

с е т е в о й н а у ч н ы й ж у р н а л ISSN 2518-1092

НАУЧНЫЙ РЕЗУЛЬТАТ

R E S E A R C H R E S U L T

Том 2 | № 2
Volume 2 |

ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ

INFORMATION
TECHNOLOGY

Сайт журнала:
rinformation.ru

сетевой научный рецензируемый журнал
online scholarly peer-reviewed journal



Журнал зарегистрирован в Федеральной службе по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзор)
Свидетельство о регистрации средства массовой информации Эл. № ФС77-69101 от 14 марта 2017 г.

The journal has been registered at the Federal service for supervision of communications information technology and mass media (Roskomnadzor)
Mass media registration certificate El. № FS 77-69101 of March 14, 2017



Том 2, № 2. 2017

СЕТЕВОЙ НАУЧНО-ПРАКТИЧЕСКИЙ ЖУРНАЛ

Издается с 2016 г.

ISSN 2518-1092



Volume 2, № 2. 2017

ONLINESCHOLARLYPEER-REVIEWED JOURNAL

First published online: 2016

ISSN 2518-1092

РЕДАКЦИОННАЯ КОЛЛЕГИЯ:

ГЛАВНЫЙ РЕДАКТОР: **Жиликов Е.Г.**, доктор технических наук, профессор, заведующий кафедрой информационно-телекоммуникационных систем и технологий Белгородского государственного национального исследовательского университета.

ЗАМЕСТИТЕЛЬ ГЛАВНОГО РЕДАКТОРА: **Черноморец А.А.**, кандидат технических наук, профессор кафедры прикладной информатики и информационных технологий Белгородского государственного национального исследовательского университета.

ОТВЕТСТВЕННЫЙ СЕКРЕТАРЬ: **Болгова Е.В.**, старший преподаватель кафедры прикладной информатики и информационных технологий Белгородского государственного национального исследовательского университета.

РЕДАКТОР АНГЛИЙСКИХ ТЕКСТОВ СЕРИИ: **Ляшенко И.В.**, кандидат филологических наук, доцент

ЧЛЕНЫ РЕДАКЦИОННОЙ КОЛЛЕГИИ:

Ломакин В.В., кандидат технических наук, заведующий кафедрой прикладной информатики и информационных технологий НИУ «БелГУ»

Гахова Н.Н., кандидат технических наук, доцент кафедры прикладной информатики и информационных технологий НИУ «БелГУ»

РЕДАКЦИОННЫЙ СОВЕТ:

Волчков В.П., доктор технических наук, профессор (Московский технический университет связи и информатики, г. Москва)

Дмитриенко В.Д., доктор технических наук, профессор (Харьковский национальный технический университет «ХПИ», г. Харьков, Украина)

Капалин В.И., доктор технических наук, профессор (Московский государственный институт электроники и математики (технический университет), г. Москва)

Корсунов Н.И., заслуженный деятель науки РФ, доктор технических наук, профессор (Белгородский государственный национальный исследовательский университет, г. Белгород)

Ломазов В.А., доктор физико-математических наук, профессор (Белгородский государственный аграрный университет им. В.Я. Горина, г. Белгород)

Маторин С.И., доктор технических наук, профессор (Белгородский государственный национальный исследовательский университет, г. Белгород)

Рубанов В.Г., заслуженный деятель науки РФ, доктор технических наук, профессор (Белгородский государственный технологический университет им. В.Г. Шухова, г. Белгород)

Белов С.П., доктор технических наук, профессор (Белгородский государственный национальный исследовательский университет, г. Белгород)

Коськин А.В., доктор технических наук, профессор (Орловский государственный университет им. И. С. Тургенева, г. Орел)

Иващук О.А., доктор технических наук, профессор (Белгородский государственный национальный исследовательский университет, г. Белгород)

EDITORIAL TEAM:

EDITOR-IN-CHIEF: **Evgeniy G. Zhilyakov**, Doctor of Technical Sciences, Professor, Belgorod State National Research University

DEPUTY EDITOR-IN-CHIEF: **Andrey A. Chernomorets**, Candidate of Technical Sciences, Associate Professor, Belgorod State National Research University

EXECUTIVE SECRETARY: **Evgeniya V. Bolgova**, Senior Lecturer, Belgorod State National Research University

ENGLISH TEXT EDITOR: **Igor V. Lyashenko**, Ph.D. in Philology, Associate Professor

EDITORIAL BOARD:

Vladimir V. Lomakin, Candidate of Technical Sciences, Professor, Belgorod State National Research University

Nina N. Gahova, Candidate of Technical Sciences, Associate Professor, Belgorod State National Research University

CONSULTING EDITORS:

Valery P. Volchkov, Doctor of Technical Sciences, Professor (Russia)

Valery D. Dmitrienko, Doctor of Technical Sciences, Professor (Ukraine)

Vladimir I. Kapalin, Doctor of Technical Sciences, Professor (Russia)

Nikolay I. Korsunov, Honoured Science Worker of Russian Federation, Doctor of Technical Sciences, Professor (Russia)

Vadim A. Lomazov, Doctor of Physico-mathematical Sciences, Professor (Russia)

Sergey I. Matorin, Doctor of Technical Sciences, Professor (Russia)

Vasily G. Rubanov, Honoured Science Worker of Russian Federation, Doctor of Technical Sciences, Professor (Russia)

Sergey P. Belov, Doctor of Technical Sciences, Professor (Russia)

Alexander V. Koskin, Doctor of Technical Sciences, Professor (Russia)

Olga A. Ivaschuk, Doctor of Technical Sciences, Professor (Russia)

Учредитель: Федеральное государственное автономное образовательное учреждение высшего образования

«Белгородский государственный национальный исследовательский университет»

Издатель: НИУ «БелГУ». Адрес издателя: 308015 г. Белгород, ул. Победы, 85.

Журнал выходит 4 раза в год

Founder: Federal state autonomous educational establishment of higher education
«Belgorod State National Research University»

Publisher: Belgorod State National Research University

Address of publisher: 85 Pobeda St., Belgorod, 308015, Russia

Publication frequency: 4 / year

СОДЕРЖАНИЕ

СИСТЕМНЫЙ АНАЛИЗ И УПРАВЛЕНИЕ

Осипенко А.А., Игнатенкова О.А., Григоров М.С., Басов О.О. Обоснование необходимости совместного применения автоматической оптической инспекции и неразрушающего рентгеновского контроля электронных модулей	3
Савин Л.О., Королёв М.В., Носов М.В. Анализ определяющих параметров и возможностей использования гибких стратегий технического обслуживания для повышения надежности автомобильной техники при ее эксплуатации в особых условиях	9
Седых А., Рязанов Ю.Д. Скриптовый язык «Link» для программирования сценариев исполняемых событийных систем	21
Нестерова Е.В., Игрунова С.В., Зайцева Т.В., Путивцева Н.П., Пусная О.П., Рябцева Я.Н. Обучающая экспертная система: опыт разработки и использования в учебном процессе	34

CONTENTS

SYSTEM ANALYSIS AND PROCESSING OF KNOWLEDGE

Osipenko A.A., Ignatenkova O.A., Grigorov M.S., Basov O.O. Justification for need of combined use of automatic optical inspection and non-destructive x-ray control of electronic modules	3
Savin L.O., Korolev M.V., Nosov M.V. Analysis of influential parameters and the possibilities of using flexible maintenance strategies to improve reliability of automotive vehicles during its operation in special conditions	9
Sedykh A., Ryazanov Y.D. Scripting programming language "Link" for runnable event-driven systems	21
Nesterova E.V., Igrunova S.V., Zaitseva T.V., Putivtseva N.P., Pusnaya O.P., Ryabtseva Ya.N. Teaching expert system: experience in development and use in the learning process	34

ИНФОРМАЦИОННО- ТЕЛЕКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ

Балабанова Т.Н., Лихогодина Е.С., Водуну А.К., Гурьянова О.И. Использование ортогонального базиса для стеганографического кодирования информации в мультимедиа	40
--	-----------

INFORMATION TECHNOLOGIES AND TELECOMMUNICATION

Balabanova T.N., Likhogodina E.S., Vodounou A.C., Guryanova O.I. The using of orthogonal basis for the steganographic coding of information in multimedia	40
---	-----------

**СИСТЕМНЫЙ АНАЛИЗ И УПРАВЛЕНИЕ
SYSTEM ANALYSIS AND PROCESSING OF KNOWLEDGE**

УДК 62-529

DOI: 10.18413/2518-1092-2017-2-2-3-8

Осипенко А.А.
Игнатенкова О.А.
Григоров М.С.
Басов О.О.**ОБОСНОВАНИЕ НЕОБХОДИМОСТИ СОВМЕСТНОГО ПРИМЕНЕНИЯ
АВТОМАТИЧЕСКОЙ ОПТИЧЕСКОЙ ИНСПЕКЦИИ
И НЕРАЗРУШАЮЩЕГО РЕНТГЕНОВСКОГО КОНТРОЛЯ
ЭЛЕКТРОННЫХ МОДУЛЕЙ**

Федеральное государственное казённое военное образовательное учреждение высшего образования «Академия Федеральной службы охраны Российской Федерации», ул. Приборостроительная, 35, г. Орёл, 302034, Россия

e-mail: rijaya_oska@rambler.ru, olesya_ignatenko@mail.ru, gms.orel@mail.ru, oobasov@mail.ru

Аннотация

В работе представлен результат анализа принципов и частично методов контроля качества электронных модулей и их функциональных элементов на разных этапах их производства и эксплуатации. Систематизированы типы дефектов, выявляемые с помощью автоматической оптической инспекции и неразрушающего рентгеновского контроля. Приведены результаты анализа существующего уровня техники в области автоматической оптической инспекции и неразрушающего рентгеновского контроля дефектов электронных модулей. Определены возможности систем, сочетающих данные методы, и, в частности, подсистемы программного обеспечения, а также направления совершенствования алгоритмического и программного обеспечения этих систем в условиях возрастания сложности и неоднородности контролируемых изделий.

Ключевые слова: электронный модуль, оптическая инспекция, неразрушающий рентгеновский контроль, алгоритмическое и программное обеспечение.

UDC 62-529

Osipenko A.A.
Ignatenkova O.A.
Grigorov M.S.
Basov O.O.**JUSTIFICATION FOR NEED OF COMBINED USE OF AUTOMATIC
OPTICAL INSPECTION AND NON-DESTRUCTIVE X-RAY CONTROL
OF ELECTRONIC MODULES**

Federal state military educational institution of higher professional education "Academy of the Federal security service of the Russian Federation", 35 Priborostroitelnaya St, Orel, 302034, Russia

e-mail: rijaya_oska@rambler.ru, olesya_ignatenko@mail.ru, gms.orel@mail.ru, oobasov@mail.ru

Abstract

The result of analysis of the principles and methods in part of quality control of electronic modules and their functional elements at different stages of their production and operation is provided in this paper. The defect types revealed by means of automatic optical inspection and non-destructive X-ray control are systematized. The result of analysis of the existing technique level in the field of automatic optical inspection and non-destructive X-ray control of defects of electronic modules are given. The possibilities of the systems combining these methods, and, in particular, software subsystems, and also the direction of enhancement of knoware and the software of these systems in the conditions of increase of complexity and non-uniformity of controlled products are defined.

Keywords: electronic module, optical inspection, non-destructive X-ray control, knoware and software.

В условиях постоянного развития в области производства конструктивно и функционально законченных радиоэлектронных устройств или радиоэлектронных функциональных узлов, выполненных в модульном или магистрально-модульном исполнении с обеспечением конструктивной, электрической, информационной совместимости и взаимозаменяемости, которые принято называть электронными модулями (ЭМ), промышленность предъявляет к их качеству ряд высоких требований, которые обоснованы сложностью их сборки и необходимостью быть конкурентоспособными на современном рынке [ГОСТ Р 52003-2003]. В Российской Федерации производство ЭМ регламентировано рядом нормативных документов [ГОСТ Р 56427-2015], в которых описаны технические требования к разработке, изготовлению, параметрам, размерам электронных модулей. Чтобы производимая продукция удовлетворяла данным требованиям, необходимо осуществлять процесс контроля качества ЭМ (рис. 1) путем определения наличия поверхностных и скрытых дефектов (табл. 1) с помощью специальных методов контроля на разных этапах их производства и эксплуатации [Миллер Д., 2014; Калинин Н.П., Викторова М.О., 2012; Гафт С., 2010].



Рис. 1. Технологическая схема сборки электронного модуля
Fig. 1. Technological assembly scheme of the electronic module

Таблица 1

Возможности АОИ и НРК по обнаружению и локализации дефектов различных типов

Table 1

Possibilities of AOI and AXI on detection and localization of defects of different types

Тип дефекта	АОИ	НРК
Дефект нанесения паяльной пасты		
- отсутствие паяльной пасты	+	-
- неправильное нанесение паяльной пасты	+	-
Дефекты компонентов ЭМ		
- отсутствие компонента	+	-
- смещение компонента	+	+
- отсутствие вывода компонента	+	-
- неверная полярность компонента	+	-
- неточное совмещение компонента с контактной площадкой	+	-
- приподнятый вывод компонента (эффект «надгробного камня»)	+	+
- установка несоответствующего компонента	+	-
- отсутствие электрического контакта компонента	+	-
- повреждение компонента и царапины на нем	+	-
- разворот компонента на ребро	+	-
- перевернутый компонент	+	-

Дефекты паяных соединений		
- недостаточное количество припоя	+	+
- избыточное количество припоя	+	+
- наплывы или натеки припоя	+	+
- холодная пайка	+	+
- короткое замыкание (мостик припоя)	+	+
- смещение и перекося паяного соединения	+	+
- не образуется галтели с обратной стороны шва	+	+
- неправильная форма галтели	+	+
- осадки на поверхности	+	-
- трещина в зоне паяного соединения	+	+
- некачественное состояние поверхности изделий после пайки	+	-
- недостаточная смачиваемость припоем контактных площадок печатных плат и выводов	-	+
- инспекция свинцовых и бессвинцовых паяных соединений	-	+
- наличие и процентное содержание пустот	-	+
Дефекты внутреннего состояния полупроводниковых приборов		
- качество разварки соединительного проводника кристалл – рамка	-	+
- наличие пустот между подложкой и кристаллом	-	+
- наличие пустот в корпусе	-	+
- микротрещины компонента	-	+
Дефекты изготовления печатной платы		
- качество металлизации переходных отверстий (нарушение металлизации)	-	+
- отклонения диаметра переходного отверстия	-	+
- смещение слоев	-	+
- состояние печатных проводников, в том числе и внутренних слоев	-	+

Самыми распространенными в настоящее время методами контроля качества ЭМ являются оптический и рентгеновский неразрушающий контроль. Автоматическая оптическая инспекция (АОИ) предполагает определение поверхностных дефектов компонентов и конфигурации сборок ЭМ, а также дефектов при нанесении паяльной пасты на них [Дойл Д., 2008]. Для инспекции скрытых соединений (контроль качества BGA, CSP, CGA, μ BGA, FlipChip компонентов) используется неразрушающий рентгеновский контроль (НРК) [ГОСТ 18353-79].

Оператором АОИ и/или НРК выполняется большой объем работ, связанных с анализом изображений, которые были получены в процессе контроля, на предмет наличия и локализации дефектов. Для решения этой проблемы, а также в целях повышения качества проведения инспекций, системы, реализующие представленные выше методы, активно автоматизируются.

Для обнаружения всех возможных типов дефектов (табл. 1) и снижения времени на их локализацию целесообразно совместно использовать оптический и рентгеновский неразрушающий контроль ЭМ (рисунок 2). Соответствующие системы, сочетающие указанные виды контроля, называют комбинированными [Левданский А., 2005; Григоров М.С., Басов О.О., 2015].



Рис. 2. Эффективность совместного применения АОИ и НРК для обнаружения дефектов
Fig. 2. The efficiency of combined use of AOI and AXI for the defect detection

В настоящее время разработкой, выпуском и продажей комбинированных систем оптического и рентгеновского контроля занимаются лишь несколько компаний. Самыми известными из них являются немецкая компания VISCOM и японская компания Yamaha Motor Group. Комбинированные системы данных фирм позволяют распознавать все визуальные и скрытые дефекты ЭМ без их необходимости перемещения из системы в систему.

Самой известной комбинированной системой в настоящее время является универсальная инспекционная система X7056 компании VISCOM (рис. 3). Она позволяет проводить автоматизировано параллельный оптический и трехмерный рентгеновский контроль. X7056 является модульной и по желанию потребителя может комплектоваться различными камерами. Система способна анализировать платы размером до 610x508мм (24"x20"). Ее размер составляет в длину 1,74 м, а вес, в зависимости от комплектации, - от 2500 кг до 3600 кг. В состав X7056 для проведения рентгеновского контроля входит инспекционная трубка VISCOM, предполагающая разрешение от 5 до 20 мкм/пиксель и гарантирующая получение изображения высокой четкости при 3D-инспекции ЭМ. В зависимости от модели трубки и режима ее работы при определенном напряжении (120 или 160 кВ) возможно изменять интенсивность рентгеновского излучения и получать изображение различной яркости.

Для проведения оптической инспекции в X7056 используются ортогональные и угловые камеры с высоким разрешением. Скорость работы системы при рентгеновской инспекции – 6 см²/с, при оптической инспекции – 20-40 см²/с. Система способна обрабатывать цветные изображения. Для работы с X7056 необходим только один оператор.



Рис. 3. Универсальная инспекционная система X7056 компании VISCOM
Fig. 3. The universal inspection system X7056 of the VISCOM company

Исходя из анализа существующего на сегодняшний день программного обеспечения средств оптического и рентгеновского контроля, к его основными возможностями можно отнести [www.viscom.com]:

- создание инспекционной программы с возможностью задания в ней контрольных образцов ЭМ;
- поддержка инспекционной библиотеки, в которой хранятся различные типы компонентов;
- сравнение полученных оптических и рентгеновских изображений с контрольным образцом и определение различий для каждого вида изображения в отдельности;
- классифицирование обнаруженных дефектов и поиск их в базе данных;
- вывод цветного изображения ЭМ (фотографии в реальном цвете) на экран при АОИ и изображения в градациях серого при НРК.

Несмотря на очевидные преимущества, комбинированные системы оптического и рентгеновского неразрушающего контроля и их специализированное программное обеспечение существуют и недостатки:

- данные системы не позволяют производить операции по выявлению дефектов с одновременным использованием оптического и рентгеновского изображения ЭМ с целью облегчения верификации результатов автоматического обнаружения дефектов оператором контроля;
- алгоритмы, реализующие задачу формирования крупномасштабных изображений с высоким разрешением, основаны на сшивке изображений по координатной сетке. В случае нарушения позиционирования фотокамеры или источника рентгеновского излучения, процесс обнаружения дефектов будет связан с большим количеством ошибок;
- программно-алгоритмическое обеспечение рассматриваемых систем является интеллектуальной собственностью производителя, что не позволяет применять его и адаптировать для систем другого типа.

Следовательно, актуальным остается вопрос совершенствования алгоритмического и программного обеспечения комбинированных систем, позволяющего получить новые возможности в решении задач оптической и рентгеновской инспекций в условиях возрастания сложности и неоднородности контролируемых изделий.

Список литературы

1. Гафт С., 2010. Современные методы обеспечения качества и надежности электронных модулей и блоков. Электронные компоненты. 7: 13-17.
2. ГОСТ 18353-79, 1979. Контроль неразрушающий. Классификация видов и методов. М.: ИПК Изд-во стандартов. 12 с.
3. ГОСТ Р 52003-2003, 2003. Уровни разукрупнения радиоэлектронных средств. Термины и определения. М.: ФГУП "Стандартинформ", IV, 7 с.
4. ГОСТ Р 56427-2015, 2015. Пайка электронных модулей радиоэлектронных средств. Автоматизированный смешанный и поверхностный монтаж с применением бессвинцовой и традиционной технологий. Технические

требования к выполнению технологических операций. М.: ФГУП "Стандартинформ". 36 с.

