

3(70)/2014

INFORMATSIONNO- UPRAVLIAIUSHCHIE SISTEMY (INFORMATION AND CONTROL SYSTEMS)

REFEREED EDITION

Founder

«Information and Control Systems», Ltd.

Editor-in-Chief

M. Sergeev

Dr. Sc. Tech., Professor, St.-Petersburg, Russia

Deputy Editor-in-Chief

E. Krouk

Dr. Sc. Tech., Professor, St.-Petersburg, Russia

Executive secretary

O. Muravtsova

Editorial Council

L. Chubraeva

RAS Corr. Member, Dr. Sc. Tech., Professor, St. Petersburg, Russia

L. Fortuna

PhD, Professor, Catania, Italy

A. Fradkov

Dr. Sc. Tech., Professor, St. Petersburg, Russia

V. Kozlov

Dr. Sc. Tech., Professor, St. Petersburg, Russia

C. Christodoulou

PhD, Professor, Albuquerque, New Mexico, USA

B. Meyer

PhD, Professor, Zurich, Switzerland

A. Ovodenko

Dr. Sc. Tech., Professor, St. Petersburg, Russia

Y. Podoplyokin

Dr. Sc. Tech., Professor, St. Petersburg, Russia

Yu. Shokin

RAS Academician, Dr. Sc. Phys.-Math., Novosibirsk, Russia

V. Simakov

Dr. Sc. Tech., Professor, Moscow, Russia

V. Vasilev

RAS Corr. Member, Dr. Sc. Tech., Professor, St. Petersburg, Russia

R. Yusupov

RAS Corr. Member, Dr. Sc. Tech., Professor, St. Petersburg, Russia

Editorial Board

V. Anisimov

Dr. Sc. Tech., Professor, St. Petersburg, Russia

B. Bezruchko

Dr. Sc. Phys.-Math., Saratov, Russia

N. Blaunstein

Dr. Sc. Phys.-Math., Professor, Beer-Sheva, Israel

A. Dudin

Dr. Sc. Tech., Professor, Minsk, Belarus

V. Khimenko

Dr. Sc. Tech., Professor, St. Petersburg, Russia

G. Maltsev

Dr. Sc. Tech., Professor, St. Petersburg, Russia

V. Melekhin

Dr. Sc. Tech., Professor, St. Petersburg, Russia

A. Shalyto

Dr. Sc. Tech., Professor, St. Petersburg, Russia

A. Shepeta

Dr. Sc. Tech., Professor, St. Petersburg, Russia

A. Smirnov

Dr. Sc. Tech., Professor, St. Petersburg, Russia

Z. Yuldashev

Dr. Sc. Tech., Professor, St. Petersburg, Russia

A. Zeifman

Dr. Sc. Phys.-Math., Vologda, Russia

Editor: A. Larionova**Proofreader:** T. Zvertanovskaia**Design:** A. Koleshko, M. Chernenko**Layout and composition:** N. Karavaeva**Contact information**

The Editorial and Publishing Center, SUAI

67, B. Morskaia, 190000, St. Petersburg, Russia

Website: <http://i-us.ru/en>, E-mail: ius.spb@gmail.com

Tel.: +7 - 812 494 70 02

The Journal was registered in the Ministry of Press, Broadcasting and Mass Media of the Russian Federation. Registration Certificate JD № 77-12412 from April, 19, 2002. Re-registration in the Federal Service for Supervision in the Sphere of Telecom, Information Technologies and Mass Communications (ROSKOMNADZOR) due to change of the founder: «Information and Control Systems», Ltd., JD № FS77-49181 from March, 30, 2012.

The journal is distributed by subscription. Subscription can be made in the Editorial and publishing center, SUAI as well as in any post office based on «Rospechat» catalogue: № 48060 — annual subscript, № 15385 — semiannual subscript.

