by using an online survey using (Google Forms), data was collected by the answers of 32 questions. People can answer the questions from almost any web browser – including Laptops, mobile Smartphone, and Tablet browsers [5, 6]. The questions were designed to serve the objectives of this research and the results of the questionnaire completed by respondents who uses internet in central region. The survey was available online for 3 days on May 2019 in which all the data was collected and responds from other regions where neglected. The Questions were simple and written in both Arabic and English languages to be understood by participants and they covered personal topics like age and education to

specific topics (frequency of internet usage, reliability, quality, price, preferred time of day, speed, type of devices used, etc.).

Findings and discussions

The Data collected from the survey showed many interesting facts, but first the table below shows the diversity of the participants:

Diversity of survey participants

	gender		Age group				Level of education			
	male	female	15-24	25-34	35-44	45-over	High school	bachelor	High education	other
No.	193	107	64	123	77	36	46	162	79	13
%	64.3	35.6	21.3	41	25.7	12	15.3	54	26.3	4.3

The results showed that although there is a large number of ISP's actively working in the central region, though the majority of users prefer only a handful of ISP's, the figure 5 below shows the main Service provider players in the central region of Iraq's internet market.

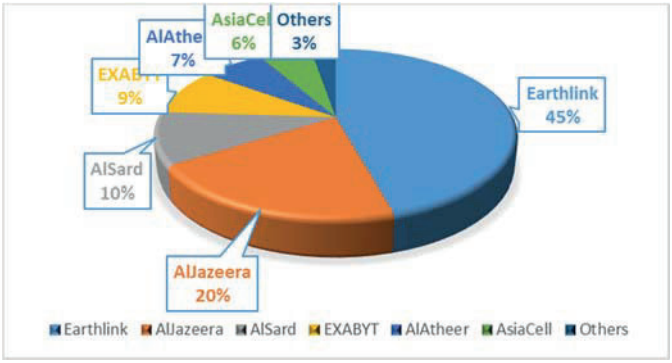

Fig. 5. Main ISP share in Central region of Iraq

The average speed was collected and tested using different software and in different times of day for each provider and an average was calculated using Geometric Mean Return Formula [7, 8] and gave the bellow results in figure 6.

$$m = \sqrt[n]{(1+r1)\times(1+r2)\times\ldots(1+rn)-1},$$

where r = rate of return, n = number of periods.

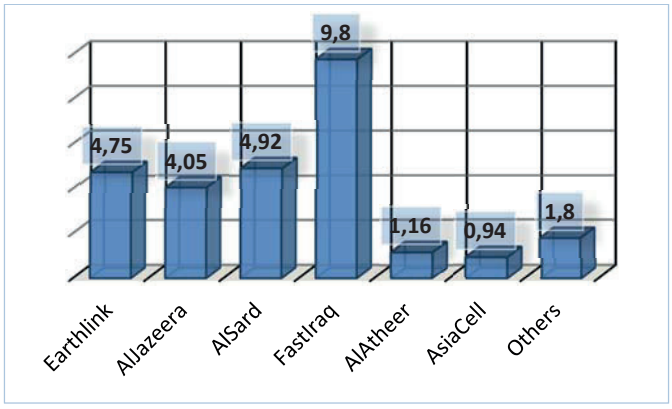

Fig. 6. Average download speed in Mbps

Other factors like coverage, reliability, price were calculated and scaled to a degree of 1 to 10 and gave the following results in figure 7. Due to electricity shortage in Iraq, continuous and reliable internet availability is a valuable factor that is demanded by most users. Coverage is also an issue where only highly populated urban areas are covered leaving many low density populated areas with minimal available options (mostly cell phone networks).

Correlation is a statistical measure that indicates the extent to which two or more variables fluctuate together. A positive correlation indicates the extent to which those variables increase or decrease in parallel; a negative correlation indicates the extent to which one variable increases as the other decreases [9, 10]. So we tested to see the correlation between customer satisfaction and the following results as shown in figure 8.