5. Дойл Д., 2008. Раскрытие возможностей автоматической оптической инспекции. Производство электроники: технологии, оборудование, материалы. 4: 65-67.
6. Калинин Н.П., Викторова М.О., 2012. Атлас дефектов паяных соединений: учебное пособие. Томск: Изд-во Томского политехнического университета. 83 с.
7. Левданский А., 2005. Оптический и рентгеновский контроль печатных плат при помощи одной системы. Технологии в электронной промышленности. 6: 52-54.
8. Миллер Д., 2014. Возможности и перспективы АОИ и рентгеновского контроля. Производство электроники, 7: 124-127.
9. Григоров М.С., Басов О.О., 2015. Метод формирования рентгеновского мультиизображения изделия микроэлектроники с неоднородной структурой. Научные ведомости БелГУ. Серия Экономика. Информатика. 7 (204): 67-72.
10. AOI programming with EasyPro3D. URL: www.viscom.com/europe/products/software/.
11. Viscom auxiliary modules. URL: www.viscom.com/europe/products/software/.
12. X7056RS. In-line X-ray and optical inspection for electronic assemblies. URL: www.viscom.com/europe/products/software/.

References

1. Gaft S., 2010. Modern methods of ensuring the quality and reliability of electronic modules and units. Electronic components. 7: 13-17.
2. GOST 18353-79, 1979. Non-destructive testing. Classification of species and methods. Moscow: IPK Publishing Standards. 12 p.
3. GOST 52003-2033, 2003. Levels of disaggregation of radio electronic means. Terms and Definitions. Moscow: FSUE "Standartinform", IV, 7 p.
4. Soldering of electronic modules of radio-electronic means. Automated mixed and surface mounting using lead-free and traditional technologies. Technical requirements for the execution of technological operations. Moscow: FSUE "Standartinform". 36 p.
5. Dojl D., 2008. Opportunity of automatic optical inspection. Electronics production: technologies, equipment, materials. 4: 65-67.
6. Kalinichenko N.P., Viktorova M.O., 2012. Atlas of defects of soldered joints: a manual. Tomsk: Publishing house of Tomsk Polytechnic University. 83 p.
7. Levdanskiy A., 2005. Optical and X-ray inspection of printed circuit boards using a single system. Technologies in the electronics industry. 6: 52-54.
8. Miller L., 2014. Opportunities and prospects of IDF and X-ray control. Electronics production, 3: 124-127.
9. Grigorov M.S., Basov O.O., 2015. Method of formation of the x-ray multiimage of the product of microelectronics with non-uniform structure. Belgorod State University Scientific Bulletin. Economics Information technologies. 7 (204): 67-72.
10. AOI programming with EasyPro3D. URL: www.viscom.com/europe/products/software/.
11. Viscom auxiliary modules. URL: www.viscom.com/europe/products/software/.
12. X7056RS. In-line X-ray and optical inspection for electronic assemblies. URL: www.viscom.com/europe/products/software/.

Осипенко Анна Александровна, сотрудник Академии Федеральной службы охраны Российской Федерации
Игнатенкова Олеся Александровна, сотрудник Академии Федеральной службы охраны Российской Федерации

Григоров Михаил Сергеевич, кандидат технических наук, сотрудник Академии Федеральной службы охраны Российской Федерации

Басов Олег Олегович, кандидат технических наук, сотрудник Академии Федеральной службы охраны Российской Федерации

Osipenko Anna Alexandrovna, Academy of the Federal security service of the Russian Federation

Ignatenkova Olesya Aleksandrovna, Academy of the Federal security service of the Russian Federation

Grigorov Mikhail Sergeevich, Candidate of Technical Sciences, Academy of the Federal security service of the Russian Federation

Basov Oleg Olegovich, Candidate of Technical Sciences, Academy of the Federal security service of the Russian Federation

УДК 621.3

DOI: 10.18413/2518-1092-2017-2-2-9-20

Савин Л.О.
Королёв М.В.
Носов М.В.

**АНАЛИЗ ОПРЕДЕЛЯЮЩИХ ПАРАМЕТРОВ И ВОЗМОЖНОСТЕЙ
ИСПОЛЬЗОВАНИЯ ГИБКИХ СТРАТЕГИЙ ТЕХНИЧЕСКОГО
ОБСЛУЖИВАНИЯ ДЛЯ ПОВЫШЕНИЯ НАДЕЖНОСТИ
АВТОМОБИЛЬНОЙ ТЕХНИКИ ПРИ ЕЕ ЭКСПЛУАТАЦИИ
В ОСОБЫХ УСЛОВИЯХ**

Федеральное государственное казённое военное образовательное учреждение высшего образования «Академия Федеральной службы охраны Российской Федерации», ул. Приборостроительная, 35, г. Орёл, 302034, Россия

e-mail: leonidys77@yandex.ru, korol80@list.ru, nosovm@mail.ru

Аннотация

В данной статье проанализированы параметры автомобильной техники, определяющие её надежность в зависимости от особых условий ее эксплуатации. По каждому из данных параметров получены вероятностные показатели надежности автомобильной техники при помощи прогнозирования. На основе вероятностных показателей надежности сформированы показатели эффективности функционирования, на основе которых возможна организация гибкой системы технического обеспечения автомобилей с оптимальной периодичностью.

Ключевые слова: параметры автотранспортных средств, надежность автомобильной техники, показатель эффективности, техническое обслуживание.

UDC 621.3

Savin L.O.
Korolev M.V.
Nosov M.V.

**ANALYSIS OF INFLUENTIAL PARAMETERS AND THE POSSIBILITIES
OF USING FLEXIBLE MAINTENANCE STRATEGIES TO IMPROVE
RELIABILITY OF AUTOMOTIVE VEHICLES DURING ITS OPERATION
IN SPECIAL CONDITIONS**

Federal state military educational institution of higher professional education "Academy of the Federal security service of the Russian Federation", 35 Priborostroitelnaya St, Orel, 302034, Russia

e-mail: leonidys77@yandex.ru, korol80@list.ru, nosovm@mail.ru

Abstract

This article explores the options of motor vehicles, defining its reliability depending on the specific conditions of its operation. For each of these parameters obtained probabilistic reliability automotive engineering with forecasting. On the basis of probabilistic reliability indices, the generated performance indicators. Based on these values, it is possible to organize a flexible system of technical support vehicles with optimal frequency.

Keywords: the options of motor vehicles, reliability automotive engineering, performance indicator, the technical support.

В современных условиях роль и значение автомобильного транспорта постоянно увеличиваются. В настоящее время автомобильный транспорт занимает достаточно важное место в транспортной системе РФ, регулярно обслуживая почти 3 млн предприятий и организаций, а также население страны. Согласно некоторым оценкам, вклад автомобильного транспорта в перевозки грузов в настоящее время составляет более 75 %, а пассажиров – около 55 % [Хасанов Р. Х. 2003]. В связи с этим число автомобилей, используемых для решения описанных выше задач, на территории РФ постоянно возрастает (рис. 1).

Достоинства автомобильного транспорта, предопределяющие достаточно высокие темпы его развития, связаны с оперативностью (мобильностью) и гибкостью доставки грузов. При этом очевидно, что эти свойства автомобильного транспорта во многом определяются уровнем работоспособности и техническим

состоянием автомобильного парка, зависящими, во-первых, от надежности самих автомобилей, во-вторых, от мер по обеспечению их работоспособности в процессе эксплуатации, а также от условий эксплуатации [Хасанов Р. Х. 2003].

Схожие тенденции наблюдаются и в области развития военного автомобильного транспорта. Роль автомобилей в Вооруженных Силах (ВС) РФ постоянно растет, появляются новые возможности их применения, при этом число автомобилей также постоянно увеличивается.

Необходимо отметить, что применение автомобильного транспорта для решения задач, стоящих перед ВС РФ, имеет ряд существенных особенностей. Специфика применения этих автомобилей в ВС такова, что большую часть своего времени эти автомобили эксплуатируются в т.н. особых условиях – например, в условиях экстремально низких и высоких температур (Крайнего Севера, Заполярья или в пустынных условиях), в суровых климатических условиях (например, повышенной влажности), в условиях сложного рельефа местности (например, в горах), в условиях бездорожья или полного отсутствия дорог, возможного радиационного и химического заражения местности и т.д. При этом важность выполнения поставленной задачи зачастую оказывается неизмеримо важнее, чем обеспечение целостности автотранспорта и качественного ухода за ним, поскольку поставленные задачи зачастую должны быть выполнены и выполняются любой ценой.



Рис. 1. Численность парка грузовых автомобилей в РФ (на 1.07.2015)
Fig. 1. The number of freight cars in Russia (on 1.07.2015)

Таким образом, к наиболее важным факторам условий эксплуатации, изменяющимся в широких пределах, относятся климатические и дорожные условия. При этом условия эксплуатации автомобильной техники неминуемо влияют на режимы работы ее отдельных агрегатов и деталей, ускоряя или замедляя изменение параметров технического состояния техники в целом.

Ко всем ранее перечисленным факторам, оказывающим влияние на эксплуатацию автотранспорта в особых условиях, необходимо добавить зачастую недостаточно высокий уровень и квалификацию как самих водителей, так и обслуживающего автотранспорт технического персонала (например, военнослужащих срочной службы).

Все указанные выше обстоятельства приводят к тому, что рассчитанные для типовых (усредненных) условий эксплуатации сроки проведения технического обслуживания (ТО), приведенные в технической документации, зачастую оказываются недостаточно обоснованными. Это приводит к ускоренному износу деталей, досрочному выходу узлов и агрегатов из строя и в конечном итоге – к снижению коэффициента готовности автомобильного парка в целом и к повышению непроизводительных затрат на его эксплуатацию (увеличению затрат на ремонт и восстановление работоспособности автомобилей). В связи с этим для решения основных задач технической эксплуатации автомобильного транспорта необходимо изучение закономерностей изменения технического состояния автомобильных узлов, агрегатов и механизмов под влиянием различных факторов в процессе эксплуатации автотехники.

Описанные экстремальные (особые) условия эксплуатации автомобильной техники, как правило, характеризуются одновременным сочетанием нескольких неблагоприятных факторов. Так, для холодного климатического района страны характерны не только низкая температура окружающего воздуха и ветер, но и более тяжелые дорожные условия (снежные заносы, дороги без твердого покрытия и пр.). Для жарких и сухих климатических районов, кроме высокой температуры, действуют факторы солнечной радиации и большой запыленности воздуха, и т.д. При эксплуатации автомобильной техники в особых условиях необходимо учитывать все эти факторы, однако, к сожалению, вопросы такой эксплуатации автотранспорта в настоящее время исследованы и описаны недостаточно глубоко.

Общие вопросы эксплуатации автомобильной техники в особых условиях исследовались, например, в работах [Агеев Е. В., 2015, Ефремов Л. В., 2015]. К сожалению, эти работы носят по большей части теоретический характер, и в них нет практических рекомендаций по использованию конкретных типов автомобильной техники в конкретных условиях эксплуатации.

Отдельные вопросы, посвященные эксплуатации автомобилей в конкретных климатических и дорожных условиях исследовались, например, в работах П.

П. Ощепкова [Ощепков П. П., 2000], В.В. Ионова [Ионов В. В., 2013] и других авторов. Однако таких работ в настоящее время не так много, и почти все они посвящены исследованию лишь строго определенной (т.е. конкретной и почти всегда достаточно узкой) проблемы.

Тем не менее, проведенный анализ имеющихся работ по данной тематике позволил выделить из общей совокупности технических параметров автомобильной техники ряд наиболее важных параметров, которых вплотную связаны с ускоренным выходом из строя автомобильных узлов в особых условиях их эксплуатации. При этом для конкретных условий эксплуатации автотехники эти основные (т.н. определяющие) параметры могут быть различными.

Таким образом, под определяющим параметром автомобильной техники (АТ) будем понимать такой информативный (диагностический) параметр, который в данный текущий момент времени оказывает наибольшее влияние (имеет решающее значение) для обеспечения работоспособности АТ, и при выходе которого за допустимые пределы происходит отказ АТ. Очевидно, что выход одного из таких параметров за допустимые (заданные) пределы приводит к отказу автомобильной техники в целом, и наоборот, поддержание значений указанных параметров в пределах нормы позволит обеспечить работоспособность военной автомобильной техники в течение всего заданного интервала времени (срока службы). Следовательно, при эксплуатации военной автомобильной техники в особых условиях одной из достаточно важных и актуальных задач является поддержание в установленных пределах значений данных параметров. Необходимо предупреждать эти постепенные отказы, т.е. так организовать техническую эксплуатацию, чтобы путем своевременно организованных профилактических работ провести своевременную подрегулировку или замену узла (агрегата). Так как, с одной стороны, пассивное ожидание отказа ведет к потерям из-за простоя, с другой – слишком частые проверки технического состояния приводят к увеличению затрат на обслуживание АТ.

Таким образом, правильно организованная техническая эксплуатация АТ в общем виде предполагает поддержание нужных параметров в пределах нормы в течение заданного срока службы в условиях воздействия случайных внешних факторов, начальных значений этих параметров и их изменений во времени. Решение данной задачи достигается путем регулировки требуемых параметров АТ (при необходимости – и замены ее отдельных узлов), т.е. путем организации технического обслуживания АТ.

Техническое обслуживание – комплекс операций по поддержанию работоспособности или исправности изделия при использовании по назначению, ожидании, хранении и транспортировании. Под техническим обслуживанием АТ в работе будем понимать комплекс работ по контролю (проверке соответствия установленным нормам) значений основных диагностических параметров АТ, и доведению этих значений до требуемых уровней, а под периодичностью ТО АТ – периодичность проведения такого комплекса работ. ТО является одним из этапов процесса эксплуатации [Авдудевский В.С., 1986; Батьковский А.М., Батьковский М.А., Ройкоз Г.А., Чудинов С.М. 2014].

Очевидно, что во время эксплуатации АТ в целом и ее узлы и агрегаты – в отдельности могут находиться как в работоспособном состоянии, при котором значения заданных параметров не превышают своего предельного значения, так и в состоянии отказа. В установленные моменты времени, соответствующие моменту начала очередного ТО, АТ снимается с эксплуатации для контроля и, при необходимости, для восстановления ее работоспособности. Таким образом, кроме режима эксплуатации (использования по прямому назначению), АТ может находиться и в режиме контроля

работоспособности, а при необходимости – и в режиме восстановления работоспособности. Таким образом, т.н. цикл регенерации АТ состоит из трех основных составляющих: из времени использования техники по прямому назначению (времени до момента начала очередного ТО), включающего время безотказной работы и время отказа АТ, а также из времени, затрачиваемого на контроль работоспособности АТ, и из времени восстановления ее работоспособности. Регенерационный цикл повторяется после восстановления работоспособности системы активной защиты информации (рис. 2) [Ефремов Л. В., 2015].

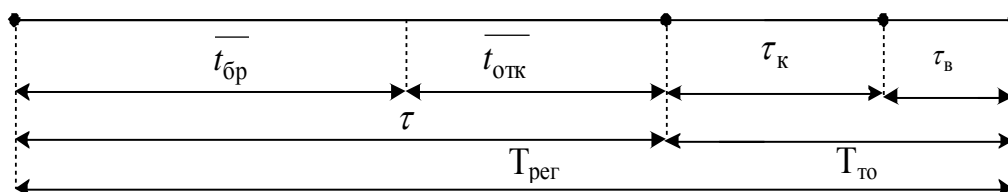


Рис. 2. Цикл регенерации АТ
Fig. 2. Regeneration cycle automotive vehicles

На рисунке 2 приняты следующие обозначения: $T_{\text{рег}}$ – период регенерации; $\bar{t}_{\text{бп}}$ – среднее время безотказной работы АТ на интервале τ ; $\bar{t}_{\text{отк}}$ – среднее время отказа АТ на интервале τ ; τ – периодичность проведения сеансов ТО АТ (момент начала очередного ТО); $T_{\text{ТО}}$ – средняя длительность проведения ТО; $\tau_{\text{к}}$ – средняя длительность контроля работоспособности АТ; $\tau_{\text{в}}$ – средняя длительность восстановления работоспособности АТ.

С термином ТО тесно связано также понятие стратегии ТО – системы правил управления техническим состоянием изделия в процессе ТО. Целью стратегии ТО является выбор управления техническим состоянием изделий в течение их срока службы, позволяющего обеспечить заданный уровень готовности изделий к использованию по назначению, их работоспособность в процессе эксплуатации, минимальные затраты времени, труда и средств на выполнение ТО. Вопросам идентификации технического состояния и стратегиям обслуживания технических систем посвящено достаточно большое количество литературы. При этом современные стратегии обслуживания принято подразделять в общем случае на три вида:

1. Первый вид – обслуживание оборудования после выхода его из строя.

В этом случае оборудование эксплуатируется до его выхода из строя. В основном это касается недорогого вспомогательного оборудования при наличии его резервирования, когда замена оборудования дешевле, чем затраты на его ремонт и обслуживание. В отсутствие резервирования на время ремонта процесс использования по назначению приостанавливается. Описанный вид обслуживания для АТ в силу многих причин на практике практически не применяется.

2. Второй вид обслуживания – обслуживание оборудования по регламенту. Этот вид предусматривает два возможных правила: проведение ТО после выработки ресурса по определенному показателю (ТО по выработке ресурса – например, ТО отдельных узлов и агрегатов АТ через каждые 10 тысяч км пробега) или проведение ТО через фиксированные сроки (календарное обслуживание – например, ежегодное ТО).

Принцип обслуживания по наработке предполагает, что перечень и периодичность выполнения операций определяются значением наработки объекта с начала эксплуатации. Этот принцип применяется, как правило, для организации ТО технических устройств, имеющих подверженные относительно быстрому износу важнейшие элементы и при условии возможности выделить параметр, определяющий работоспособность устройства. Более сложные технические устройства, как правило, имеют в своем составе множество элементов и узлов с различными показателями надежности. В этом случае невозможно выделить определяющий параметр изделия и применить принцип обслуживания по выработке ресурса, и для организации ТО используется календарный принцип. Календарное обслуживание производится в соответствии с рекомендациями завода-изготовителя через фиксированные промежутки времени независимо от технического состояния оборудования. Такое обслуживание является основным для многих технических устройств и по своему содержанию носит плано-предупредительный характер.

ТО по регламенту предусматривает строго определенный объем и периодичность проводимых работ, которые определяются на основе статистических данных, полученных путем организации опытной эксплуатации значительного количества техники в среднестатистических условиях их применения. Период

между проведением ТО устанавливают равным времени, в течение которого определенная доля оборудования, подвергаемого опытной эксплуатации, работает без отказов.

Предупредительный характер ТО по регламенту приводит к тому, что большая часть регулируемых или заменяемых элементов оборудования, как правило, не дорабатывает свой ресурс. При обслуживании по регламенту многие операции ТО выполняются без фактической их необходимости. Кроме того, для многих узлов технических систем обслуживание по регламенту не снижает вероятность выхода их из строя. Исследования показали, что причиной около 70 % дефектов технических средств, находящихся в эксплуатации, является проведение на них ТО [Зеленцов В.А. 1991]. Проведенный анализ показывает, что при обслуживании по регламенту только 30 % от общего времени, затрачиваемого на ТО, тратится на необходимые для поддержания работоспособности операции, а 70 % приходится на операции, без которых можно было бы обойтись. Таким образом, ТО по регламенту характерны завышенные объемы работ и необоснованно частая периодичность их проведения, которые назначаются изготовителем для того, чтобы с запасом обеспечить требуемые показатели надежности различных узлов и агрегатов [Зеленцов В.А. 1991].

С другой стороны, как уже отмечалось выше, при эксплуатации АТ в особых условиях достаточно часто наблюдается иная картина: рассчитанная для типовых условий эксплуатации периодичность ТО, напротив, оказывается недостаточной, т.е. узлы и агрегаты АТ выходят из строя чаще, чем при типовых условиях эксплуатации АТ.