© Corporate authors, 2014

INFORMATION AND CONTROL SYSTEMS

Viktorov D. S., Chislov S. G. *Method of Correction of the Non-Linear Distortions Entered by an Analog Key in Probing Signals* 2

Turubanov M. A., Shishlakov V. F., Shyshlakov A. V. *Impulse Control System for Combined Solar and Wind Installation with Superconductor Equipment* 8

Zakharova O. L., Kirsanova J. A., Kniga E. V., Zharinov I. O. *Algorithms and Software of Testing Onboard Digital Computer Systems Integrated Modular Avionics* 19

SYSTEM AND PROCESS MODELING

Kuchmin A. Yu. *Modeling of Equivalent Stiffness of Adaptive Platforms with the Parallel Structure Executive Mechanism* 30

HARDWARE AND SOFTWARE RESOURCES

Balonin N. A., Marley V. E., Sergeev M. B. *New Opportunities of the Mathematical Network for Collaborative Research and Modeling in the Internet* 40

Marakhovsky V. B. *CMOS Implementation of the Trainee's Threshold Logical Element. Part I. Design and Training Diagram* 47

Kolchin I. V., Filippov S. N. *The Architecture of Bare-Metal Real-Time Microhypervisor and Automated Measurement of Time Response* 57

Shoshmina I. V. *A Methodology of Eliciting Context Requirements to Program Logic Control Systems* 68

INFORMATION SECURITY

Bezzateev S. V., Voloshina N. V., Sankin P. S. *Safety Analysis Methodology of Complex Systems Taking Into Account the Threats to Information Security* 78

Boyko A. A., Djakova A. V. *Method of Developing Test Remote Information-Technical Impacts on Spatially Distributed Systems of Information-Technical Tools* 84

INFORMATION CODING AND TRANSMISSION

Cheprukov Yu. V., Socolov M. A. *Correlation Characteristics and Application of Some Binary Codes* 93

Alekseev M. O. *On the Detection of Algebraic Manipulations by Means of Multiplication Operation* 103

INFORMATION AND MEASURING SYSTEMS

Allakhverdiyeva N. R. *Development of a Method for Improving the Accuracy of the Measuring Channel* 109

INFORMATION INSTRUMENTATION AND EDUCATION

D'yachuk P. P., Loginov D. A., Karabalykov S. A. *Synergetic Approach to Management of Educational Activity in Verbal Problem Environments* 118

CONTROL IN MEDICAL AND BIOLOGICAL SYSTEMS

Tichonov E. P. *Adaptive Filtering Algorithms Electrocardiogram High Time Resolution Part I. Background Information and Analysis Approach to Solving the Problem* 125

CHRONICLES AND INFORMATION

IV International Forum «TELECOM NETWORKS 2.0. Sharing, Engineering, Outsourcing, Development & Metering» 132

INFORMATION ABOUT THE AUTHORS

134

Submitted for publication 07.04.14. Passed for printing 17.06.14. Format 60×84/8. Offset paper. Phototype SchoolBookC. Offset printing.

Layout original is made at the Editorial and Publishing Center, SUAI.
67, B. Morskaia, 190000, St. Petersburg, Russia
Printed from slides at the Editorial and Publishing Center, SUAI.
67, B. Morskaia, 190000, St. Petersburg, Russia

Учредитель
ООО «Информационно-управляющие системы»

Главный редактор
М. Б. Сергеев,
д-р техн. наук, проф., С.-Петербург, РФ

Зам. главного редактора
Е. А. Крук,
д-р техн. наук, проф., С.-Петербург, РФ

Ответственный секретарь
О. В. Муравцова

Редакционный совет:

Председатель А. А. Оводенко,
д-р техн. наук, проф., С.-Петербург, РФ
В. Н. Васильев,
чл.-корр. РАН, д-р техн. наук, проф., С.-Петербург, РФ
В. Н. Козлов,
д-р техн. наук, проф., С.-Петербург, РФ

К. Кристоделу,
д-р. наук, проф., Альбукерке, Нью-Мексико, США
Б. Мейер,
д-р наук, проф., Цюрих, Швейцария

Ю. Ф. Подоплекин,
д-р техн. наук, проф., С.-Петербург, РФ
В. В. Симаков,
д-р техн. наук, проф., Москва, РФ

Л. Фортуна,
д-р наук, проф., Катания, Италия
А. Л. Фрадков,
д-р техн. наук, проф., С.-Петербург, РФ

Л. И. Чубраева,
чл.-корр. РАН, д-р техн. наук, С.-Петербург, РФ
Ю. И. Шокин,
акад. РАН, д-р физ.-мат. наук, проф., Новосибирск, РФ
Р. М. Юсупов,
чл.-корр. РАН, д-р техн. наук, проф., С.-Петербург, РФ