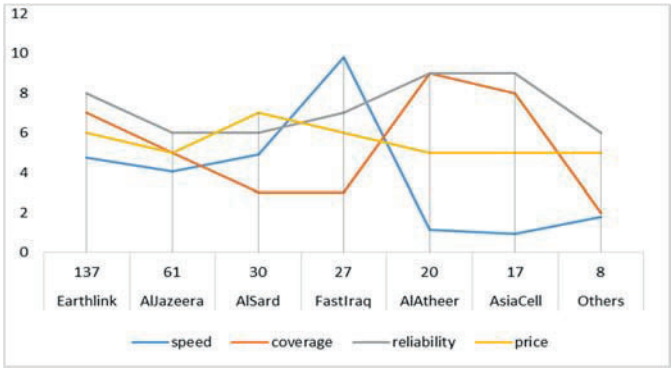

Fig. 7. ISP's score for multiple factors

To calculate the correlation, one of the most commonly used formulas in stats is Pearson's correlation coefficient formula it was used as shown below.

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}},$$

where n is the sample size.

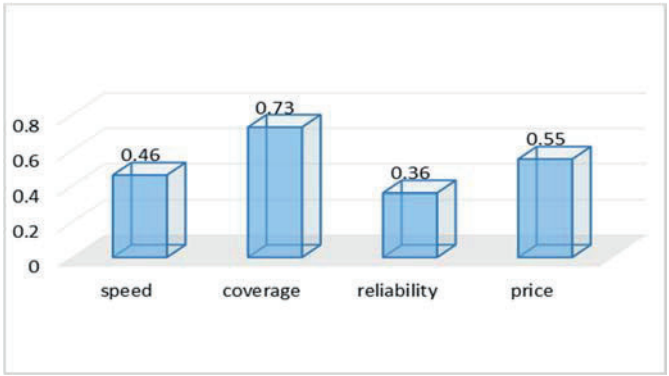

Fig. 8. Customer Satisfaction

Also the correlation of wiliness of a customer to pay more for certain factors was calculated and the results are shown below in figure 9.

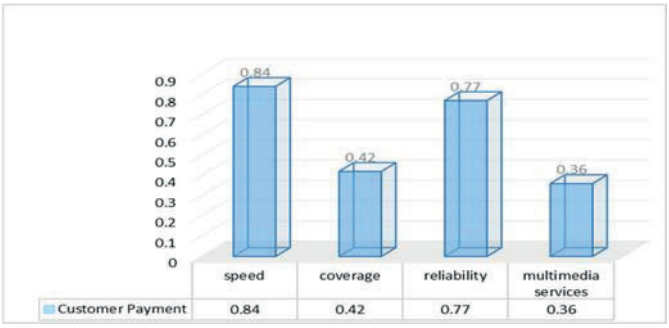

Fig. 9. Correlation of wiliness to pay more

Conclusion

Clearly the results show that Iraq is a thirsty market for good internet services, a country with a potential of millions of new internet users and millions of very active users although the low quality services and infrastructure. The study showed a decent understanding of the situation and characteristics of internet users in Central Region of Iraq, which can be beneficial for ISPs and governmental agencies in planning future internet trend in this area. The majority of participants expressed there wiliness to pay more for better services as all the correlations were positive. Also it is very clear that more studies and interest should be done on this developing part of the world, good investment could come out of it.

References

1. National Development Plan 2018–2022, Ministry of Planning Republic of Iraq, June 2018. – 235 p.

2. Rashid B.N., Faraj A.A. and Shareef T.H., Investigating and Evaluating Internet Usage in Kurdistan Region of Iraq, *International Journal of Multidisciplinary and Current Research*, 2016, Vol. 4. – Pp. 474–479.

3. Anwar M., Nawzad S., Qadir B. Customer Perceptions on Internet Services in Kurdistan Region of Iraq, *International Journal of Social Sciences & Educational Studies*, 2018, Vol. 5, No. 1. – Pp. 28–51.

4. Al-Hammadany F.H., Heshmati A. Determinants of Internet Use in Iraq, *International Journal of Communication*, 2011, Vol. 5. – Pp. 1967–1989.

5. Aydinli C., Demir A. Impact of Non-Technical Dimensions of Service Quality on the Satisfaction, Loyalty, and the Willingness to Pay More: A Cross-National Research on GSM Operators, *International Journal of Economics, Commerce and Management*, 2015, Vol. 3, Is. 11. – Pp. 1–16.

6. Demir A. Importance of Data Analysis on Achieving the Organizational Goals during the Short Term Strategic Plan: Case of Service Quality and Student`s Satisfaction Level at Ishik University, *International Journal of Social Sciences and Educational Studies*, 2017, Vol. 3, No. 3. – Pp. 110–121.