Таким образом, применение ТО по регламенту для АТ при особых условиях ее эксплуатации на практике зачастую оказывается недостаточно эффективным, поскольку не учитывает особенностей временного дрейфа контролируемых параметров и особенностей эксплуатации АТ.

3. Третий вид обслуживания – обслуживание по состоянию, при котором перечень и периодичность выполнения операций определяется фактическим техническим состоянием объекта в момент начала ТО. Этот вид ТО предусматривает оценку срока работоспособности на основе прогноза времени нахождения в допустимых пределах таких параметров технического средства, которые определяют возможность его применения по назначению. Обслуживание по фактическому состоянию относится к гибкой системе ТО.

Для определения периодичности проведения ТО при обслуживании по состоянию используются сведения о закономерностях процессов изменения параметров их элементов и данные контроля этих параметров. Такой подход позволяет обеспечить требуемое качество функционирования технических объектов при устранении недостатков, присущих ТО по регламенту [Ситчихина М. В., 2003].

Обслуживание по состоянию требует периодических измерений значений определенных параметров изделия. Эти измеренные значения являются основой для прогноза технического состояния изделия. На основе прогноза принимается решение о сроках проведения ТО и объеме его операций [Черноморец А.А., Болгова Е.В., Черноморец Д.А., Коваленко А.Н. 2015].

Обслуживание по состоянию достаточно широко используется при эксплуатации различных образцов техники и активно внедряется в целом ряде развивающихся отраслей, особенно в радиоэлектронной промышленности. Применительно же к АТ и в особенности – для экстремальных условий ее эксплуатации такие гибкие системы ТО, к сожалению, в настоящее время разработаны и исследованы пока недостаточно широко.

Многие специалисты по эксплуатации АТ, осознавая эту проблему, предпринимают попытки самостоятельно обосновать периодичность проведения ТО и назначить сроки проведения ТО АТ на базе собственного опыта. В качестве примера можно привести поправочные коэффициенты к нормативам на проведение ТО, приведенные в [Техническая эксплуатация автомобилей: Учебник, 1991].

Так, для жаркого и холодного климата для периодичности ТО авторами введен поправочный коэффициент 0,9; для оценки трудоемкости работ по ТО – от 1,1 до 1,3; для пробега до капитального ремонта коэффициент предусматривает 0,7 – 0,9 установленного пробега, и т.д. Такому подходу присущи субъективность и недостаточная научная обоснованность принятия решений о моментах начала очередного ТО аппаратуры. В связи с этим возникает задача разработки научно обоснованного подхода к определению оптимального периода ТО АТ, при котором АТ удовлетворяла бы всем предъявляемым к ней требованиям по надежности при снижении затрат на эксплуатацию.

Следует отметить, что необходимыми организационными условиями применения ТО по состоянию являются экономическая целесообразность, наличие приборной базы, наличие методики прогнозирования технического состояния, обученный персонал и контролепригодность оборудования. Необходимым техническим условием применения ТО по состоянию является наличие элементов с ярко выраженным

временным дрейфом контролируемых (определяющих) параметров, которые являются определяющими для обеспечения работоспособности устройства [http://thepit.oaosng.ru/babenko_r.htm].

В работах Е.Ю. Барзиловича, Н.Н. Смирнова, В.А. Каштанова и некоторых других авторов показано, что обслуживание по состоянию является наиболее перспективным среди существующих в настоящее время видов ТО. Обслуживание по состоянию позволяет обеспечить требуемую надежность систем (а в ряде случаев и повысить ее) при уменьшении затрат на ТО [Смирнов Н. Н, Ицкевич А. А., 1980; Петухов Г.Б., 1989]. Отечественная и зарубежная практика показывает, что при этом затраты на эксплуатацию сокращаются до 30 %, а ресурс оборудования по сравнению с календарным принципом обслуживания увеличивается на 35 – 40 % [Ситчихина М. В., 2003; http://thepit.oaosng.ru/babenko_r.htm].

Высокая эффективность методов обслуживания по состоянию достигается за счет предупреждения в процессе ТО большего, по сравнению с другими видами ТО, числа отказов, которое реализуется на практике посредством прогнозирования технического состояния. Прогнозирование позволяет эксплуатировать контролируемый объект до появления признаков опасного снижения работоспособности, тем самым продлевая срок службы оборудования за пределы нормативного срока, исключая преждевременные вмешательства в его работу и снижая затраты на обслуживание. Разработка гибких стратегий ТО применительно к АТ и использование разработанных гибких стратегий позволят своевременно проводить ТО АТ, т.е. обеспечить постоянное нахождение значений ОП в пределах нормы, предотвращая тем самым как выход из строя отдельных узлов (агрегатов) АТ, так и отказ АТ в целом. При этом описанный выше подход к определению параметров ТО как этапа эксплуатации АТ может быть реализован совместно со статистическим подходом, например, использоваться для корректирования ТО по регламенту. При этом ТО АТ, проводимое по гибкой стратегии, в отличие от обслуживания по регламенту, позволит учесть такие факторы, как техническое состояние и особые условия эксплуатации АТ для каждого конкретного случая.

Таким образом, на основе анализа имеющихся видов ТО и с учетом их выявленных недостатков достаточно актуальной является задача разработки научно обоснованного подхода к определению оптимального периода ТО АТ $\tau_{\text{опт}}$. При этом под оптимальной периодичностью проведения ТО АТ будем понимать периодичность, при которой обеспечивается выполнение основных требований, предъявляемых к АТ (ее к надежности и затратам на ее эксплуатацию).

Задачу определения оптимальной периодичности ТО целесообразно решать с использованием теории векторного анализа эффективности [Петухов Г.Б., 1989]. В соответствии с данным подходом из общей совокупности свойств АТ могут быть выделены основные свойства, обуславливающие ее пригодность к использованию по назначению и определяющие качество ее функционирования: надежность АТ и ее экономичность (затраты на ее эксплуатацию). Выбранную совокупность свойств назовем качеством функционирования АТ [Петухов Г.Б., 1989].

В качестве первого показателя качества целесообразно использовать показатель готовности АТ, характеризующий ее готовность к использованию по назначению в произвольный момент времени эксплуатации. Для одного цикла регенерации (рис. 2) можно записать соотношение:

$$K_{\text{гр}} = \frac{M[t_{\text{бп}}]}{M[T_p]}, \quad (1)$$

где $M[t_{\text{бп}}]$ – математическое ожидание (МО) длительности безотказной работы АТ за период времени τ ;

$M[T_p]$ – МО периода регенерации T_p .

Показатель готовности, в отличие от коэффициента готовности, учитывает все основные составляющие простоев, которые могут иметь место при функционировании АТ, и может быть представлен в виде функции, зависящей от значений периодичности ТО:

$$K_{\text{гр}}(\tau) = \frac{\int_0^{\tau} P_0(t) dt}{P_0(\tau) \times (\tau + \tau_{\text{к}}) + P_{\text{отк}}(\tau) \times (\tau + \tau_{\text{к}} + \tau_{\text{в}})}, \quad (2)$$

где $\int_0^{\tau} P_0(t) dt = \overline{t_{\text{бп}}}$ – среднее время безотказной работы АТ на интервале τ ;

$P_0(\tau)$ – вероятность безотказной работы АТ на интервале τ ;

$P_{отк}(\tau)$ – вероятность отказа АТ на интервале τ , а остальные обозначения соответствуют ранее введенным.

После преобразования получим:

$$K_{пр}(\tau) = \frac{\int_0^{\tau} P_0(t) dt}{\tau + \tau_{\kappa} + P_{отк}(\tau) \times \tau_{\text{в}}}, \quad (3)$$

Проведенные расчеты и моделирование функции (3) в среде «MathCad» показали, что при определенном значении периодичности проведения сеансов контроля и ТО выражение вида (3) имеет единственный максимум. Значение момента начала очередного ТО, при котором показатель готовности максимален, может быть определено методом перебора с использованием ЭВМ.

В качестве второго показателя качества выбраны средние относительные непроизводительные затраты на эксплуатацию АТ, которые необходимо минимизировать. Их также можно представить в виде функции, зависящей от периодичности проведения ТО:

$$C = \lim_{t \rightarrow \infty} \frac{\sum_{k=1}^n C_k}{t}, \quad (4)$$

где C_k , $k = 1, 2, \dots, n$ – затраты на эксплуатацию элемента АТ в n -м состоянии.

Для одного участка регенерации (цикла обслуживания) можно записать:

$$C = \frac{M(C_{\text{непр}})}{M(C)}, \quad (5)$$

где $M(C_{\text{непр}})$ – МО непроизводительных затрат на эксплуатацию АТ;

$M(C)$ – МО общих затрат на эксплуатацию АТ.

Средние относительные непроизводительные затраты на эксплуатацию также можно представить в виде функции, зависящей от периодичности проведения ТО:

$$C(\tau) = \frac{P_0(\tau) \times (C_{\kappa} \times \tau_{\kappa}) + P_{отк}(\tau) \times (C_{отк} \times \bar{t}_{отк} + C_{\text{в}} \times \tau_{\text{в}} + C_{\kappa} \times \tau_{\kappa})}{P_0(\tau) \times (C_{\phi} \times \bar{t}_{\phi} + C_{\kappa} \times \tau_{\kappa}) + P_{отк}(\tau) \times (C_{отк} \times \bar{t}_{отк} + C_{\text{в}} \times \tau_{\text{в}} + C_{\kappa} \times \tau_{\kappa})}$$

где C_{κ} – средние удельные затраты на эксплуатацию АТ в режиме ТО;

$C_{отк}$ – средние удельные затраты на эксплуатацию АТ в режиме отказа;

$C_{\text{в}}$ – средние удельные затраты на эксплуатацию АТ в режиме восстановления работоспособности;

C_{ϕ} – средние удельные затраты на эксплуатацию АТ при ее безотказной работе (нормальном функционировании);

$\bar{t}_{отк} = \tau - \bar{t}_{\phi}$ – среднее время отказа АТ на интервале τ .

Все остальные обозначения соответствуют ранее введенным.

После преобразования получим итоговое выражение для средних относительных непроизводительных затрат на эксплуатацию АТ:

$$C(\tau) = \frac{C_{\kappa} \times \tau_{\kappa} + P_{отк}(\tau) \times (C_{отк} \times \bar{t}_{отк} + C_{\text{в}} \times \tau_{\text{в}})}{P_0(\tau) \times C_{\phi} \times \bar{t}_{\phi} + P_{отк}(\tau) \times (C_{отк} \times \bar{t}_{отк} + C_{\text{в}} \times \tau_{\text{в}}) + C_{\kappa} \times \tau_{\kappa}} \quad (6)$$

Проведенные расчеты и моделирование функции в среде «MathCad» показали, что при определенном значении периодичности проведения сеансов контроля и ТО выражение вида (6) имеет единственный

минимум. Значение момента начала очередного ТО, при котором средние непроизводительные затраты на эксплуатацию минимальны, также могут быть определены методом перебора с использованием ЭВМ.

Очевидно, что первый из введенных показателей необходимо свести к максимуму, а второй – к минимуму. Иными словами, критерии оптимальности для введенных показателей вида (3) и (6) могут быть представлены следующим образом.

1. При постановке и решении прямой задачи – в виде системы:

$$\begin{cases} K_{\text{пр}}(\tau_{\text{опт}}) = \sup_{\tau_{\text{опт}} \in T_c} K_{\text{пр}}(\tau), \\ C(\tau) \leq C_{\text{доп}} \end{cases}$$

2. При обратной задаче

$$\begin{cases} C(\tau_{\text{опт}}) = \inf_{\tau_{\text{опт}} \in [0; T_{\text{пр}}]} C(\tau), \\ K_{\text{пр доп}}(\tau) \geq K_{\text{пр доп}} \end{cases}$$

где $C_{\text{доп}}(\tau)$ – допустимое значение затрат на эксплуатацию АТ;

T_c – множество возможных значений периодичности проведения ТО, при которых выполняется условие $C(\tau) \leq C_{\text{доп}}$;

$[0; T_{\text{пр}}]$ – интервал возможных значений периодичности ТО (область пригодности), формируемый при выполнении условия

$$K_{\text{пр доп}}(\tau) \geq K_{\text{пр зад}}.$$

На практике чаще используется обратная постановка задачи, т.е. необходимо определить оптимальную периодичность ТО АТ $\tau_{\text{опт}}$, при которой обеспечивается заранее заданное значение показателя готовности АТ при минимально возможных при этом затратах на ее эксплуатацию. Данная обратная задача может быть представлена в виде:

$$C(\tau) \longrightarrow \min_{\tau_{\text{опт}} \in [0; T_{\text{пр}}]}.$$

Под эффективностью функционирования АТ будем понимать степень достижения указанных выше требований к показателю готовности и к затратам на эксплуатацию АТ. При этом оценка оптимальности выбранной периодичности ТО АТ должна производиться по изменению эффективности функционирования АТ, т.е. по степени удовлетворения вышеприведенных требований к показателям качества. Для этого была введена скалярная мера удовлетворения показателями качества заданным к ним требованиям – показатели эффективности функционирования (ПЭФ) АТ.

В качестве частных ПЭФ АТ в работе выбраны следующие показатели:

1. $K_{\text{пр}}(\tau)$ – показатель готовности АТ, который должен удовлетворять условию $K_{\text{пр}}(\tau) \geq K_{\text{пр зад}}$;
2. Разность между средними относительными непроизводительными затратами на эксплуатацию АТ и их минимальным значением:

$$C(\tau) - C_{\text{min}} = C(\tau)$$

Примеры расчетов зависимости $C(\tau)$ от периодичности проведения ТО приведены на рис. 3.

Все введенные ПЭФ безразмерны, нормированы к единице, имеют одинаковый порядок и зависят от периодичности проведения ТО. При этом $K_{\text{пр}}$ формирует т.н. область пригодности, т.е. интервал допустимых значений времени для поиска оптимального периода ТО.

На основе введенных частных ПЭФ может быть сформирован т.н. обобщенный показатель эффективности функционирования (ОПЭФ) АТ в виде:

$$\text{ОПЭФ}(\tau) = C(\tau) [K_{\text{пр}}(\tau) \geq K_{\text{пр зад}}] \quad (7)$$

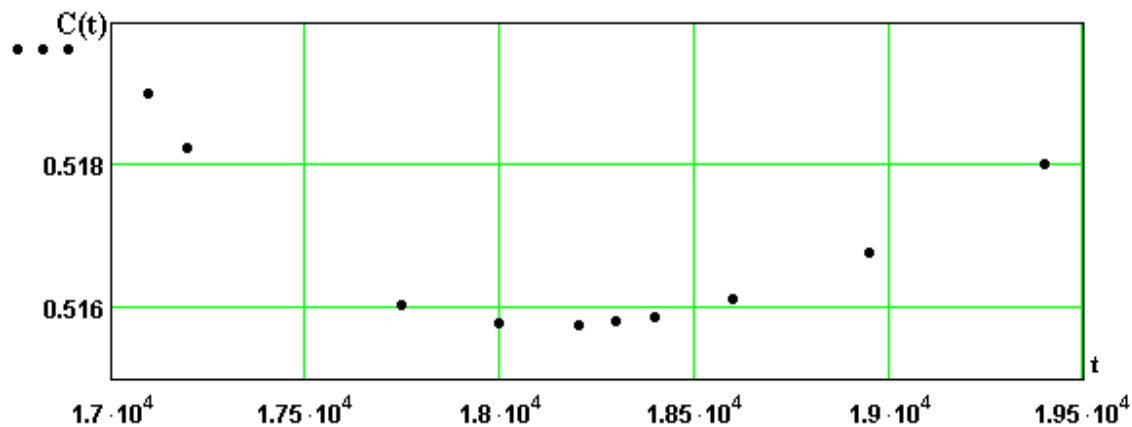


Рис. 3. Пример расчета зависимости $C(\tau)$ от периодичности проведения ТО АТ

Fig. 3. Example of calculation based on $C(\tau)$ from the periodicity of the technical support of automotive vehicles

И частные, и обобщенный показатели эффективности функционирования зависят от периода проведения ТО. В итоге необходимо определить оптимальную периодичность проведения ТО АТ $\tau_{\text{опт}}$, при которой обеспечивается заданное значение показателя готовности АТ при минимально возможных затратах на ее эксплуатацию, т.е.:

$$\text{ОПЭФ}(\tau) \longrightarrow \min_{\tau_{\text{опт}} \in [0; T_{\text{пр}}]}$$

Критериями оценки эффективности функционирования АТ, как и для других технических устройств, могут служить понятия пригодности, оптимальности и превосходства [Петухов Г.Б., 1989]. Используя введенный ОПЭФ (7) и задавая требуемые значения $K_{\text{пр}}$, можно определить интервал времени, на котором будет выполняться заданные требования по надежности, то есть сформировать т.н. область пригодности. В этой области, в свою очередь, могут быть выделены область оптимальности (в которой один из ПЭФ достигает экстремального значения при соблюдении ограничений и условий на другие показатели эффективности), а также область превосходства (в которой достигается минимум отклонения ПЭФ от своих экстремумов, т.е. экстремальное значение ОПЭФ).

Изложенный материал иллюстрируется рис. 4.

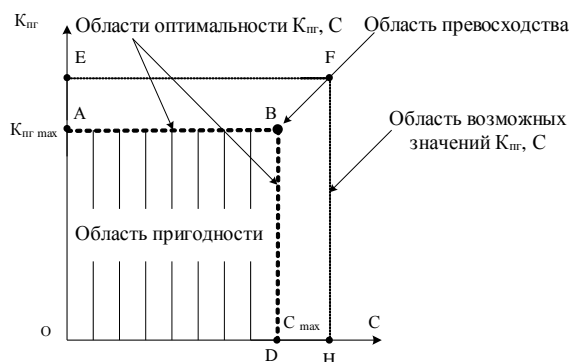


Рис. 4. Области пригодности, оптимальности и превосходства ЧПЭФ

Fig. 4. The field of fitness, optimality and superiority of the partial indicators of efficiency of functioning

На рисунке 4 в общем виде показаны: в виде фигуры OEFH – область возможных значений $K_{\text{пр}}$ и C ; в виде фигуры OABD – область пригодности; отрезками АВ, BD – области оптимальности показателей $K_{\text{пр}}$ и C соответственно; т. В – область превосходства.

При записи ОПЭФ в виде (7) подразумевается, что требования по надежности для АТ имеют более важное значение, чем требования по минимизации затрат на ее эксплуатацию. Однако возможны ситуации, при которых требования по обеспечению надежности и экономичности имеют одинаковый приоритет. В

этом случае ОПЭФ может быть сформирован в другом виде. Так, свертка методом идеальной точки [Петухов Г.Б., 1989] первого и второго ПЭФ – $K_{пр}$ и C – в ОПЭФ позволяет получить зависимость вида:

$$ОПЭФ(\tau) = \sqrt{(1 - K_{пр}(\tau))^2 + C(\tau)^2}.$$

Как и в предыдущем случае, перебор всех возможных значений τ позволяет определить момент начала ТО, доставляющий минимум этому выражению, т.е. приводящий к компромиссу между показателем готовности АТ и средними непроизводительными затратами на ее эксплуатацию. В этом случае периодичность проведения ТО АТ $\tau_{опт}$, при которой достигается минимум ОПЭФ – минимум отклонения $K_{пр}$ и C от своих оптимумов, т.е. обеспечивается компромисс между надежностью ($K_{пр}$) и стоимостью эксплуатации АТ (C), также можно считать оптимальной. Это оптимальное значение периодичности проведения ТО, доставляющее минимум ОПЭФ, также может быть определено методом перебора с использованием ЭВМ (рис. 5). При этом очевидно, что возможности как современной вычислительной техники, так и специализированного программного обеспечения позволяют решить эту задачу в реальном масштабе времени [Парфенов А.В., Чудинов С.М. 2016; Петриченко Г.С., Петриченко В.Г. 2016].