Редакционная коллегия:

В. Г. Анисимов,
д-р техн. наук, проф., С.-Петербург, РФ
Б. П. Безручко,
д-р физ.-мат. наук, проф., Саратов, РФ
Н. Блаунштейн,
д-р физ.-мат. наук, проф., Беэр-Шева, Израиль

А. Н. Дудин,
д-р физ.-мат. наук, проф., Минск, Беларусь
А. И. Зейфман,
д-р физ.-мат. наук, проф., Вологда, РФ

Г. Н. Мальцев,
д-р техн. наук, проф., С.-Петербург, РФ
В. Ф. Мелехин,
д-р техн. наук, проф., С.-Петербург, РФ

А. В. Смирнов,
д-р техн. наук, проф., С.-Петербург, РФ
В. И. Хименко,
д-р техн. наук, проф., С.-Петербург, РФ

А. А. Шалыто,
д-р техн. наук, проф., С.-Петербург, РФ
А. П. Шепета,
д-р техн. наук, проф., С.-Петербург, РФ

З. М. Юлдашев,
д-р техн. наук, проф., С.-Петербург, РФ

Редактор: А. Г. Ларионова
Корректор: Т. В. Звертановская

Дизайн: А. Н. Колешко, М. Л. Черненко
Компьютерная верстка: Н. Н. Караваева

Адрес редакции: 190000, Санкт-Петербург,
Б. Морская ул., д. 67, ГУАП, РИЦ
Тел.: (812) 494-70-02, e-mail: ius.spb@gmail.com, сайт: http://i-us.ru

Журнал зарегистрирован в Министерстве РФ по делам печати,
телерадиовещания и средств массовых коммуникаций.
Свидетельство о регистрации ПИ № 77-12412 от 19 апреля 2002 г.
Перерегистрирован в Роскомнадзоре.
Свидетельство о регистрации ПИ № ФС77-49181 от 30 марта 2012 г.

Журнал входит в «Перечень ведущих рецензируемых научных журналов и изданий,
в которых должны быть опубликованы основные научные результаты диссертации
на соискание ученой степени доктора и кандидата наук».

Журнал распространяется по подписке. Подписку можно оформить
через редакцию, а также в любом отделении связи по каталогу «Роспечать»:
№ 48060 — годовой индекс, № 15385 — полугодовой индекс.

© Коллектив авторов, 2014

ОБРАБОТКА ИНФОРМАЦИИ И УПРАВЛЕНИЕ

Викторов Д. С., Числов С. Г. Метод коррекции нелинейных искажений, вносимых аналоговым ключом в зондирующие сигналы 2

ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

Турубанов М. А., Шишлаков В. Ф., Шишлаков А. В. Импульсная система управления комбинированной солнечно- и ветроэнергетической установкой со сверхпроводниковым оборудованием 8

Захарова О. Л., Кирсанова Ю. А., Книга Е. В., Жаринов И. О. Алгоритмы и программные средства тестирования бортовых цифровых вычислительных систем интегрированной модульной авионики 19

МОДЕЛИРОВАНИЕ СИСТЕМ И ПРОЦЕССОВ

Кучмин А. Ю. Моделирование эквивалентной жесткости адаптивных платформ с исполнительными механизмами параллельной структуры 30

ПРОГРАММНЫЕ И АППАРАТНЫЕ СРЕДСТВА

Балонин Н. А., Марлей В. Е., Сергеев М. Б. Новые возможности математической сети для коллективных исследований и моделирования в Интернете 40

Мараховский В. Б. КМОП-реализация обучаемого порогового логического элемента. Часть 1: Проектирование и схема обучения **Колчин И. В., Филиппов С. Н.** Архитектура автономного микро-гипервизора реального времени и автоматизированное измерение его временных характеристик 47

Шошмина И. В. Методика составления контекстных требований к программным системам логического управления 57

68

ЗАЩИТА ИНФОРМАЦИИ

Беззатеев С. В., Волошина Н. В., Санкин П. С. Методика расчета надежности сложных систем, учитывающая угрозы информационной безопасности 78

Бойко А. А., Дьякова А. В. Способ разработки тестовых удаленных информационно-технических воздействий на пространственно распределенные системы информационно-технических средств 84