7. Hastie T., Tibshirani R. Friedman J., The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer-Verlag, New York, 2001. – 533 p.

8. Grigoriev O.A., Filipchenko S.A., Khakhin I.S., Yashin A.I. Innovative Thinking in Approaches to the Assessment and Analysis of the Transport Complex [Innovatsionnoe myshlenie v podkhodakh k otsenke i analizu transportnogo kompleksa], *Intellectual Technologies on Transport [Intellektual`nye tekhnologii na transporte]*, 2015, No. 1 (1). – Pp. 5–10.

9. Bubnov V.P. et al. Information system modules: Study guide [Modeli informatsionnykh sistem: Uchebnoe posobie], Moscow, Training Center Education in Railway Transport, 2015. – 188 p.

10. Gerasimenko P.V., Stasishina A.Ye. Forecasting the Size of the Grant Facilities for the Transportation of Passengers by Rail, *Intellectual Technologies on Transport [Intellektual`nye tekhnologii na transporte]*, 2015, No. 2 (2) – Pp. 43–46.

Расчет качества услуг интернет-провайдера в центральном регионе Ирака с использованием статистики

В.А. Ал-Нуайми, канд. техн. наук М.М. Халил, канд. техн. наук А.Х. Аббас
Педагогический колледж чистой науки университета Диялы
Бакууба, Республика Ирак
Uby.waleed2006@gmail.com, maadalomar@gmail.com, Ahmed.k.abbas@outlook.com

Аннотация. **Рассматриваются особенности использования Интернета и качества обслуживания в центральном регионе Ирака. Демонстрируется потенциал иракского рынка и стремление иракского народа получать более качественные услуги. Указаны основные трудности и препятствия, мешающие развитию хорошей интернет-инфраструктуры. Многие статистические и математические методы были применены, чтобы показать ориентацию использования Интернета, был проведен опрос 300 интернет-пользователей, результаты показаны и обсуждены.**

Ключевые слова: **качество обслуживания клиентов, интернет-провайдер (ISP), качество сети.**

Литература

1. National Development Plan 2018–2022, Ministry of Planning Republic of Iraq, June 2018. – 235 p.

2. Rashid B.N., Faraj A.A. and Shareef T.H., Investigating and Evaluating Internet Usage in Kurdistan Region of Iraq, *International Journal of Multidisciplinary and Current Research*, 2016, Vol. 4. – Pp. 474–479.

3. Anwar M., Nawzad S., Qadir B. Customer Perceptions on Internet Services in Kurdistan Region of Iraq, *International Journal of Social Sciences & Educational Studies*, 2018, Vol. 5, No. 1. – Pp. 28–51.

4. Al-Hammadany F.H., Heshmati A. Determinants of Internet Use in Iraq, *International Journal of Communication*, 2011, Vol. 5. – Pp. 1967–1989.

5. Aydinli C., Demir A. Impact of Non-Technical Dimensions of Service Quality on the Satisfaction, Loyalty, and the Willingness to Pay More: A Cross-National Research on GSM Operators, *International Journal of Economics, Commerce and Management*, 2015, Vol. 3, Is. 11. – Pp. 1–16.

6. Demir A. Importance of Data Analysis on Achieving the Organizational Goals during the Short Term Strategic Plan: Case of Service Quality and Student`s Satisfaction Level at Ishik University, *International Journal of Social Sciences and Educational Studies*, 2017, Vol. 3, No. 3. – Pp. 110–121.

7. Hastie T., Tibshirani R. Friedman J., The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer-Verlag, New York, 2001. – 533 p.

8. Инновационное мышление в подходах к оценке и анализу транспортного комплекса / О.А. Григорьев, С.А. Филипченко, И.С. Хахин, А.И. Яшин // Интеллектуальные технологии на транспорте. – 2015. – № 1 (1). – С. 5–10.

9. Модели информационных систем : учебное пособие / В. П. Бубнов, М. Л. Глухарев, А. А. Корниенко [и др.] ; под ред. А. Д. Хомоненко. – М. : Учебно-методический центр по образованию на железнодорожном транспорте, 2015. – 188 с. – (Высшее образование).

10. Gerasimenko P.V., Stasishina A.Ye. Forecasting the Size of the Grant Facilities for the Transportation of Passengers by Rail // Интеллектуальные технологии на транспорте. – 2015. – № 2 (2). – С. 43–46.