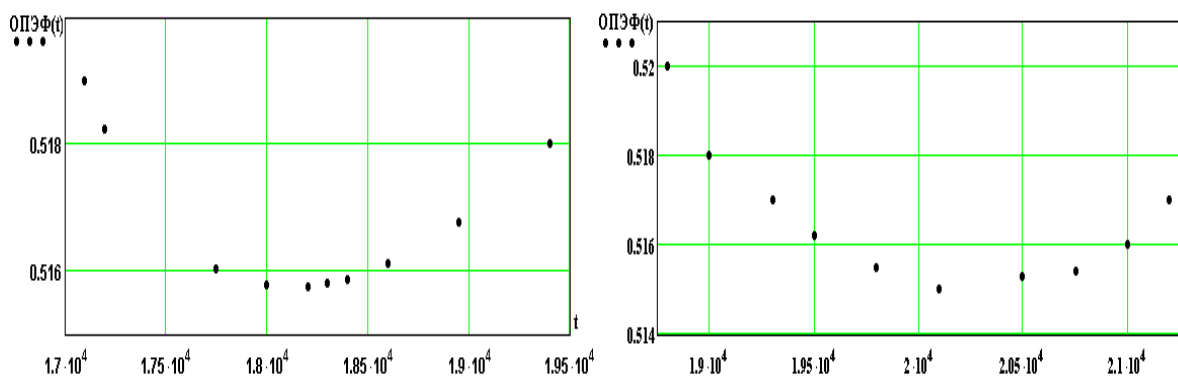


Рис. 5. Пример расчета зависимости ОПЭФ АТ от периодичности проведения ее ТО

Fig. 5. Example of calculation based on generalized performance indicators for automotive vehicles from the periodicity of the maintenance

Выражения $P_0(\tau)$ и $P_{отк}(\tau)$ (вероятность безотказной работы и вероятность отказа АТ на интервале τ) в формулах (2 – 6) и связанные с ними аналитические зависимости для плотности распределения времени до отказа АТ $\omega(\tau)$ могут быть найдены путем вероятностного прогнозирования момента выхода определяющего параметра АТ за допустимые пределы (например, при помощи подхода, изложенного в работе [Шляпцев С. Н., Ходжаев И. А., Королёв М. В., 2005]).

Таким образом, проанализированные в данной работе определяющие параметры, выбранные для АТ, могут быть различными в зависимости от конкретных (особых) условий ее эксплуатации. Тем не менее, по каждому из данных параметров при помощи прогнозирования могут быть получены вероятностные показатели надежности АТ, а на их основе сформированы показатели эффективности ее функционирования, на базе которых возможна дальнейшая организация гибкой системы ТО АТ с оптимальной (с точки зрения введенных показателей) периодичностью.

Список литературы

1. Хасанов Р.Х. Основы технической эксплуатации автомобилей: Учебное пособие. – Оренбург: ГОУ ОГУ, 2003. – 193 с.
2. Агеев Е.В. Особые условия технической эксплуатации и экологическая безопасность автомобилей: Учебное пособие. – Курск: ЮЗГУ, 2015. – 212 с.
3. Ефремов Л.В. Проблемы управления надежно-ориентированной технической эксплуатацией машин. – СПб: Art-Xpress, 2015. – 206 с.
4. Ощепков П.П. Оценка влияния надежности автомобиля «Камаз» на безопасность дорожного движения в условиях Севера. Диссертация на соискание степени к.т.н. Якутск: ЯГУ им. М. М. Аммосова. – 2000. – 147 с.

5. Ионов В.В. Исследование эксплуатационной надежности агрегатов трансмиссии автомобилей «Камаз» // Вестник СВГУ. – 2013. – № 20. – С. 82.
6. Надежность и эффективность в технике: Справочник в 10 т./ Ред. совет: В.С. Авдеевский и др. – М: Машиностроение, 1986. – 224 с.
7. Зеленцов В.А. Гибкие стратегии ТО – проблемы внедрения и пути их решения // Стандарты и качество. – 1991. – № 12. – С.35-38.
8. Ситчихина М.В. Разработка моделей и программных средств прогнозирования остаточного ресурса оборудования. Диссертация на соискание степени к.т.н. Иркутск: Байкальский университет экономики и права. – 2003.
9. Техническая эксплуатация автомобилей: Учебник. – М: Транспорт, 1991. – 413 с.
10. Бабенко И.А. Внедрение системы ТО по фактическому состоянию машинного парка. http://thepit.oaosng.ru/babenko_r.htm (электронный ресурс).
11. Барзилович Е.Ю. Модели технического обслуживания сложных систем. – М: Высшая школа, 1982. – 238 с.
12. Смирнов Н.Н, Ицкевич А.А. Обслуживание и ремонт авиационной техники по состоянию. – М: Транспорт, 1980. – 232 с.
13. Петухов Г.Б. Основы теории эффективности целенаправленных процессов. Часть 1. Методология, методы, модели. – Л: МО СССР, 1989. – 660 с.
14. Шляпцев С.Н., Ходжаев И.А., Королев М.В. О возможности индивидуального прогнозирования показателей безотказности систем виброакустической маскировки // Техника и технология. – 2005. – № 6 (12). – С. 66-68.
15. Парфенов А.В., Чудинов С.М. Тенденции развития вычислительной техники // Научные ведомости Белгородского государственного университета. – 2016. – № 16(237). – С. 98 – 106.
16. Петриченко Г.С., Петриченко В.Г. Оценка эффективности программного обеспечения // Научные ведомости Белгородского государственного университета. – 2016. - № 9 (230). – С. 108 – 112.
17. Батьковский А.М., Батьковский М.А., Ройкоз Г.А., Чудинов С.М. Методы оптимизации жизненного цикла разработки радиоэлектронной продукции // Научные ведомости Белгородского государственного университета. – 2014. - № 15 (186). – С. 121 – 127.
18. Черноморец А.А., Болгова Е.В., Черноморец Д.А., Коваленко А.Н. Метод прогнозирования на основе частотных представлений // Научные ведомости Белгородского государственного университета. – 2015. - № 13 (210). – С. 164 – 169.

References

1. Khasanov R.H. fundamentals of technical exploitation of cars: a tutorial. – Orenburg: GOU OGU, 2003. – 193 p. (in Russian)
2. Ageev E.V. Special conditions of technical operation and environmental safety of vehicles: textbook. – Kursk: SWSU, 2015. – 212 p. (in Russian)
3. Efremov L.V. Problems of management nadejnosti-oriented technical operation of machines. – St. Petersburg: Art Express 2015. – 206 p. (in Russian)
4. Oshchepkov P.P. Assessment of the impact of the reliability of the vehicles for road safety in the North. The dissertation on competition of a scientific degree of candidate of technical Sciences, Yakutsk: YSU them. M. M. Ammosov. – 2000. – 147 p. (in Russian)
5. Ions V.V. Research of operational reliability of the transmission units of vehicles // journal of northeastern state University. – 2013. - No. 20. – P. 82. (in Russian)
6. Reliability and efficiency in engineering: Guide in 10 volumes/ Ed. tip: V.S. Avduevskii and others – М.: Mashinostroenie, 1986. – 224 p. (in Russian)
7. Zelentsov V.A. Flexible strategy – implementation problems and ways of their solution // Standards and quality. – 1991. – No. 12. – P. 35-38. (in Russian)
8. Sitchikhin M.V. Development of models and software for forecasting the residual resource of the equipment. Thesis for the degree of Ph. D. Irkutsk: Baikal University of Economics and law. – 2003. (in Russian)
9. Maintenance of vehicles: the Textbook. – М: Transport, 1991. – 413 p. (in Russian)
10. Babenko A.I. the implementation of the system according to the actual condition of the machinery. http://thepit.oaosng.ru/babenko_r.htm (in Russian)
11. Barzilovich E.Y. Models of maintenance of complex systems. – М: Higher school, 1982. – 238 p(in Russian)
12. Smirnov N.N, Itskovich AA. Maintenance and repair of aircraft as of. – М: Transport, 1980. – 232 p. (in Russian)
13. Petuchov, G. B., Foundations of the theory of the effectiveness of targeted processes. Part 1. Methodology, methods, models. – LENINGRAD: USSR MINISTRY OF DEFENSE, 1989. – 660 p. (in Russian)
14. Slepcev S.N., Khodzhaev I.A., Korolev M.V. On the possibility of individual prediction of reliability of systems of vibroacoustic masking // engineering and technology. – 2005. – № 6(12). – S. 66-68. (in Russian)
15. Parfenov A.V., Chudinov S.M. Technology trends of computer engineering // Belgorod State University Scientific Bulletin. – 2016. - № 16(237). – P. 98 – 106. (in Russian)

16. Petrichenko G.S., Petrichenko V.G. Performance evaluation software // Belgorod State University Scientific Bulletin. – 2016. - № 9 (230). – P. 108 – 112. (in Russian)
17. Batkovskij A.M., Batkovskij M.A., Rojkoz G.A., Chudinov S.M. The optimization techniques of radio equipment lifetime // Belgorod State University Scientific Bulletin. – 2014. - № 15 (186). – P. 121 – 127. (in Russian)
18. Chernomorets A.A., Bolgova E.V., Chernomorets D.A., Kovalenko A.N. The prediction method of frequency representation // Belgorod State University Scientific Bulletin. – 2015. - № 13 (210). – P. 164 – 169. (in Russian)

Савин Леонид Олегович, сотрудник Академии Федеральной службы охраны Российской Федерации

Королёв Михаил Викторович, кандидат технических наук, сотрудник Академии Федеральной службы охраны Российской Федерации

Носов Максим Васильевич, кандидат технических наук, сотрудник Академии Федеральной службы охраны Российской Федерации

Savin Leonid Olegovich, Academy of the Federal security service of the Russian Federation

Korolev Mikhail Viktorovich, Candidate of Technical Sciences, Academy of the Federal security service of the Russian Federation

Nosov Maksim Vasilyevich, Candidate of Technical Sciences, Academy of the Federal security service of the Russian Federation

УДК 004.434

DOI: 10.18413/2518-1092-2017-2-2-21-33

Седых А.
Рязанов Ю.Д.

**СКРИПТОВЫЙ ЯЗЫК «LINK» ДЛЯ ПРОГРАММИРОВАНИЯ
СЦЕНАРИЕВ ИСПОЛНЯЕМЫХ СОБЫТИЙНЫХ СИСТЕМ**

Белгородский государственный технологический университет им. В.Г. Шухова,
ул. Костюкова, 46, г. Белгород 308012, Россия

e-mail: chronoexp@gmail.com, ryazanov.iurij@yandex.ru

Аннотация

Языки описания сценариев (скриптовые языки) – высокоуровневые языки программирования, которые интерпретируются некоторой программой во время её выполнения. Скриптовые языки предоставляют довольно привлекательные возможности, обладают более сложным инструментарием и поддерживают более прогрессивные техники программирования, чем, например, компилируемые языки. Они, как правило, позволяют простым и компактным способом описывать управление большим количеством объектов предметной области. Однако многие языки базируются на некоторых шаблонных подходах, которые могут создавать проблемы во время использования и породить множество ошибок. В статье представлен разработанный авторами специализированный язык Link для описания сценариев асинхронных событийных систем. В языке Link решены такие известные проблемы языков программирования, как использование значения null, тесная взаимосвязь данных с кодом, большая сложность разработки и чтения кода, глобальные области видимости переменных. Некоторые его возможности, такие как ссылки, разделение функций на типы, операции применения функций, классы функций, частичная настройка и наследование функций обеспечивают отличающийся от большинства языков подход в разработке сценариев исполняемых событийных систем.

Ключевые слова: скриптовые языки программирования; описание сценариев; асинхронные событийные системы; теория языков программирования.

UDK 004.434

Sedykh A.
Ryazanov Y.D.

**SCRIPTING PROGRAMMING LANGUAGE “LINK” FOR RUNNABLE
EVENT-DRIVEN SYSTEMS**

Belgorod State Technological University named after V.G. Shoukhov, 46 Kostyukova St., Belgorod, 308012, Russia

e-mail: chronoexp@gmail.com, ryazanov.iurij@yandex.ru

Abstract

Script description languages (scripting languages) are high-level programming languages that are interpreted by some program during its execution. Script languages provide quite attractive features, has more complex tools and supports more advanced programming techniques, in comparison to compiled languages. Scripting languages allow a simple and compact way to describe the control flow of a large number of objects in the subject area. However, many languages are based on some template approaches that can create problems during use and cause a lot of errors. The article presents a specialized language developed by the authors for describing scenarios of asynchronous event-driven systems. The Link language solved such well-known problems of programming languages as the use of null, close correlation of data with the code, great difficulty in developing and reading code, global scope of variables. Some of its features, such as links, the separation of functions into types, operation of application functions, function classes, partial tuning and inheritance of functions offer an approach different from most languages in the development of scenarios of runnable event-driven systems.

Keywords: scripting programming languages; script composition; asynchronous event-driven systems; theory of programming languages.

Введение

На текущее время существует множество языков программирования, созданных под самые различные нужды. Помимо универсальных языков программирования, таких как С, Java и С# также разрабатываются специфичные определенной среде выполнения сценарные языки, такие как sh, Perl, Elixir и Erlang [1]. Необходимость в создании нового языка или улучшении существующего обуславливается как появлением новых окружающих условий или систем для программирования, так и возникновением идей для усовершенствования существующей концепции написания программ на определенном языке. К примеру, недавно был разработан новый проблемно-ориентированный язык для удаленного программирования контроллеров, в котором была применена идея удаленного программирования с помощью модуля GSM [2].

Языки описания сценариев (скриптовые языки) – высокоуровневые языки программирования, которые интерпретируются некоторой программой во время её выполнения [3]. Это отличает их от компилируемых языков, таких как С, С++ или Go. В общем плане, языки подобного рода предназначены для более удобного управления объектами в сценариях какой-то уже существующей системы [4]. Создание новых и усовершенствование существующих скриптовых языков также является актуальной темой для проведения научной работы [5].

Языки описания сценариев предоставляют довольно привлекательные возможности, обладают более сложным инструментарием и поддерживают более прогрессивные техники программирования. Они, как правило, помогают простым и компактным способом обрабатывать и управлять большим количеством объектов в применяемой предметной области. Однако многие языки базируются на некоторых шаблонных подходах, которые могут создавать проблемы во время использования и породить множество ошибок. Но существуют языки программирования, в которых удалось избавиться от некоторых недостатков такого рода.

Целью данной работы является разработка нового скриптового языка для управления множеством объектов асинхронных событийных систем различной структуры и поведения, позволяющего упростить программирование, чтение и отладку сценариев, а также минимизировать возникающие в процессе программирования проблемы.

В первой части статьи приведены некоторые существующие актуальные проблемы современных языков программирования. Описываются различные способы решения каждой из проблем в различных языках программирования на данный момент. Во второй части статьи приводится неформальное описание принципа работы и особенностей разработанного авторами скриптового языка программирования, который включает в себя композицию хорошо зарекомендовавших себя и совершенно новых методов решения описанных проблем.

1. НЕКОТОРЫЕ ПРОБЛЕМЫ СОВРЕМЕННЫХ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

Среди наиболее известных проблем популярных языков программирования присутствуют такие как наличие неопределенности пустоты переменной (использование значения null), тесная взаимосвязь данных с кодом в императивных языках программирования, большая сложность разработки и чтения кода. В сценарной среде исполнения поднимается вопрос расположения точки входа и привязки необходимого кода к определенной ситуации в системе [6]. Базовым решением, например, как в языке JavaScript, является предоставление объекту или прототипу требуемой функции обратного вызова. Однако реальные системы нередко требуют более детального описания ситуации, при которой нужно запускать код. Ещё одной серьёзной проблемой является отведение специального значения для обозначения отсутствия чего-либо – так называемое null-значение. Возможность каждой переменной иметь ссылку не только на реально существующий объект, но и на отсутствие этого объекта значительно усложняет процесс разработки и отлаживания кода. Рассмотрим эти и другие проблемы более подробно.

1.1 Глобальная видимость объектов, замыкания и callback hell

В скриптовых языках зачастую в функциях обратного вызова требуется доступ к внешнему окружению объекта, что реализуется системой замыканий. В результате может возникнуть избыточная загруженность большим количеством переменных и замыканий, разобраться в которой затруднительно. В проектах с небольшим количеством разработчиков это может не являться сильной проблемой, но в крупных корпоративных разработках такая сложность разработки превращается в ад. Так и зародился термин «callback hell» – «ад коллбэков», который часто применяют, критикуя JavaScript [7].

Чем меньше будет область видимости функции, тем проще её реализовывать и поддерживать. Нет необходимости нагружать текущий контекст всем вышестоящим содержимым, как это сделано в случае замыканий в JavaScript. Примером идеального в этом плане языка можно привести Haskell – в этом языке отсутствуют глобальные области видимости переменных. Функция зависит только от переданных ей параметров, и вся разработка сводится к описанию зависимостей выходных данных от входных. Такие функции называются чистыми, их проще разрабатывать, отлаживать и поддерживать [8]. Как правило, код написанный на Haskell, работает правильно с первой же компиляции. Во многом это заслуга именно отсутствия глобальных переменных и различных замыканий. Такой код хорошо структурируется и его легче «разложить по полочкам». В некоторых языках программирования разработчики пытаются хоть и не избавиться, но наиболее безопасно использовать замыкания. Например, в php для передачи определенных переменных в контекст вложенной функции используется служебное слово use. Таким образом, внутри функции доступны только переменные, которые туда нужно передать. Тем самым фильтруется набор доступной изнутри функции информации. Такой подход уменьшает сложность написания и поддержки программ с использованием замыканий по сравнению с JavaScript [9].

1.2 Ошибка несоответствия типов

В динамических языках программирования вроде Python ошибка несоответствия типов переменных во время выполнения – довольно частое явление. Возможность не указывать тип данных в совокупности с тем, что тип данных конкретной переменной может меняться со временем стала одной из основных проблем динамических языков. Вместе с этой возможностью писать код стало легче, однако чтение и отладка того же кода представляет собой довольно сложный процесс [10]. Более свободная и менее ограничивающая динамическая типизация накладывает на разработчика более строгие требования по документированию кода. Чтобы читатели кода не догадывались, как могут повести себя функции, что они принимают и что возвращают, разработчик должен всегда следить за актуальностью информации, которая в статических языках частично прошита на уровне системы типов и автоматически обновляется при рефакторинге [11]. Использование же строго типизированного языка сильно увеличивает объем кода и порой ограничивает разработчика в гибкости его кода.

В качестве компромисса возникает идея создания скриптового языка с универсальной динамической типизацией, в котором все значения будут обрабатываться одним и тем же образом, без возможности изменить значение идентификатора. Исходя из этого, можно одновременно использовать удобство автоматического вывода типов и не испытывать проблем с возникновением ошибок разной типизации, ведь большинство значений будет являться одним типом и при необходимости будет преобразовываться в нужный применяемый тип.