КОДИРОВАНИЕ И ПЕРЕДАЧА ИНФОРМАЦИИ

Чепруков Ю. В., Соколов М. А. Корреляционные характеристики и применение некоторых бинарных R3-кодов 93

Алексеев М. О. Об обнаружении алгебраических манипуляций с помощью операции умножения 103

ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНЫЕ СИСТЕМЫ

Аллахвердиева Н. Р. Разработка метода повышения точности измерительного канала 109

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ОБРАЗОВАНИЕ

Дьячук П. П., Логинов Д. А., Карабалыков С. А. Синергетический подход к управлению учебной деятельностью в вербальных проблемных средах 118

УПРАВЛЕНИЕ В МЕДИЦИНЕ И БИОЛОГИИ

Тихонов Э. П. Адаптивные алгоритмы фильтрации и фрагментации электрокардиограмм высокого временного разрешения. Часть 1: Исходные сведения и анализ подхода к решению проблемы 125

ХРОНИКА И ИНФОРМАЦИЯ

IV Международный Форум «TELECOM NETWORKS 2.0. Sharing, Engineering, Outsourcing, Development & Metering» 132

СВЕДЕНИЯ ОБ АВТОРАХ

134

Сдано в набор 07.04.14. Подписано в печать 17.06.14. Формат 60×84/8.
Бумага офсетная. Гарнитура SchoolBookC. Печать офсетная.
Усл. печ. л. 16,0. Уч.-изд. л. 20,1. Тираж 1000 экз. Заказ 258.

Оригинал-макет изготовлен в редакционно-издательском центре ГУАП.
190000, Санкт-Петербург, Б. Морская ул., 67.

Отпечатано с готовых диапозитивов в редакционно-издательском центре ГУАП.
190000, Санкт-Петербург, Б. Морская ул., 67.

УДК 004-4

МЕТОДИКА СОСТАВЛЕНИЯ КОНТЕКСТНЫХ ТРЕБОВАНИЙ К ПРОГРАММНЫМ СИСТЕМАМ ЛОГИЧЕСКОГО УПРАВЛЕНИЯ

И. В. Шошмина^{а, 1}, старший преподаватель, соискатель

^аСанкт-Петербургский государственный политехнический университет, Санкт-Петербург, РФ

Постановка проблемы: современные методики составления формальных требований к поведению программ не дают инженеру механизмов, позволяющих структурировать разрозненную информацию о будущей системе. Работа посвящена одному из обязательных этапов формальной проверки корректности программных систем логического управления — составлению спецификации требований. **Результаты:** предложена методика составления контекстных требований к поведению программ логического управления при помощи схем целей. Методика строится на упорядочивании темпоральных причинно-следственных связей между событиями системы и контекста ее функционирования. Формально требования задаются на языке линейной темпоральной логики. **Практическая значимость:** методика демонстрируется на примерах разработки требований к двум системам логического управления.

Ключевые слова — реагирующие системы, спецификация требований, линейная темпоральная логика, метод проверки моделей, целеориентированные методики составления требований, схемы целей.

Введение

Программные системы логического управления относятся к классу реагирующих систем [1]. Их основная функция — реагировать на воздействия внешнего окружения: других программных компонент, оборудования или пользователей. Такие системы характеризуются бесконечным поведением, а не преобразованием входных данных в выходные. К настоящему времени формальная проверка корректности реагирующих систем достигла впечатляющих успехов: разработаны формальные основы доказательства корректности; реализованы программные пакеты верификации; крупные разработчики программных систем включили совокупность методов верификации, называемых «Model checking» (проверка модели), в технологический цикл проектирования программ.

В то же время проблема выбора, формирования множества тех требований к поведению реагирующей программы, которые необходимо проверять, остается открытой. Эта проблема является неформальной: нельзя построить алгоритм, генерирующий требования к поведению программы из технического задания на ее разработку. Какие требования к реагирующей программе должен сформировать разработчик для доказательства ее корректности? Из интуитивных соображений естественно предположить, что формально должны проверяться требования к поведению программы, гарантирующие в совокупности абсолютно правильное ее функционирование. Однако известно, что, за исключением

тривиальных случаев, невозможно выразить формально требование «полной, абсолютной правильности» функционирования программы вследствие того, что это понятие не формализуемо. Это обстоятельство соответствует общему принципу, согласно которому не может быть выполнена никакая полная формальная проверка корректности алгоритмов. Следует, однако, констатировать, что в настоящее время не существует приемлемой инженерной методики спецификации требований к поведению систем логического управления, помогающей разработчику принять решение.