1.3 Ошибка пустого указателя (NullPointerException)

В марте 2009-го года сэр Тони Хоар выступил на конференции Qcon в Лондоне с докладом на тему "Нулевые ссылки: ошибка на миллиард долларов" (Null References: The Billion Dollar Mistake), в котором признался, что считает изобретение нулевых указателей одной из главных своих ошибок, стоившей индустрии миллиарды долларов [12]. Попытка доступа к элементу объекта, который является нулевым указателем (null), вызывает ошибку времени выполнения. В языке Java это называется NullPointerException (NPE). В языке JavaScript присутствуют целых два пустых состояния – null и undefined [13]. Наличие таких специальных значений в языках программирования очень сильно увеличивает число потенциальных ошибок. В результате программистам требуется проверять переменные на наличие требуемых объектов каждый раз перед совершением какого-либо действия. На данный момент существует масса способов избежать такой ошибки [14]. Некоторые способы, вроде повсеместного указания условных операторов слишком нагружают код, прямолинейны и требуют дополнительных вычислительных действий. Другие, как специальные атрибуты NotNull, Nullable, Optional, метод orNull и операторы coalesce, Элвиса и тернарные – описывают соглашения по спецификации конкретных переменных, а также их использование [15]. Среди подобных решений основная проблема – унаследованный код и сторонние библиотеки, которые по-прежнему имеют дело с нулевыми указателями. Также, ввиду своей необязательности, они позволяют разработчику создавать как безопасный, так и небезопасный в плане ошибки NPE код. Такая двойственность заставляет разработчиков по-прежнему заботиться по поводу ошибки нулевого указателя, хоть и в более мягкой форме. В языке Haskell для обработки результата функции, которая может вернуть пустое значение используется специальная монада Maybe [16]. Остальные же функции *обязаны* вернуть

значимый результат. Также одним из вариантов решения этой проблемы является разработка нового языка, без установленного изъяна в процессе проектирования. К примеру, в языке описания шаблонов Freemarker было решено не включать null в поддерживаемые значения [17]. Однако, в процессе взаимодействия с другими языками и системами, работающими на других языках – по-прежнему остается вероятность получения подобной ошибки. Если в генератор файлов языка Freemarker передать значение null, то возникнет ошибка во время использования шаблона. Требуется спроектировать такой язык, который не будет давать возможности возникновения таких ошибок. Само по себе отсутствие чего-либо не должно вызывать никаких исключительных ситуаций. В большинстве случаев, если в программе встречается значение отсутствия объекта, дальнейший код просто не должен выполняться. Таким образом, даже в результате взаимодействия с системами на других языках, ошибка NPE будет отсутствовать, потому что интерпретатор самостоятельно позаботится об остановке выполнения инструкции в нужный момент.

2. ПРИНЦИП РАБОТЫ И ОСНОВНЫЕ ОСОБЕННОСТИ СКРИПТОВОГО ЯЗЫКА LINK

Разработанный авторами язык **Link** – интерпретируемый внедряемый скриптовый язык программирования. Он исполняется в среде виртуальной машины JVM во время выполнения программы, написанной на языке Java [18]. Внутри языка полностью отсутствует хранимое состояние переменных – все данные считываются и записываются в интерпретирующую его программу, которая использует систему идентификации объектов языка Link. Программы, использующие Link должны выполняться с поддержкой Link-машины, которая предоставляет каркас для создания и управления всеми объектами системы. Link-машина содержит в себе подсистемы управления событиями и объектами, а также – подсистему интерпретатора, позволяющую исполнять код на этом языке.

В рамках языка введено понятие «типа» функции. Функции могут быть различных типов, где каждый тип означает то, как эта функция будет обрабатываться в системе. Программа на языке Link представляет собой набор функций различных типов. Точку входа определяют функции типа «линк» – они связываются с определенными условиями и событиями, произошедшими в событийной и объектной подсистемах. Таким образом, в Link-машине присутствует собственная подсистема мониторинга, которая контролирует все события и запускает по мере необходимости нужные линки.

Основной упор при разработке языка сделан на удобство и скорость программирования для создания высокоуровневых функций обработки и управления большими наборами объектов различных сущностей системы. Скрипт общается с внешним миром только с помощью своей машины и вызова «нативных» функций на пользовательской платформе. Нативными функциями называются такие функции, которые были реализованы в основной системе [19]. Таким образом, Link-машина встраивается в исполняемое на JVM приложение и вызывает любые методы классов, которые были зарегистрированы в нём.

Подсистема объектов представляет собой гибкий контейнер для хранения и поиска по всем зарегистрированным объектам. Объект в подсистеме обладает своим уникальным идентификатором, типом, а также атрибутами-классами. В любое время можно получить все необходимые объекты с помощью строки-селектора объектов на языке Link. Результатом действия селектора является список подошедших под условие объектов, с которыми можно производить любые доступные действия. Принцип работы селектора напоминает функцию \$ в библиотеке jQuery для JavaScript [20].

Далее опишем основные особенности разработанного языка. Описание будет сопровождаться примерами кода на языке Link для решения задач из следующей предметной области. Разрабатывается моделируемая на 3D-движке демонстрационная интерактивная сцена офиса [21]. Каждый тип предмета или персонажа сцены в исполняемом приложении описан на языке Java как класс ООП и зарегистрирован в подсистеме объектов Link-машины. Для данной работы в офисе представлен следующий список классов: Guard – охранник, Document – документ, Computer – компьютер, Mouse – мышь, Chair – кресло. Также присутствует общий родитель для нескольких классов – LivingCreature – живое существо. Каждый объект этих классов имеет характерные для объекта свойства и методы поведения. Поставлена задача реализовать различные сценарии поведения некоторых типов объектов в зависимости от происходящих в офисе действий. Перейдем к описанию выделяющихся возможностей языка Link.

2.1 Линки

Линк – специальный тип функции, представляющий собой точку входа для сценария. Линк позволяет «линковать» – привязывать сценарий к определенным объектам, событиям, времени ожидания и

дополнительным произвольным условиям. Линк состоит из секции описания ситуации вызова сценария, и, собственно, самого сценария. Пример:

```
// после того как в офисе был прочитан документ, вывести в
// консоль имя этого документа и количество страниц
link documentProcessed
| object: [/Document/]^
| event: FinishedReading
| code: {
    print("Просмотрен документ " + object:name + ". Обработано страниц: " +
object:lines)
}
```

В представленном коде описана функция вывода в консоль имени документа и числа страниц документа, находящегося в офисе. Этот сценарий запускается всякий раз при совершении события `FinishedReading` объекта типа `Document`.

Приведем другой пример. Допустим, требуется написать код, который будет реализовывать следующий сценарий: если человек с классом охранника заходит в офис и наблюдает отсутствие всех компьютеров, то он через 5 секунд запускает сигнал тревоги. Для начала, попробуем написать этот сценарий на нативном для Link языке проекта – Java. Для реализации сценария добавим в класс охранника `Guard` метод `guardComputersStolen`:

```
class Guard extends NPC {
    // методы и свойства охранника
    // включая callOverallAlarm - запуск тревоги
    ...

    // необходимо вызывать в конструкторе, т.к. идет привязка
    // к событийной подсистеме
    void guardComputersStolen() {
        final Guard guard = this;
        EventManager.link(this, "EnteredOffice", new Callable {
            void exec(EventParams params) {
                try {
                    List<Computer> computers =ObjectMgr.findByClass("Computer");
                    if (computers.isEmpty()) {
                        Thread.Sleep(5000);
                        guard.callOverallAlarm();
                    }
                } catch (InterruptedException ie) { /**/ }
            }
        });
    }
}
```

При разработке сценариев на том же языке, на котором написано основное ПО, требуется прописывать все необходимые для работы сценария инструкции. В данном примере большую часть сценария занимает именно такой код: связывание функции обратного вызова с событием; присвоение статичной ссылки на требуемый объект охранника для использования замыкания; обработка исключения, которое может возникнуть при отсчете времени [22]; громоздкое обращение к объектной подсистеме для выбора списка компьютеров; проверка данного списка на пустоту – всё это значительно усложняет написание, чтение и отладку исходного кода. На первый взгляд простая задача описания сценария реакции объекта на событие превращается в достаточно сложную реализацию. Если в разрабатываемом проекте запланировано создать

множество таких или, как правило, более сложных сценариев – исходный код проекта будет разрастаться высоконагруженными цепочками кода, обрабатывающего сценарии с различным поведением объектов. Однако, большая часть этого кода будет заниматься лишь обслуживанием и связыванием самих сценариев, проверкой условий выполнения сценариев, в то время как сама логика действия будет занимать гораздо меньший объём.

Далее приводится реализация того же сценария с использованием языка Link:

```
link guardComputersStolen
| object: [/Guard/]^
| event: EnteredOffice
| condition: ![/Computer/]
| time: 5
| code: {
    object~Guard.callOverallAlarm$
}
```

Преимущество в краткости и лаконичности кода по сравнению с использованием нативного решения очевидно. Проведем детальный обзор происходящего в скриптовом коде. На второй строчке с пометкой `object` указывается объект, который выступает в роли инициатора события `event`. В данном случае это объект типа `Guard`. После того, как охранник заходит в офис, вызывается событие с именем `EnteredOffice` и проверяется условие, помеченное как `condition` – в нашем случае это обычный селектор всех объектов с типом `Computer` (подробнее механизм селектора описывается в пункте 2.2 этой статьи). Если данный селектор возвратит пустой список, отрицание пустого списка будет являться истинным выражением и сценарий запустится. Пометка `time` указывает количество секунд, которое требуется подождать перед началом выполнения сценария. Внутри секции `code` доступны все объекты линка, включая `object`, который будет обозначать инициатора данного события. Символ `~` обозначает операцию применения функции к левому операнду – охраннику, которая также будет описана позднее. Справа от этого символа указывается нативная функция `Guard.callOverallAlarm`, которая принадлежит исполняемой программе сцены офиса. Таким образом, разработчик скрипта знает, что в сцене у класса `Guard` реализован метод с названием `callOverallAlarm`, который позволяет охраннику поднимать тревогу.

2.2 Селектор объектов

Селектор объектов предоставляет специальный синтаксис для выборки объектов из подсистемы управления объектами. Оператор выборки является строкой, которая определяет условия для выборки. Условия могут быть самыми разнообразными: от классов, идентификаторов и типов объектов до выборки и фильтрации дочерних/родительских элементов. Схема использования селектора похожа на CSS-выборку в WEB – в строке указывается набор атрибутов объекта, по которым система выполняет поиск среди объектов исполняемой среды [23]. В селекторе языка Link можно использовать следующие атрибуты для выборки объектов:

1) Идентификатор. Идентификатор является уникальным значением для каждого объекта системы. В результате выборки по идентификатору селектор возвращает объект с заданным идентификатором, если он зарегистрирован в системе. Пример: `[/#Guard_32/]`.

2) Тип объекта. Типом объекта в Link-машине называется его класс в исполняемой среде JVM. При помещении контейнера объекта в подсистему объектов машины название класса регистрируется как атрибут типа для поиска и использования его в качестве селектора. Для запроса по типу объекта нужно просто указать в селекторе его название, к примеру - `[/Computer/]`.

3) Класс объекта. Каждый объект в подсистеме объектов Link-машины может иметь набор классов, его описывающих. Выборка по классам представляет собой фильтрацию всех доступных объектов на соответствие заданным классам, подобно CSS-селектору. К примеру, описываемый офис может содержать белые и черные стулья с мягкой и жесткой спинкой. Допустим, что эти свойства реализованы в подсистеме объектов как классы `black`, `white`, `soft` и `hard` у объектов типа `Chair`. Чтобы произвести выборку в селекторе по классу, необходимо поставить символ “.” перед названием класса. В нашем случае, для

выборки всех черных стульев с мягкой спинкой мы можем использовать селектор [/Chair .black .soft/].

4) Методы манипуляции с объектами иерархической структуры. При регистрации объекта в объектной подсистеме можно указать его родительский объект, который будет выступать в качестве контейнера новому объекту. Чтобы выбрать дочерние элементы контейнеров, нужно записать в селектор специальный символ '>', который возьмёт из текущих отфильтрованных объектов все дочерние объекты, и в дальнейшем фильтрация будет производиться именно по ним. Если контейнер не содержит дочерних объектов, он автоматически отклоняется. Для выборки родительских объектов существует специальный символ '<'. К примеру, в интерактивной сцене существуют компьютеры типа Computer с классом IBM, внутри каждого из которых могут быть зарегистрированы платы оперативной памяти RAM класса DIMM3. Для выборки всей доступной памяти этого класса среди компьютеров IBM можно использовать следующий селектор: [/Computer .IBM > RAM .DIMM3/].

Подобный синтаксис очень удобен для выборки целевых объектов сценариев. Система использования селекторов существенно облегчает написание и использование сценариев.

2.3 Вызов нативных методов

Связь языка Link с исполняемой на JVM программой реализуется с помощью объектной подсистемы. В этой подсистеме объекты Java содержатся внутри специальных контейнеров, называемых нативными. Нативный контейнер – это специальный объект, который содержит ссылку на настоящий объект Java, базовую информацию о хранимом объекте – его тип, персональный идентификатор в подсистеме, а также набор атрибутов, называемых классами. По этим параметрам и реализуется выборка значений с помощью селектора в объектной подсистеме.

В исходном коде Link разрешено вызывать нативные JVM методы, описанные в классах исходного проекта на языке Java. Для вызова нативного метода в сценарии достаточно сделать выборку контейнера требуемого объекта, и указать знак \$ в конце названия функции в случае применения метода над контейнером. Например, для выполнения нативного метода callOverallAlarm объекта с классом Guard достаточно написать [/Guard/~callOverallAlarm\$. Статические методы без привязки к конкретным объектам вызываются с помощью указания класса через символ точки: System.println\$("Привет!"). Применение нативных функций ничем не отличаются от применения функций самого языка Link. Благодаря рефлексии JVM, разработчику сценариев доступен весь набор классов и методов, который присутствует в исполняемом приложении [24]. Таким образом достигается прозрачность использования нативного кода приложения.

2.4 Расширенные операции, типы и их приведение

Автоматическая конвертация типов в случае использования литералов в Link позволяет в большинстве случаев не задумываться об используемых динамических типах. Все типы выводятся самостоятельно за счёт четкой логики вывода из исходного или нативного кода. Всего в языке существует 10 типов литералов:

Таблица 1

Литералы языка программирования Link

Table 1

Literals of Link programming language

№	Название	Пример	Описание
1	Integer	2, 4, 5, 42	Целое число
2	Float	3.14, -5.24, 0.001	Число с плавающей точкой
3	String	"Hello", ""	Строка из символов
4	Boolean	false, true	Логический тип
5	Identifier	a, parseValue	Идентификатор переменной
6	List	[2, 3, "123"], []	Набор значений, список
7	Map	{literal: true, name: "map"}, {}	Словарь

8	Selector	[/Chair .black .soft/], [//]	Селектор объектной подсистемы
9	Function	Guard.alarm\$, random	Функция
10	Container	[/Chair .black .soft/]^	Контейнер объектной подсистемы

Как можно заметить, функция в Link также является литералом. Функции здесь не исполняются, а применяются к левому операнду с помощью операций применения функций (см. пункт 2.8). Специальный тип Selector отличается от списка лишь формой записи и поведением – когда потребуется вычислить значение этого литерала, выполнится запрос к объектной подсистеме и результат преобразится в список контейнеров. Запись конкретного контейнера в коде невозможна, контейнер объекта можно получить только с помощью селектора или вызова нативного метода.

Помимо привычных операций над обычными литералами, таких как вычитание и умножение чисел, в язык встроены дополнительные полезные операции над списками и объектами. Также расширен доступный набор действий при операциях со строками. К примеру, арифметические операции над списками и словарями реализуют базовые алгоритмы дискретной математики над упорядоченными множествами – + => объединение, - => разница, * => пересечение и / => исключаящая разница [25]. Для списков добавлены операции взятия головы и хвоста, для словарей – выборка значения [26]. Всё это существенно облегчает работу по выборке, отсеиванию, выделению нужных параметров и изменению объектов сценария, ведь результатом работы селектора является список контейнеров.

Среди типов и операций отсутствует значение null. Необходимость в null исключается за счёт того, что результатом запросов селектора или вызова нативной функции всегда будет являться список. С точки зрения программиста видимо будет лишь одно значение – список объектов, который может быть пустой. Пустой же список объектов будет значить, что требуемые объекты не были найдены. В этом случае код над пустым списком не будет выполняться. Это позволит максимально упростить написание и отладку кода, достигнув логической простоты сценарных функций. Благодаря этому нововведению, при написании сценария можно будет не указывать повсеместно проверки на null или на пустоту списка там, где это на самом деле не требуется.

2.5 Встроенные типы функций

Все функции в языке Link классифицированы по типу функции. Для функций определенных типов применяется специальный синтаксис. На данный момент существуют такие типы как map, link, filter, reduce, transform. Некоторые типы функций созданы как безопасная замена циклов в императивных языках программирования. Идеология этих типов взята из основных функций высшего порядка в функциональных языках программирования, где они с успехом применяются [27]. Предопределенные ссылки идентификаторов вроде x, y, z – уже встроены в язык в пределах функции. Эти ссылки позволяют оперировать с входными и выходными значениями. Применение функции к объекту будет значить проход по одному элементу, если это один элемент, проход по списку – если это список, и отсутствие действий в случае пустого списка. В следующей таблице отражены все существующие типы функций.

Таблица 2

Типы функций языка программирования Link

Table 2

Function types of Link programming language

Название	Описание	Синтаксис
map	Отображает один набор данных в другой. x – элемент входного списка y – элемент выходного списка	map <name> params: <arguments> code: { <code> }
link	Связывает ситуацию в системе с исполняемым кодом сценария. object – связуемый объект	link <name> object: <selector> event: <name>

	event – связуемое событие time – связуемое время condition – связуемое условие	time: <time> condition: <expression> code: { <code> }
filter	Фильтрует входной список. x – элемент входного списка y – логическое значение	filter <name> params: <arguments> code: { <code> }
reduce	Склеивает элементы списка в 1 элемент. x – входной элемент y – аккумулятор склеивания z – выходной элемент initial – начальное значение аккумулятора	reduce <name> params: <arguments> initial: <expression> code: { <code> }
function	Стандартная функция в обычном понимании. xs – входной элемент ys – выходной элемент	function <name> params: <arguments> code: { <code> }
transform	Последовательное применение функции к контексту, пока не будет достигнуто условие. x – входной элемент until – условие завершения code – код преобразования	transform <name> params: <arguments> until: <expression> code: { <code> }

2.6 Параметры функций и их наследование

Любая функция в Link состоит из параметров. Обязательными параметрами являются имя и тип функции. Обычно функции содержат и другие параметры: код и список аргументов. В случае нестандартных типов функций – например, линков – условия привязки. Идея разбиения функции на параметры заключается в том, что эти параметры можно будет указывать частично – например, указать для функции только аргументы, а реализацию описать в другом месте. В совокупности с возможностью наследования функций это даёт гибкую возможность описать шаблон функции, а позже лишь менять базовую часть. Далее описывается новая уникальная возможность языка – настройка функций.

Настройка функции – возможность описать все необходимые для работы функции параметры. По своему принципу работы она напоминает каррирование в языке Haskell, – функция агрегируется со своими аргументами по очереди, и лишь когда все параметры собраны функция может быть вычислена [28]. Однако, в Link функции собирают свои параметры, а не аргументы. Поэтому агрегировать их можно совершенно в любом порядке.

Наследование функций – возможность представленного языка повторно использовать один и тот же код. В совокупности с частичной настройкой, предоставляет разработчику гибкую возможность расширять уже существующий код. Подобно ООП, в наследовании функции участвуют два объекта: наследник и родитель [29]. Для наследования существует ограничение по типу функции – функция-наследник должна быть такого же типа, как и функция-родитель. В отличие от наследования классов, наследовать функции безопаснее, ведь отсутствует изменяемое состояние в рамках наследования.

Сущность наследования функции в том, что функция-наследник перенимает у функции-родителя все её параметры. Также, в теле наследника можно использовать специальную функцию parent, которая означает вызов функции родителя с указанными аргументами. Наследование функции, как и наследование классов в Java, записывается с помощью ключевого слова extends, после чего следует название функции-родителя.

2.7 Операции применения функций

В объектно-ориентированных языках программирования встречается следующий подход вызова методов у объектов:

```
Объект.метод1(аргументы);  
Объект.метод2(аргументы);
```

В языке Link было решено пойти другим путём. Выделяются две операции применения функций: операция синхронного применения и асинхронного. Операция применения функций работает следующим образом: она помечает левый операнд как контекст, связывает его с функцией – правым операндом, и вызывает эту функцию выбранным способом с поддержкой связанного контекста. Контекст доступен внутри функции обычно по ссылке `x` или `xs`. После выполнения функции, контекст является изменённым, и он меняется каждый раз, когда далее по коду происходит ещё применение функций над этим контекстом. Таким образом получаются цепочки применения функций на контексте.

Объект~функция1(аргументы)~функция2(аргументы);

Стоит выделить, что те же самые методы можно применить асинхронно, если они не связаны друг с другом:

Объект|функция1(аргументы)|функция2(аргументы);

Таким образом, применение функций становится частью выражения. Подобный синтаксис позволяет максимально прозрачно для разработчика использовать как синхронные, так и асинхронные вызовы [30]. Во время разработки сценария очень важно, чтобы сценарист не задумывался об организационных вопросах вроде распараллеливания подзадач, общих переменных и синхронизации. Это всё должно происходить автоматически, оставив для управления исполняемым потоком выбор из двух вариантов применения конкретной функции.

2.8 Классы функций

Классы функций являются уникальной возможностью привязывать дополнительный шаблонный код к функциям. Классом функции называется определенный именованный блок кода, который расположен вне всякой функции. Этим классом можно пометить любую функцию, тогда каждый раз при вызове функции будет выполняться указанный в блоке класса код. На данный момент существует три типа классов, которые отличаются временем выполнения: класс `before` запускается до выполнения основного кода привязанной функции, класс `after` – после выполнения. Третий тип класса `wrap` предназначен для оборачивания подписываемых функций в дополнительный код. Для класса `wrap` существует ключевое слово `execute`, которое производит выполнение основного кода функции. Приведем простой пример с использованием `wrap`:

```
// класс логирования в консоль любых исполняемых линков
// вместе с данными объекта и события
class wrap loggingLink {
  consoleLog("Произошел вызов линка " + this:name + "!")
  consoleLog("Данные объекта: " + object)
  consoleLog("Данные события: " + event)
  execute
  consoleLog("Закончился линк " + this:name)
}

// опишем несколько линков

// каждое существо при входе в комнату начинает искать своих друзей
link creatureFindFriendsOnEntering using loggingLink
| object: [/LivingCreature/] ^
| event: EnteredRoom
| code: {
  object~findFriends$
}

// при выходе одного из существ из комнаты все остальные существа
производят прыжок
link allCreaturesJumpOnLeaving using loggingLink
| object [/LivingCreature/] ^
| event LeavingRoom
| code {
```

```
// все кроме самого объекта совершают прыжок
([/LivingCreature/] - [object])~jump$
}

// и так далее...
```

Данный функционал напоминает аспектно-ориентированное программирование, где подобно классам функции можно описать общий логический слой для разных мест в исходном коде [31]. Например, при разработке финансовых операции мы можем определить класс транзакционных функций, который будет оборачивать функции с этим классом в транзакции. Также в Link разрешено комбинировать несколько классов для функции. Таким образом можно организовать многоуровневую логику для определенного вида функций.

ЗАКЛЮЧЕНИЕ

Разработанный скриптовый язык программирования Link предназначен для описания сценариев в системах с большим количеством разнотипных объектов, пользовательских и сторонних событий, требующих гибкое и удобное решение внедрения сценариев. Он подходит любым приложениям на любой платформе, так как Link является универсальным внедряемым языком программирования, и выполняется в мультиплатформенной исполняемой среде Java Virtual Machine [32]. В языке Link решены такие известные проблемы языков программирования, как использование значения null, тесная взаимосвязь данных с кодом, большая сложность разработки и чтения кода, глобальные области видимости переменных. Некоторые его возможности, такие как линки разделение функций на типы, операции применения функций, классы функций, частичная настройка и наследование функций обеспечивают отличающийся от большинства языков подход в разработке сценариев исполняемых событийных систем.

Дальнейшим расширением языка может быть введение модульности и системы экспорта и импорта функций между модулями, введение возможности выделить действующее лицо сценария и других возможностей, направленных на упрощение разработки сценариев.

Список литературы

1. Computer Languages History. URL: <https://www.levenez.com/lang/> (дата обращения: 04.05.2017).
2. Муромцев В.В., Ломакин В.В., Цоцорина Н.В. Разработка специализированного языка для удаленного программирования микроконтроллеров // Научные ведомости Белгородского государственного университета. Серия: Экономика. Информатика. 2011. Т. 19. № 13-1 (108). С. 180-185.
3. Сценарные языки программирования на сайте Игоря Гаршина. Скрипты. URL: <http://www.garshin.ru/it/computer-languages/scripts.html> (дата обращения: 04.05.2017).
4. Седых А. Обзор языков описания сценариев // Образование, наука, производство Белгородский государственный технологический университет им. В.Г. Шухова. 2015. С. 2714-2719.
5. Гевлич М.В., Рязанов Ю.Д. Интерпретатор скриптов графической системы // Информационные технологии в управлении и моделировании: Сб. докл. Международной науч.-технич. интернет-конф. – Белгород: Изд-во БГТУ им. В.Г. Шухова, 2005. – С.54–56.
6. Непейвода. Н. Н. 13. Лекция: Событийное программирование // Стили и методы программирования. курс лекций. учебное пособие. – М.: Интернет-университет информационных технологий, 2005. – С. 213-222. – 316 с. – ISBN 5-9556-0023-X.
7. Callback Hell. URL: <http://callbackhell.com/> (дата обращения: 04.05.2017).
8. Неизменность и чистота <- О Haskell по-человечески. URL: <https://www.ohaskell.guide/immutability-n-purity.html> (дата обращения: 04.05.2017).
9. Замыкания в php / Блог компании Mail.Ru Group / Хабрахабр. URL: <https://habrahabr.ru/company/mailru/blog/103983/> (дата обращения: 04.05.2017).
10. Ликбез по типизации в языках программирования / Хабрахабр. URL: <https://habrahabr.ru/post/161205/> (дата обращения: 04.05.2017).
11. SoftKey.info: Статьи - Рефакторинг в Visual Studio. URL: <http://www.softkey.info/reviews/review5449.php> (дата обращения: 04.05.2017).
12. Null References: The Billion Dollar Mistake. URL: <https://www.infoq.com/presentations/Null-References-The-Billion-Dollar-Mistake-Tony-Noare> (дата обращения: 04.05.2017).

13. Exploring the Abyss of Null and Undefined in JavaScript - Modern Web. URL: <https://modernweb.com/exploring-the-abyss-of-null-and-undefined-in-javascript/> (дата обращения: 04.05.2017).
14. Null, великий и ужасный / Хабрахабр. URL: <https://habrahabr.ru/post/309462/> (дата обращения: 04.05.2017).
15. Значение Null. Nullable-типы. Оператор ?. URL: http://mycsharp.ru/post/47/2014_09_30_znachenie_null_nullable-tipy_operator_?.html (дата обращения: 04.05.2017).
16. Haskell/Understanding monads/Maybe - Wikibooks, open books for an open world. URL: https://en.wikibooks.org/wiki/Haskell/Understanding_monads/Maybe (дата обращения: 04.05.2017).
17. Expressions - Apache FreeMarker Manual. URL: http://freemarker.org/docs/dgui_template_exp.html#dgui_template_exp_missing (дата обращения: 04.05.2017).
18. Charlie Hunt, Binu John. Java Performance. Prentice Hall, 2012, 693 P.
19. Нативные методы и вызовы нативного интерфейса Java (JNI) в Android приложении | Src-CODE.Net. URL: <http://src-code.net/nativnye-metody-i-vyzovy-nativnogo-interfejsa-java-jni-v-android-prilozhenii/> (дата обращения: 04.05.2017).
20. Selectors | jQuery API Documentation. URL: <https://api.jquery.com/category/selectors/> (дата обращения: 04.05.2017).
21. Анатомия игровых движков / Игры. URL: <https://3dnews.ru/games/engines/> (дата обращения: 04.05.2017).
22. Interrupted Exception. URL: <http://bazhenov.me/blog/2009/09/04/interrupted-exception.html> (дата обращения: 04.05.2017).
23. CSS Selectors Reference. URL: https://www.w3schools.com/cssref/css_selectors.asp (дата обращения: 04.05.2017).
24. Введение в Java Reflection API. URL: <http://www.quizful.net/post/java-reflection-api> (дата обращения: 04.05.2017).
25. Операции над множествами - MathHelpPlanet. URL: <http://mathhelpplanet.com/static.php?p=operatsii-nad-mnozhestvami> (дата обращения: 04.05.2017).
26. Parlante, Nick. Linked List Basics. Stanford CS Education Library (2001). 26 P.
27. Функции высшего порядка | Выразительный Javascript. URL: https://karmazzin.gitbooks.io/eloquentjavascript_ru/content/chapters/chapter5.html (дата обращения: 04.05.2017).
28. Каррирование и частичное применение | Src-CODE.Net. URL: <http://src-code.net/karrirovanie-i-chastichnoe-primenenie/> (дата обращения: 04.05.2017).
29. Васильев А. Н. Java: объектно-ориентированное программирование : для магистров и бакалавров : базовый курс по объектно-ориентированному программированию. СПб: Издательский дом "Питер", 2011. 395 с.
30. Асинхронное программирование. URL: <http://javascript.ru/unsorted/async> (дата обращения: 04.05.2017).
31. Аспектно-ориентированное программирование (АОП): Для чего его лучше использовать? URL: <https://www.ibm.com/developerworks/ru/library/pollice/> (дата обращения: 04.05.2017).
32. Java Programming Environment and the Java Runtime Environment (JRE) (JDK 1.1 for Solaris Developer's Guide). URL: <https://docs.oracle.com/cd/E19455-01/806-3461/6jck06gqd/index.html> (дата обращения: 04.05.2017).

References

1. "Computer Languages History", last modified May 04, 2017, <https://www.levenez.com/lang/>.
2. Muromcev V.V., Lomakin V.V., Tsotsoryna N.V. Developing of specified language for microcontroller remote programming. Scientific bulletins of the Belgorod State University. Series: Economics. Informatics. Т. 19. № 13-1 (2011): p. 180-185.
3. "Scenario programming languages on the site of Igor Garshin. Scripts", last modified May 04, 2017, <http://www.garshin.ru/it/computer-languages/scripts.html>.
4. Sedykh A. Overview of scripting languages. Education, science, and production of Belgorod State Technological University. named after V.G. Shoukhov (2015): p. 2714-2719.
5. Gievlich M.V., Ryazanov U.D. Script interpreter for the graphics system. Information technologies in management and modeling. The collection of reports of the scientific and technical conference. Belgorod: publishing house of BSTU named after V.G. Shoukhov (2005): p.54-56.
6. Nepeyvoda N. N. 13. Lecture: Event-driven programming. Programming styles and methods. Lecture course. Tutorial. Moscow: Internet University of Information Technologies (2005): p. 213—222. ISBN 5-9556-0023-X.
7. "Callback Hell", last modified May 04, 2017, <http://callbackhell.com/>.
8. "Immutability and purity <- simply about Haskell", last modified May 04, 2017, <https://www.ohaskell.guide/immutability-n-purity.html>.
9. "Php closures / Mail.Ru Group company blog / Habrahabr", last modified May 04, 2017, <https://habrahabr.ru/company/mailru/blog/103983/>.
10. "Educational program about programming languages type systems / Habrahabr", last modified May 04, 2017, <https://habrahabr.ru/post/161205/>.

11. "SoftKey.info: Articles - Refactoring in Visual Studio", last modified May 04, 2017, <http://www.softkey.info/reviews/review5449.php>.
12. "Null References: The Billion Dollar Mistake", last modified May 04, 2017, <https://www.infoq.com/presentations/Null-References-The-Billion-Dollar-Mistake-Tony-Hoare>.
13. "Exploring the Abyss of Null and Undefined in JavaScript - Modern Web", last modified May 04, 2017, <https://modernweb.com/exploring-the-abyss-of-null-and-undefined-in-javascript/>.
14. "Null, the great and terrible / Habrahabr", last modified May 04, 2017, <https://habrahabr.ru/post/309462/>.
15. "Null value. Nullable types. ?? operator", last modified May 04, 2017, http://mycsharp.ru/post/47/2014_09_30_znachenie_null_nullable-tipy_operator____.html.
16. "Haskell/Understanding monads/Maybe - Wikibooks, open books for an open world", last modified May 04, 2017, https://en.wikibooks.org/wiki/Haskell/Understanding_monads/Maybe.
17. "Expressions - Apache FreeMarker Manual", last modified May 04, 2017, http://freemarker.org/docs/dgui_template_exp.html#dgui_template_exp_missing.
18. Charlie Hunt, Binu John. Java Performance. Prentice Hall (2012). 693 P.
19. "Native methods and call of native Java interfaces with JNI in Android application | Src-CODE.Net", last modified May 04, 2017, <http://src-code.net/nativnye-metody-i-vyzovy-nativnogo-interfejsa-java-jni-v-android-prilozhenii/>.
20. "Selectors | jQuery API Documentation", last modified May 04, 2017, <https://api.jquery.com/category/selectors/>.
21. "Game engine anatomy / Games", last modified May 04, 2017, <https://3dnews.ru/games/engines/>.
22. "Interrupted Exception", last modified May 04, 2017, <http://bazhenov.me/blog/2009/09/04/interrupted-exception.html>.
23. "CSS Selectors Reference", last modified May 04, 2017, https://www.w3schools.com/cssref/css_selectors.asp.
24. "Introduction into Java Reflection API", last modified May 04, 2017, <http://www.quizful.net/post/java-reflection-api>.
25. "Operations under the mathematical sets - MathHelpPlanet", last modified May 04, 2017, <http://mathhelpplanet.com/static.php?p=operatsii-nad-mnozhestvami>.
26. Parlante, Nick. Linked List Basics. Stanford CS Education Library (2001). 26 P.
27. "Higher order functions | Eloquently Javascript", last modified May 04, 2017, https://karmazzin.gitbooks.io/eloquentjavascript_ru/content/chapters/chapter5.html.
28. "Currying and partial application | Src-CODE.Net", last modified May 04, 2017, <http://src-code.net/karrirovanie-i-chastichnoe-primenenie/>.
29. Vasylyev A.N. Java: object-oriented programming: for master and bachelor degrees : basic course of object-oriented programming. St. Petersburg: Publishing house "Piter", 2011. p. 395.
30. "Asynchronous programming", last modified May 04, 2017, <http://javascript.ru/unsorted/async>.
31. "Aspect-oriented programming (AOP): Best use cases", last modified May 04, 2017, <https://www.ibm.com/developerworks/ru/library/pollice/>.
32. "Java Programming Environment and the Java Runtime Environment (JRE) (JDK 1.1 for Solaris Developer's Guide)", last modified May 04, 2017, <https://docs.oracle.com/cd/E19455-01/806-3461/6jck06gqd/index.html>.

Седых Артём, магистрант кафедры программного обеспечения вычислительной техники и автоматизированных систем

Рязанов Юрий Дмитриевич, доцент, доцент кафедры программного обеспечения вычислительной техники и автоматизированных систем,

Sedykh Arteom, Master Degree Student

Ryazanov Yuriy Dmitrievich, Associate Professor, Department of Software Computer Technology and Automated Systems

УДК 004.891

DOI: 10.18413/2518-1092-2017-2-2-34-39

Нестерова Е.В.
Игрунова С.В.
Зайцева Т.В.
Путивцева Н.П.
Пусная О.П.
Рябцева Я.Н.**ОБУЧАЮЩАЯ ЭКСПЕРТНАЯ СИСТЕМА: ОПЫТ РАЗРАБОТКИ
И ИСПОЛЬЗОВАНИЯ В УЧЕБНОМ ПРОЦЕССЕ**Белгородский государственный национальный исследовательский университет, ул. Победы д.85,
г. Белгород, 308015, Россия*e-mail: nesterova@bsu.edu.ru, igrunova@bsu.edu.ru, zaitseva@bsu.edu.ru, putivzeva@bsu.edu.ru, pusnaya@bsu.edu.ru,
1020342@bsu.edu.ru***Аннотация**

В статье описываются принципы проектирования обучающей экспертной системы, выделены типы знаний необходимых для построения экспертных систем образовательного назначения, реализован пример «Выбор банка для получения кредита».

Ключевые слова: интеллектуальная обучающая система, экспертная система.

UDC 004.891

Nesterova E.V.
Igrunova S.V.
Zaitseva T.V.
Putivtseva N.P.
Pusnaya O.P.
Ryabtseva Ya.N.**TEACHING EXPERT SYSTEM: EXPERIENCE IN DEVELOPMENT
AND USE IN THE LEARNING PROCESS**

Belgorod State National Research University, 85 Pobedy St., Belgorod, 308015, Russia

*e-mail: nesterova@bsu.edu.ru, igrunova@bsu.edu.ru, zaitseva@bsu.edu.ru, putivzeva@bsu.edu.ru, pusnaya@bsu.edu.ru,
1020342@bsu.edu.ru***Abstract**

The article contains a description of the design principles of the training expert system. In the article, the types of knowledge that are necessary for building expert systems for educational purposes are highlighted, and the example "Choosing a bank to obtain a loan" is implemented.

Keywords: intelligent learning system, expert systems.

В современном мире образовательных услуг информационно-коммуникационные интеллектуальные обучающие системы (ИОС) занимают лидирующие позиции [1]. ИОС имеют большой потенциал и высокие перспективы развития и внедрения достижений в образовательный процесс с целью его глобальной модернизации и перехода на качественно новый уровень обучения для достижения максимально эффективного результата в данной области [2]. Ведущей функцией ИОС является передача информации обучаемому, которая обеспечит оптимальное достижение поставленной цели обучения, создав комфортные условия для обучаемого [1].

Достижения современных информационных технологий (ИТ) в образовательной сфере по направлению подготовки 09.03.02 Информационные системы и технологии по профилю «Информационные системы в финансовых рынках» дают возможность на новом уровне углубиться в изучение дисциплин учебного плана экономического направления путем реализации разработки обучающих экспертных систем для практического освоения требуемых компетенций.

Обучающая экспертная система визуализирует изучаемые объекты, процессы, явления, а также их модели, представляемые в динамике, развитии с одновременным сохранением возможности интерактивного диалогового взаимодействия пользователя с обучающей системой [2].

При использовании современных мультимедийных возможностей в образовательном процессе, а также применяя интегрируемость знаний из экономических наук усиливается мотивация, активизируется познавательная деятельность, происходит интенсификация обучения, усвоение большего объема полезной информации, интерактивное взаимодействие пользователя и системы, обеспечивается лучшее запоминание полученной информации, более длительное ее хранение в памяти и наиболее высокие результаты практического применения полученных знаний [2].

В рамках дисциплины ИСиТ предполагается разработка ОИС с элементами интеллектуализации и экспертных систем (ЭС).

Это предполагает создание и наполнение специализированной информацией баз данных и баз знаний (БЗ) проблемных областей для практического освоения своего направления подготовки и формализации процесса принятия решений.

В процессе разработки ИОС перед студентами были поставлены задачи: исследование предметной области; проектирование интеллектуальной системы; создание интеллектуальной системы по подбору объекта; тестирование системы.

Структурный подход к построению модели предметной области предполагает выделение следующих когнитивных элементов знаний: понятия, взаимосвязи, метапонятия, семантические отношения. В качестве предметной области была взята тема «Выбор банка для получения кредита».

В связи с тем, выделяемые понятия предметной области должны образовывать систему, под которой понимается совокупность понятий, обладающая следующими свойствами: уникальностью (отсутствием избыточности); полнотой (достаточно полным описанием различных процессов, фактов, явлений и т. д. предметной области); достоверностью (смысловой информации их реальным наименованиям и непротиворечивостью).

При проектировании ЭС студент выступает в двух ролях как эксперта, так и инженера по знаниям.

Эксперт разбивает задачу на подзадачи для перечисления целевых состояний и описания общих категорий цели, используя «метод локального представления». Далее для каждого разбиения он формулирует информационные факты и дает им четкое наименование.

Решено использовать стратегию, ориентированную на узкую проблемную область.

Выступая в роли инженера по знаниям студент определяет контуры разрабатываемой экспертной системы, а как эксперт осуществляет параметризацию системы.

К основным параметрам проблемной области (например, выбор банка для кредитования) относятся следующие: «интерпретация, диагностика, коррекция, прогнозирование, планирование, проектирование; критерии эффективности результатов решения задач (минимизация использования ресурсов, повышение качества обслуживания, ускорение оборачиваемости капитала и т.д.); критерии эффективности процесса решения задач (повышение точности принимаемых решений, учет большего числа факторов, просчет большего числа альтернативных вариантов, адаптивность к изменениям проблемной области и информационных потребностей пользователей, сокращение сроков принятия решений»; цели решаемых задач (выбор из альтернатив, например, выбор банка для кредитования); «подцели (разбиение задачи на подзадачи, для каждой из которых определяется своя цель); исходные данные (совокупность используемых факторов); особенности используемых знаний» [3].

Результатом построения концептуальной модели выступают: объектная модель, которая описывает структуру предметной области как совокупности взаимосвязанных объектов; функциональная модель, отражающая действия и преобразования над объектами; поведенческая модель, которая рассматривает взаимодействия объектов во временном аспекте.

Рассмотренная объектная модель состоит из элементарных единиц структурного знания, каждая из которых описывает одно свойство или одну связь объекта и представлена в виде «триплета: предикат (Объект, Значение)» [3].

На основании этой модели можно выделить, например, шесть видов кредита, представленных на рисунке 1.

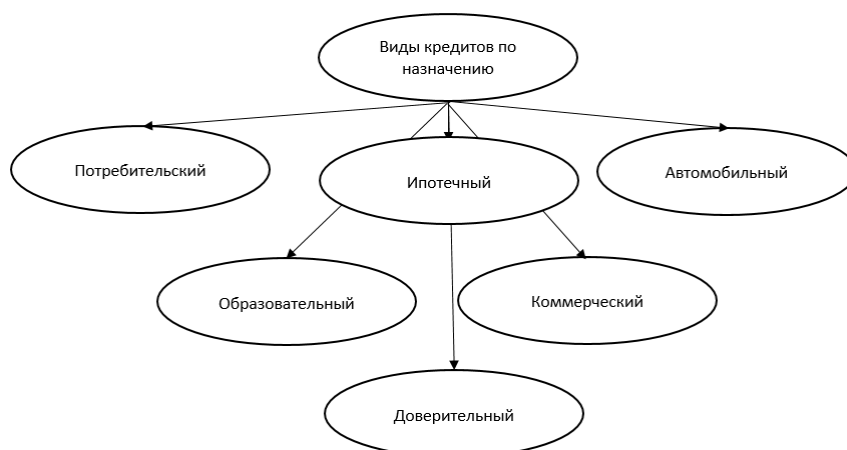


Рис. 1. Виды кредитов по назначению
Fig. 1. Types of loans by appointment

Затем, следует учитывать различные критерии, характеризующие объект, а именно: наличие возможности залога; сроки погашения кредита; ставка процента; сумма кредита; наличие возможности поручительства. Так как предикат определяет название свойства объекта, то в качестве значения выступает конкретное свойство, например: кредит («Потребительский», «Сумма кредита») и т.д.

После определения основных понятий были выявлены связи между ними. На рисунке 2 представлены связи между понятиями.

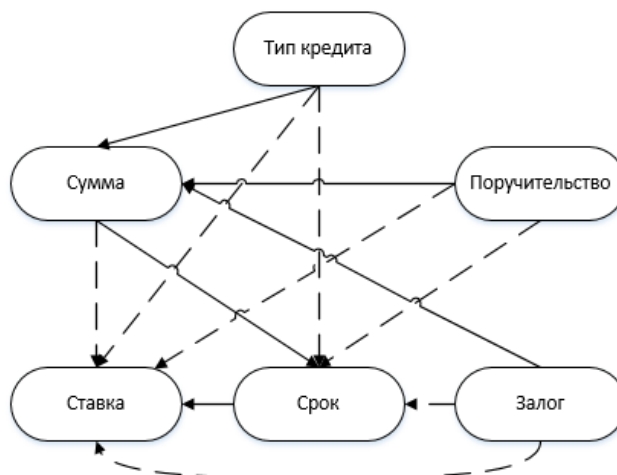


Рис. 2. Связи между понятиями
Fig. 2. Relations between concepts

Срок погашения напрямую зависит от выбранной суммы, он влияет на процентную ставку, таким образом, сумма косвенно влияет на ставку. Тип кредита непосредственно влияет на сумму и косвенно на ставку и срок. Залог и поручительство напрямую влияет на сумму и в свою очередь косвенно на срок погашения кредита и процентную ставку.

При построении функциональной модели, преобразовываются понятия и зависимости между ними, показывающие, как одни понятия образуются из других. В качестве единицы функционального знания определим функциональную зависимость понятий в виде импликации:

$$K_1 \cap K_2 \cap K_3 \dots \cap K_n \rightarrow B,$$

где

K_n – критерии понятий предметной области,

B – понятие предметной области.

Например, $Z \cap SrK \cap Pr \cap SumK \cap Por \rightarrow$ Потребительский,
где

Z – наличие возможности залога; SrK – сроки погашения кредита; Pr – ставка процента; $SumK$ – сумма кредита; Por – наличие возможности поручительства.

Потребительский – факт выбора кредита, будет иметь место только в том случае, если имеет место конъюнкция фактов или их отрицаний $Z, SrK, Pr, SumK, Por$, например Потребительский (Ставка процента, 12%), и Потребительский (Поручительство, Да) и Потребительский (Социальный статус, Зарплата) \rightarrow Банк (Кредит, Потребительский).

Следовательно, предметная область в каждый момент времени может быть представлена в виде совокупности сущностей, понятий и ситуаций. Выделенная совокупность сущностей, понятий и ситуаций предметной области называется ее состоянием [4]. Так как понятия различаются между собой с помощью признаков, то состояние предметной области можно задать, если известны значения всех признаков понятий, используемых для ее описания.

Для отражения отношений между понятиями создается концептуальная модель, представленная на рисунке 3.



Рис. 3. Концептуальная модель выбора банка для получения кредита

Fig. 3. Conceptual model for choosing a bank to obtain a loan

Онтологическое исследование, отображенное в концептуальной модели, включает совокупность сущностей, понятий и связей между ними, согласно которым эти сущности могут быть скомбинированы для построения достоверных утверждений о состоянии рассматриваемой системы в некоторый момент времени. Эти утверждения помогают сделать выводы пользователю системы.

Таким образом, для области кредитования определены основные понятия и метапонятия, построена концептуальная схема, определяющая отношения между понятиями.

В качестве интегрированной среды разработки было решено применить инструментальное средство для создания экспертных систем ESWin. Основу программной оболочки ESWin представляет база знаний, которая состоит из правил-продукций и набора фреймов. Структура представления базы знаний представлена на рисунке 4.

```

TITLE = <название экспертной системы>
COMPANY = <название предприятия>
FRAME // фрейм
<описание фрейма>
ENDF

FRAME // фрейм
<описание фрейма>
ENDF
RULE // правило-продукция
<описание условий правила>
DO
<описание заключений правила>
ENDR

RULE // правило-продукция
<описание условий правила>
DO
<описание заключений правила>
ENDR
    
```

Рис. 4. Структура представления базы знаний

Fig. 4. Structure of knowledge representation

Правило – продукция представляет знания как предложения ЕСЛИ (условие)? ТО (заключение). В данном случае условием будет являться образец, по которому проводится поиск в базе знаний, а заключением операции, которые выполняются при благоприятном результате поиска. Выделяют два вида действий: целевые, которые завершают работу системы и являются результатом решения задачи и промежуточные, которые выступают далее, как условия.

Описание структуры фреймов включает в себя: заголовок базы знаний, начало и конец фрейма, слоты, вопросы слотов, их значения. Ключевыми словами для формирования заголовка базы знаний является запись: TITLE=Выбор банка.

Начало фрейма «Тип» представлено в виде строки: FRAME=Тип.

Формулировка задачи, решаемой экспертной системой записана, как: Тип кредита (Какой кредит Вы хотите получить?).

Конец фрейма «Тип» определен словом «ENDF».

Слово «Тип кредита» – имя слота.

(Какой кредит Вы хотите получить?) – вопрос слота.

(потребительский; ипотечный; автомобильный; образовательный; коммерческий; доверительный).

На рисунке 5 представлен пример описания правила «Rule 1».

```

Rule 1
EQ(Тип.Тип кредита; потребительский)
<(Сумма.Предполагаемая сумма кредита; 1000000)
=(Срок.Срок; 1-3 года)
=(Ставка.Ставка; 10-15%)
=(Поручитель.Поручитель; да)
=(Залог.Залог; да)
Do
=(Выбрать банк; Альфа банк) 100
EndR
    
```

Рис. 5. Описание правила

Fig. 5. Description of the rule

В описание структуры правила «Rule 1» входит следующее: начало и конец правила, проверка условия и запись заключения, выдаваемого экспертной системой на экран в виде текста «Выбрать банк; Альфа банк с уверенностью 100%».

Чтобы увидеть вышеуказанное заключение системы, пользователю необходимо ответить на вопросы таким образом, чтобы в первом вопросе он выбрал значение «потребительский», во втором – ввести сумму

до 1 млн. руб., в третьем – «1-3 года», в четвертом – «10-15%», в пятом – «да», в шестом – «да». Структура созданной учебной базы знаний включает в себя 6 вопросов и 1152 правила.

Таким образом, для упрощения выбора банка предлагается специально созданная обучающая экспертная система. Она построена так, что вопросы получаются простыми и незатруднительными, но при этом происходит четкий анализ ответов, предусмотренных экспертом, для предложения наиболее точного решения задачи.

Список литературы

1. Carbonell J. R. AI in CAI: An artificial-intelligence approach to computer-assisted instruction // IEEE transactions on man-machine systems. – 1970. – Т. 11. – №. 4. – С. 190-202.
2. Журкин А. А. Использование технологий визуализации и полисенсорного представления обучающего материала в интеллектуальных обучающих системах // Ученые записки. Электронный научный журнал Курского государственного университета. 2013. №3(27). URL: <http://cyberleninka.ru/article/n/ispolzovanie-tehnologiy-vizualizatsii-i-polisensornogo-predstavleniya-obuchayuschego-materiala-v-intellektualnyh-obuchayuschih> (дата обращения: 30.04.2017).
3. Зайцева Т.В., Игрунова С.В., Путивцева Н.П., Пусная О.П., Нестерова Е.В. Использование семиотического подхода к представлению знаний для построения модели логической структуры учебного материала, Серия История. Политология. Экономика. Информатика. Белгород Научные ведомости БелГУ. - 2011. - № 19(108). – С. 143-149.
4. Тельнов Ю. Ф., Трёмбач В. М. Интеллектуальные информационные системы. – 2009. – 222 с.

References

1. Carbonell J. R. AI in CAI: An artificial-intelligence approach to computer-assisted instruction // IEEE transactions on man-machine systems. – 1970. – Т. 11. – №. 4. – P. 190-202.
2. Zhurkin A. A. Use of visualization technologies and polysensory representation of teaching material in intelligent training systems // Scientists note. Electronic scientific journal of the Kursk State University. – 2013. – № 3(27).
3. Zajceva T.V., Igrunova S.V., Putivceva N.P., Pusnaja O.P., Nesterova E.V. The use of a semiotic approach to knowledge representation to build a model of logical structure of educational material. Belgorod State University Scientific Bulletin. – 2011. – № 19(108). – P. 143-149.
4. Telnov Yu. F., Trembach V. M. Intellektualnyie informatsionnyie sistemyi. – 2009. – 222 p.

Нестерова Елена Викторовна, старший преподаватель кафедры информационных систем

Игрунова Светлана Васильевна, доцент кафедры информационных систем, кандидат социологических наук

Зайцева Татьяна Валентиновна, доцент кафедры прикладной информатики и информационных технологий, кандидат технических наук, доцент

Путивцева Наталья Павловна, доцент кафедры прикладной информатики и информационных технологий, кандидат технических наук

Пусная Ольга Петровна, старший преподаватель кафедры прикладной информатики и информационных технологий

Рябцева Яна Николаевна, студент кафедры информационных систем

Nesterova Elena Victorovna, Senior Lecturer, Department of Information Systems

Igrunova Svetlana Vasilievna, Associate Professor, Department of Information Systems, PhD in Sociology

Zaitseva Tatyana Valentinovna, Associate Professor, Department of Applied Informatics and Information Technologies, PhD in Technical Sciences

Putivtseva Natalia Pavlovna, Associate Professor, Department of Applied Informatics and Information Technologies, PhD in Technical Sciences

Pusnaya Olga Petrovna, Senior Lecturer, Department of Applied Informatics and Information Technologies

Ryabtseva Yana Nicolaevna, Student, Department of Information Systems

**ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ
INFORMATION TECHNOLOGIES AND TELECOMMUNICATION**

UDC 004.415.24

DOI: 10.18413/2518-1092-2017-2-2-40-48

Balabanova T.N.
Likhogodina E.S.
Vodounou A.C.
Guryanova O.I.

**THE USING OF ORTHOGONAL BASIS FOR THE STEGANOGRAPHIC
CODING OF INFORMATION IN MULTIMEDIA**

Belgorod State National Research University, 85 Pobedy St., Belgorod, 308015, Russia

e-mail: Sozonova@bsu.edu.ru, elza9313@gmail.ru, aaron.vodounou@gmail.com, Guryanova_o@bsu.edu.ru

Abstract

This article discusses methods for steganographic encoding additional information using three different orthogonal bases. The bases are represented by the functions, which occupies a different bandwidth in the spectrum. There are comparison of the approach to the definition of DCT-coefficients with the approaches used in the methods of the spread spectrum and subband projections. The approaches of the coefficients of the implementation to ensure secrecy by adaptive determination of their value are considered. However, their value is determined based on the energy structure of the segment of the speech signal. Criteria to evaluate steganographic encoding are the secrecy and accuracy of decoding control information. As the control information is a sequence of numbers in binary form. For the proposed principles of adaptation the results of numerical experiments the estimates that determine stealth: mean square error, the distance Itakura-Saito, correlation. In the computational experiments was found the probability of error for bits at different signal-to-noise ratio. The corresponding computational experiments were carried out for all outlined approaches.

Keywords: steganography; orthogonal basis; adaptive threshold of implementation; discrete cosine transform.

УДК 004.415.24

Балабанова Т.Н.
Лихогодина Е.С.
Водуну А.К.
Гурьянова О.И.

**ИСПОЛЬЗОВАНИЕ ОРТОГОНАЛЬНОГО БАЗИСА
ДЛЯ СТЕГАНОГРАФИЧЕСКОГО КОДИРОВАНИЯ ИНФОРМАЦИИ
В МУЛЬТИМЕДИА**

Белгородский государственный национальный исследовательский университет, ул. Победы д.85,
г. Белгород, 308015, Россия

e-mail: Sozonova@bsu.edu.ru, elza9313@gmail.ru, aaron.vodounou@gmail.com, Guryanova_o@bsu.edu.ru

Аннотация

В данной статье рассматриваются методы стеганографического кодирования дополнительной информации с использованием трех различных ортогональных базисов. Базисы представлены функциями, занимающими разные по ширине полосы частот в спектре. Приведено сравнение подхода к определению ДКП-коэффициентов с подходами, использующимися в методах расширения спектра и субполосных проекций. Рассмотрены подходы выбора коэффициентов внедрения для обеспечения скрытности путем адаптивного определения их величины. При этом их величина определяется, исходя из

энергетической структуры отрезка речевого сигнала. Критериями, оценивающими стеганографическое кодирование, являются скрытность и достоверность декодирования контрольной информации. В качестве контрольной информации используется последовательность чисел в двоичном виде. Для предложенных принципов адаптации в результате вычислительных экспериментов получены оценки, определяющие скрытность: среднеквадратическая ошибка, расстояние Итакуры-Сайто, корреляция. В ходе вычислительных экспериментов была найдена вероятность ошибки на бит при различном отношении сигнал/шум. Соответствующие вычислительные эксперименты были проведены для всех изложенных подходов.

Ключевые слова: стеганография; ортонормальный базис; адаптивный порог внедрения; дискретное косинусное преобразование.

INTRODUCTION

Speech is the most common and natural method of the transmission of information between the people. For the transfer up to the distance spoken language is fixed, and they convert the result of fixation into the code sequence. In the methods of coding, it is possible to isolate a number of the characteristic operations, one of which is the removal of redundancy for decreasing the volume of the transferred code combinations. With the strong decrease of volume (high compression ratio) are possible the changes with which the reproducible speech will be essentially they will differ from the initial. Often this does not influence the transmission of information. In cases when information is important and it is necessary to ensure its authenticity, but the channel capacity of communications does not make it possible to transmit redundant information, in this case for guaranteeing the authenticity it is possible to use methods of cryptography.

The procedures of the decrease of redundancy, as the methods of cryptography are combined with the use of psychoacoustic models. Naturally, for achievement maximum compression are moved away all frequency-time components, which carry in themselves the redundancy, determined based on psychoacoustic models [1, 2]. This does not make it possible to use excess frequency-time components for coding of additional information. By additional information, we will understand the digital code, which makes it possible to determine the authenticity of speech.

MAIN PART

For the solution of the problem of coding additional information, it is proposed to use the methods, based on the mathematical approach different from that, which was used with the compression. It is worthwhile to note that for guaranteeing the durability of information coding must be accomplished in the space (further the space of coding), and decoding in other space (further the space of decoding).

Ensuring reserve one of requirements imposed to steganographic methods [1-8]. Ensuring reserve is reached when decoding in a component(s) containing the smallest share of energy (fig. 1).

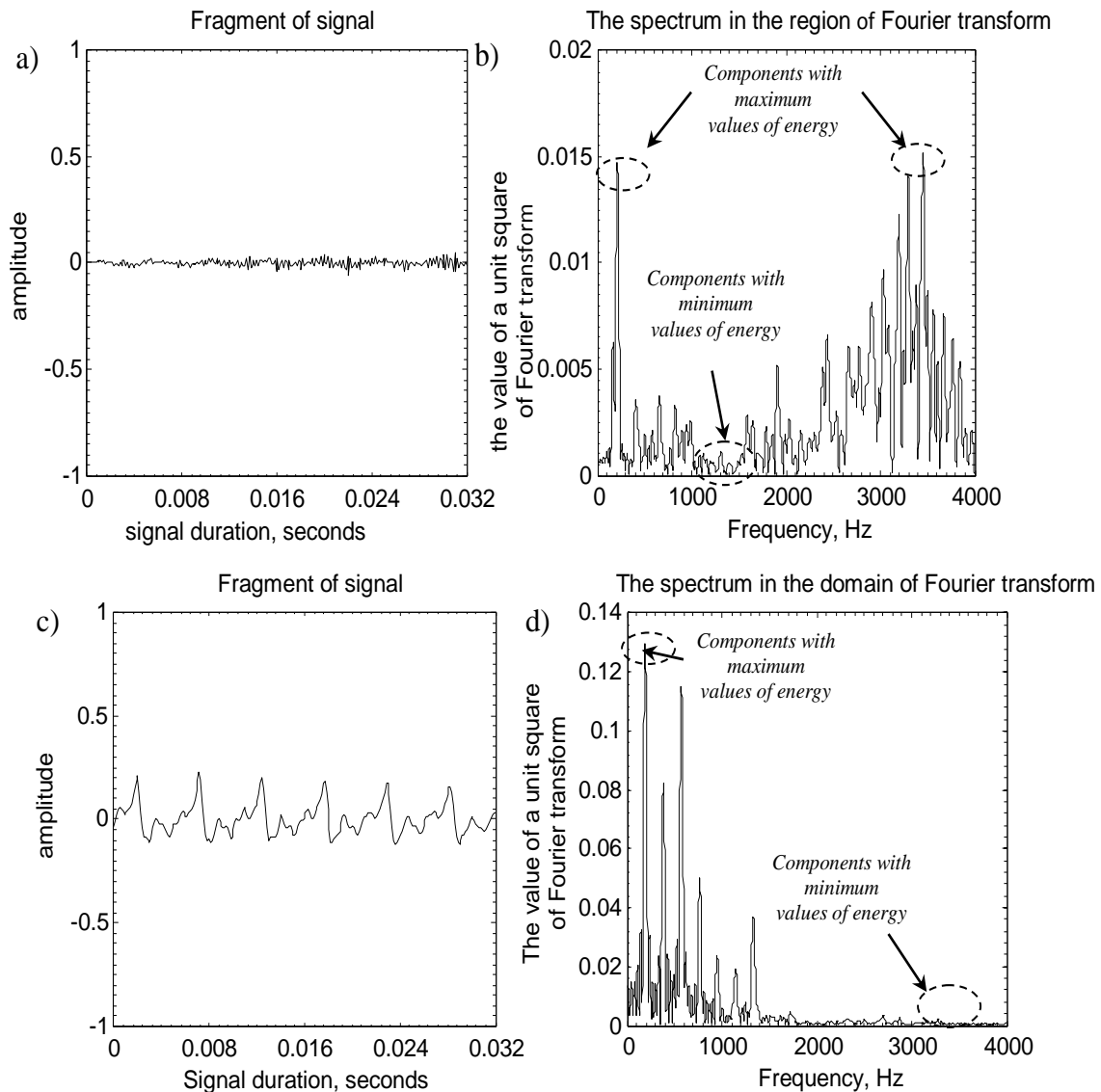


Fig. 1. Audio-signal pieces: a) sound "sh"; b) a range for a sound "sh"; c) a sound "o"; d) a range for a sound "o"

At the same time, not unimportant value plays ensuring probability of an error of decoding of the hidden information, close to zero. Reduction of probability of an error of decoding can be reached thanks to coding of information in a signal component(s) the having overwhelming share of energy of rather synthesizable piece (fig. 1). In this regard, there is a need of the choice between firmness and reserve, for this purpose choose to a component in which coding is carried out. The choice of the fixed threshold or coding in in advance set number components, doesn't provide necessary reserve [9], it is visually illustrated in fig. 1. Apparently from ranges (fig. 1, b and d) sounds "o" and "ш" having different distribution of energy on a frequency axis. The choice, components need to be carried out proceeding from time-and-frequency characteristics of a piece in which reserved coding is carried out, i.e. is adapted to choose to a component for coding [1, 10]. For achievement of high reserve and reduction of probability of a mistake, adaptation under each piece, it is offered to carry out, using the average value having on a component.

Let us consider one of widespread methods of the steganography coding using decomposition of a piece of an audio-signal on DCT coefficients of a look [11, 12]:

$$g_0 = \frac{\sqrt{2}}{N} \sum_{i=1}^N x_i, \quad (1)$$

$$g_m = \frac{2}{N} \sum_{i=0}^{N-1} x_i \cdot \cos\left(\frac{(2i+1) \cdot m\pi}{2N}\right), \quad m=1,2,\dots,(N-1), \quad (2)$$

where x_i – value of signal amplitude; m – number of DCT coefficient; g_m – DCT coefficient.

Results of calculation of DCT coefficients for segments of the audio signals given on fig. 1 are given below.

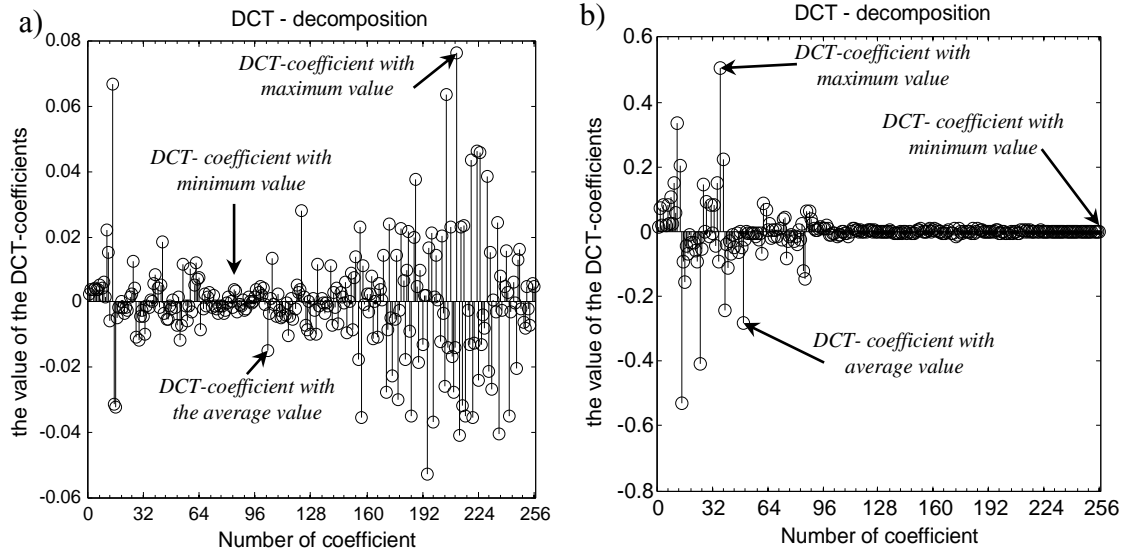


Fig. 2. DCT coefficients: a) sound "sh"; b) sound "o"

An alternative proposed method [13] choice of coefficients is underwritten method.

Among the calculated DCT coefficients (2), it is possible to select the component defined according to one of rules:

– the DCT coefficient having the minimum value:

$$\left(\|\bar{x}\|^2 - g_k^2\right) = \max_{k \in \{1,2,\dots,N\}} \varepsilon. \quad (3)$$

– the DCT coefficient the close to mean value:

$$\left(\frac{2}{N} \cdot \|\bar{x}\|^2 - g_k^2\right) = \min_{k \in \{1,2,\dots,N\}} \varepsilon. \quad (4)$$

– the DCT coefficient having the maximum value:

$$\left(\|\bar{x}\|^2 - g_k^2\right) = \min_{k \in \{1,2,\dots,N\}} \varepsilon. \quad (5)$$

The operations procedure explained below allows realizing steganographic coding of bit in DCT coefficient:

Input data:

- bit of encoded information of a segment $e_m \in \{-1, 1\}$.
- segment duration N .

- values of amplitudes of a segment: $\bar{x} = (x_1, x_2, \dots, x_i, \dots, x_N)^T$.

Output data:

- Values of amplitudes of a segment: $\bar{y} = (y_1, y_2, \dots, y_i, \dots, y_N)^T$.

1. Let's divide an audio signal into segments \bar{x} , the size N of reports.

2. According to conversion (1) we will calculate DCT coefficients for a segment \bar{x} ,

$\bar{g} = (g_0, g_1, \dots, g_k, \dots, g_m, \dots, g_{N-1})^T$ i.e. it is feasible direct DCT conversion.

3. Let's calculate energy of a segment

$$\|\bar{x}\|^2 = \sum_{i=1}^N x_i^2. \quad (6)$$

4. It agrees to one of rules (3)-(5) we will define number k of DCT coefficients in which we will realize that coding.

5. We realize coding of bit of information e_m , by means of change of a sign of DCT coefficient:

$$c_k = e_m \cdot \text{abs}(g_k). \quad (7)$$

where $\text{abs}(\)$ – the operation discarding a sign y at number; s_k – value of DCT coefficient;

6. We realize the reverse IDCT conversion:

$$y_i = \frac{1}{\sqrt{2}} g_0 + \sum_{m=1}^{k-1} g_m \cdot \cos\left(\frac{(2i-1)m\pi}{2N}\right) + c_k \cdot \cos\left(\frac{(2i-1)k\pi}{2N}\right) + \sum_{m=k+1}^{N-1} g_m \cdot \cos\left(\frac{(2i-1)m\pi}{2N}\right), \quad i = 1, 2, \dots, N. \quad (8)$$

Method of expansion of a range

The essence of a method of expansion of a range consists in addition to a piece of an initial speech signal of the pseudorandom sequence (SSp) according to expression [1, 3, 4, 14]:

$$\vec{y} = \vec{x} + \alpha \cdot e \cdot \vec{u}, \quad (9)$$

where \vec{x} – an initial piece of speech data; \vec{u} – the piece corresponding to the pseudorandom sequence; α – weight coefficient; e – the code display of binary bit of the hidden speech message determined by a formula:

The weight coefficient α defines reserve of system. In works [8, 9] him is offered to be chosen equal:

$$\alpha = \langle \vec{x}, \vec{u} \rangle / \|\vec{u}\|^2. \quad (10)$$

Decoding of bit of control information comes from data by definition of a sign of a scalar product of a piece of data and the pseudorandom sequence:

$$\tilde{e} = \text{sign}(\langle \vec{y}, \vec{u} \rangle), \quad (11)$$

where $\text{sign}(\)$ – operation of allocation of a sign.

Method of subband projections

Also for assessment, the model of a method of subband projections, which is carrying out reserved coding of bits of control information b_m in a piece of speech data \vec{x} is offered [6, 14]:

$$y = \vec{x} + (\text{sign}(e_m) \cdot |\alpha| - \alpha) \cdot \vec{q}, \quad (12)$$

The weight coefficient α defines reserve of system. In works [6, 14] him is offered to be chosen equal:

$$\alpha = \langle \vec{x}, \vec{q} \rangle \quad (13)$$

Decoding of control information is carried out by definition of signs of projections α for own vectors \vec{q} of a subband matrix A_r :

$$\tilde{e}_m = \text{sign}(\langle \vec{y}, \vec{q} \rangle), \quad m \in M, \quad (14)$$

where \hat{e}_m – the symbol decoded by method of subband projections.

Reserve assessment technique

For determination of overall performance of a method, we use indicators the estimating misstatements brought in an audio-signal when coding by the offered approach. For identification of statistics, the following metrics were counted [1, 2, 7, 8, 15]:

Mean square error, MSE:

$$MSE = \sum_{i=1}^N (x_i - y_i)^2, \quad (15)$$

where x_i - value of amplitude of the initial audio signal; y_i - value of amplitude of the synthesized audio signal.

Correlation ρ :

$$\rho = \frac{\left(\sum_{i=1}^N (x_i - \bar{x}) \cdot \sum_{i=1}^N (y_i - \bar{y}) \right)}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2 \cdot \sum_{i=1}^N (y_i - \bar{y})^2}}, \quad (16)$$

where \bar{x} – a constant component of an initial audio-signal; \bar{y} – a constant component of the synthesized audio-signal.

Changes in a time domain it is also necessary to consider distinctions in frequency area. The measure based on Itakura-Saito's distance is for this purpose used [15, 16]:

$$ISD = \sum_{r=1}^R \Delta\omega_r \cdot \left(\frac{\tilde{P}_r}{P_r} + \ln \frac{P_r}{\tilde{P}_r} - 1 \right) / \pi, \quad (17)$$

where \tilde{P}_r – value of energy frequency components of an initial piece of data; P_r – value of energy frequency components of the piece of data containing additional information.

The measure makes sense of distance between ranges of two signals and estimates discrepancy between energy of the changed and initial piece of data. At equality of pieces of data the measure addresses in zero.

$$ISD = \sum_{d=1}^R \Delta\omega_d \cdot \left(\frac{\tilde{y}^T \cdot A_d \cdot \tilde{y}}{\tilde{x}^T \cdot A_d \cdot \tilde{x}} + \ln \frac{\tilde{y}^T \cdot A_d \cdot \tilde{y}}{\tilde{x}^T \cdot A_d \cdot \tilde{x}} - 1 \right) / \pi, \quad (18)$$

where A_d – a subband matrix [5]; $\Delta\omega_d$ – width of a frequency interval.

As the tool, allowing to make energy calculations, without passing into the frequency area, it is offered to use a mathematical apparatus of subband matrixes [4, 5]:

$$P_r(\tilde{x}) = \tilde{x}^T A_r \tilde{x}, \quad (19)$$

where A_r – the subband matrix determined by elements:

$$A_r = \{a_{i,k}^r(r)\}, a_{i,k}^r = \sin\left(\frac{2\pi \cdot (\Delta f / 2)}{\mathcal{G}_0} \cdot (i - k)\right) \cos\left(\frac{2\pi \cdot f_0}{\mathcal{G}_0} \cdot (i - k)\right), \quad (20)$$

where i – an element line item in a line of a matrix; k – an element line item in a matrix column; \mathcal{G}_0 – sampling rate; Δf – band width (in case of normalization respectively $\Delta\omega = 2\pi \cdot \Delta f$); f_0 – central frequency (in case of normalization respectively $\omega_0 = 2\pi \cdot f_0$).

Mean squared error (MSE) measures the relative difference between the energy of segments signals in the time domain. This measure allows identifying the differences in the envelopes of the amplitudes of the segments of speech signals. The fewer changes can be made when introduced additional information, the closer the value for this score to zero [15]:

$$\sigma = \sum_{n=1}^N (x_n - y_n)^2 / \sum_{n=1}^N x_n^2. \quad (21)$$

where x_n – value of amplitude of the initial segment of data; y_n – value of amplitude of the segment of data containing additional information, N – the number of counting of the compared segments of signals.

Reliability assessment technique

Assessment of reliability of decodable information, we will carry out proceeding from probability of an error (BER) [1, 7]:

$$BER = \frac{1}{M} \sum_{m=1}^M (((\text{sign}(e_m) + 1) / 2) \oplus ((\text{sign}(\tilde{e}_m) + 1) / 2)). \quad (22)$$

where M – the number of encoded bits; \oplus – operation "the amount on the module two"; $\text{sign}()$ – operation of separation of a sign; \tilde{e}_m – decodable bit.

Results of simulation

For check of operability of a method based on DCT-conversion, audio signal fragments with sampling rate 8 kHz and digit capacity of 16 bits were used [10]. The general duration of speech material made 23 minutes, lasting 0,032sec, (the segments, which are not containing energy - pauses, were excluded from material). As noise 10^9 not repeating PSP, segments were taken. As a result of simulation it was implemented 10^9 bit, results of simulation are provided to tab. 1.

Table 1

Reliability assessment

№	BER	Noise to signal			
		0.001	0.01	0.1	1
1	Maximum DCT, (3, 8)	≈ 0	≈ 0	≈ 0	$2.1646 \cdot 10^{-5}$
2	Average DCT (4, 8)	≈ 0	≈ 0	$2,4071 \cdot 10^{-5}$	0,0396
3	Minimum DCT, (5, 8)	0,1181	0,1230	0,1247	0,1252
4	SSp, (8, 9, 10)	0,1285	0,1290	0,1439	0,2133
5	SubBand, (8, 10, 12)	0,0219	0,0675	0,1803	0,3345

Results of reserve of the introduced information are given in tab. 2 for the parameters of modelling specified above.

Table 2

Reserve assessment

№	Choice of coefficients principle	MSE	ρ	σ	ISD
1	Maximum, (3)	2.427 E-0	0.8472	5.41 E-01	3.712
2	Average (4)	2.875 E-3	0.9960	4.28 E-04	1.054
3	Minimum, (5)	8.341 E-8	0.9999	2.31 E-16	0.023
4	SSp	1.102 E-3	0.9923	0.14 E-03	0.031
5	SubBand	3.256 E-3	0.9931	1.21 E-16	0.003

CONCLUSIONS

The given algorithm is optimum from a position of the accounting of frequency properties of the audio-signal containing digital submission of the speech as solving the rule considers uneven distribution of energy on a frequency strip and perception of a sound by the person. Use of DCT coefficient with average value of energy, for reserved coding of information, will allow to reduce by two orders changes of energy in we synthesize an audio-signal piece.

References

1. Fridrich, J. 2012. Steganography in digital media: Principles, algorithms, and applications. Steganography in Digital Media, P. 1-441.
2. Furui, Sadaoki. 2000. Digital speech processing, synthesis, and recognition. 2nd ed., rev. and expanded
3. Cox I. J., Kilian J., Leighton F. T., Shamoon T. Secure spread spectrum watermarking for multimedia // IEEE transactions on image processing. – 1997. – V. 6, № 12. – P. 1673-1687.
4. Nedeljko Cvejic, Tapio Seppanen. 2004. Spread spectrum audio watermarking using frequency hopping and attack characterization. Signal Processing 84. P. 207 – 213.
5. Lykholob, P.G. Research of sensitivity of some measures of quality assessment of hidden information in the audio content [Текст] // Medvedeva, A.A., Likhogodina, E.S., Mishina, O.O. RESEARCH RESULT. Information technologies. №4. v.1. 2016. pp.21-25 URL: http://rr.bsu.edu.ru/media/information/2016/4/3_it.pdf DOI: 10.18413/2518-1092-2016-1-4-21-24
6. Zhilyakov E.G., Pashintsev V.P., Belov S.P., Likholob P.G. About the secretive method of encoding control information in the speech data// Infocommunicatsionnye tehnologii. – Samara, 2015. – V. 13, № 3. – P. 325-333.
7. Fridrich, J. 2012. Steganography in digital media: Principles, algorithms, and applications. Steganography in Digital Media, P. 1-441.
8. Furui, Sadaoki. 2000. Digital speech processing, synthesis, and recognition. 2nd ed., rev. and expanded
9. GOST 16600-72. The transmission of speech by radio communication paths. The requirements for intelligibility of speech and methods of articulation measurements [Sound recording] / GOST 16600-72; isp.: D.I. Biblev. – Belgorod: NIU

BelGU, 2016. – 1380 sec. – Access mode: https://www.researchgate.net/publication/312167036_Recording_Gost_16600-72
DOI: 10.13140/RG.2.2.33677.74720

10. Kisilenko A.V., Likhogodina E.S., Likhobol P.G. About choice of the place for hiding information [Text] / Kisilenko A.V., Likhogodina E.S., Likhobol P.G. // *Sovremennoe obschestvo, obrazovanie i nauka. Sbornik nauchnyh trudov po materialam Mezhdunarodnoi nauchno-practicheskoi konferencii: v 9 chastyah.* – Tambov: OOO "Konsaltingovaya kompaniya Yukom", 2014. – P. 76-78

11. Signal processing with lapped transforms. / Malvar H. S. – Boston: Artech House, 1992.

12. Ahmed N., Natarajan T., Rao K. R. Discrete cosine transform // *IEEE transactions on Computers.* – 1974. – V. 100, № 1. – P. 90-93.

13. On uniqueness of determination of identity-relevant frequency bands in the sounds of Russian speech affected by noise [Текст] / Zhilyakov E.G., Likhobol P.G., Kurlov A.V., Medvedeva A.A. // *Nauchnye vedomosti Belgorodskogo gosudarstvennogo universiteta. Seriya: Economica. Informatica.* 2016. V. 37. № 2 (223). P. 167-173

14. Evgeny G. Zhilyakov, Sergey P. Belov, Likhobol P. G., Pashintsev V. P. On the Steganography in Voice Data // *Asian Journal of Information Technology.* – 2016. – V. 15, № 12. – P. 1949-1952.

15. Zhilyakov E.G., Likhobol P.G., Medvedeva A.A., Prochorenko E.I. Research of the sensitivity of certain quality measures to hide information in the speech data // *Nauchnye vedomosti Belgorodskogo gosudarstvennogo universiteta. Seriya: Economica. Informatica.* – Belgorod, 2016. – V. 9, № 230. – P. 174-179.

16. Zhilyakov E.G. Optimal subband methods of analysis and synthesis of signals of finite duration. Automation and mechanics. – M.: Akademicheskij nauchno-izdatelskiy, proizvodstvenno-poligraficheskiy i knigoraspredelitel'skiy tsentr Rossiyskoi akademii nauk "Izdatelstvo "Nauka" № 4, 2015г. P. 51-66

Balabanova Tatyana Nikolaevna, Candidate of Technical Sciences, associate Professor, Department of information and telecommunication systems and technologies

Likhogodina Elizaveta Sergeevna, student, Department of information and telecommunication systems and technologies

Vodounou Aaron Candide, student, Department of information and telecommunication systems and technologies

Guryanova Oksana Igorevna, student, Department of mathematical and software information systems

Балабанова Татьяна Николаевна, доцент кафедры информационно-телекоммуникационных систем и технологий, кандидат технических наук

Федеральное государственное автономное образовательное учреждение высшего образования

Лихогодина Елизавета Сергеевна, студент кафедры информационно-телекоммуникационных систем и технологий

Водуну Аарон Кандид, студент кафедры информационно-телекоммуникационных систем и технологий

Гурьянова Оксана Игоревна, студент кафедры информационно-телекоммуникационных систем и технологий