В статье рассматривается проблема составления и формализации требований к программным системам логического управления. Мы предполагаем первичность требований: требования строятся до того, как разработана программная система.

Любая методика составления требований к системам логического управления должна структурировать информацию об их поведении. Существующие классификации формул темпоральной логики [2–4] направлены на выявление классов формул, исключающих ошибки, типичные для взаимодействующих параллельных процессов (безопасность, живость, отсутствие блокировок, справедливость). Такие классификации не дают ответа на вопрос, как формировать требования, отражающие функционирование реагирующей системы, из технического задания.

В практических рекомендациях предлагается выбирать требования по функциональным иерархиям, построенным из общих соображений. Например, в работе [5] функциональную иерархию составляют разделы: информационные потоки, описание процессов, конструкции данных, словарь данных. Такое обобщение задает лишь направление выбора требований, но не дает механизмов их построения.

¹ Научный руководитель — доктор технических наук, заведующий кафедрой распределенных вычислений и компьютерных сетей Санкт-Петербургского государственного политехнического университета Ю. Г. Карпов.

Предлагаемое в работе решение основано на следующем наблюдении. Задачей реагирующей системы является поддержание шаблонов взаимодействия с окружением в том контексте, в котором функционирует система. Взаимодействие системы с ее окружением в общем случае может быть разнообразным. Рассмотрим класс требований к поведению реагирующей системы, определяемый темпоральными отношениями причины и следствия между событиями. Этот класс требований назван контекстным.

Цель данной работы — разработать методику, позволяющую выразить контекстные требования к поведению реагирующей системы на языке линейной темпоральной логики (Linear Temporal Logic — LTL) посредством анализа темпоральных отношений причины и следствия между всеми событиями, инициируемыми системой и ее окружением.

На практике требования не удается построить сразу, поскольку приходится структурировать большой объем слабо систематизированной информации о будущей программной системе. В качестве основы используется одна из методик составления требований к программам — методика схем целей (problem frame approach), разработанная М. Джексоном [6]. С ее помощью требования к программе строятся из технического задания на разработку программы и всех других возможных источников итеративно, путем систематизации отношения между системой и объектами ее окружения. Методика схем целей была изначально разработана для выявления статических отношений, типичных для программ обработки данных. Ее использование для формирования темпоральных требований к реагирующим программам требует модификации. Именно модификация методики М. Джексона в целях формирования требований к поведению реагирующих программ составляет предмет данной работы.

Модифицированная методика опробована на примерах разработки спецификации требований к системе управления лифтом и системе управления энергоснабжением судна.

Методика схем целей и темпоральные требования

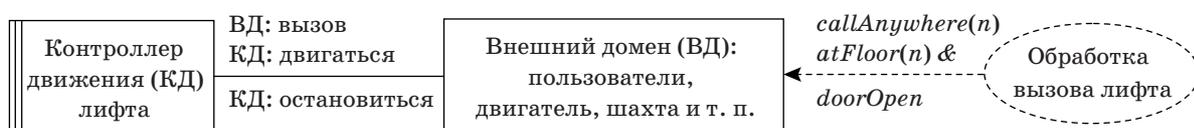
Идеи методики схем целей основываются на принципах, сформулированных Э. Хоаром и Ж.-Р. Абраилем при разработке Z-спецификаций.

Приведем их. Разрабатываемая программа есть (будущий) объект реального мира, в котором она должна реализовать некие цели. Чтобы определить эти цели, мы должны изучить будущую программу с разных ракурсов, используя техническое задание и информацию из других источников, подобно тому, как мы разглядываем новый предмет.

Информация о каждой цели собирается в одной схеме. Язык схем является графическим. В схему включаются объекты реального мира, взаимодействующие при реализации цели, а также феномены будущей программной системы (события, переменные, константы, отношения), разделяемые во время взаимодействия. Для того чтобы цель была реализована программой, к ней необходимо предъявить соответствующее требование. Требование также указывается на схеме. Разработка требования начинается с составной схемы, отражающей общие представления о цели, потом последовательно составная схема и соответствующее требование уточняются, насыщаются деталями. Уточнение завершается, если схема приведена к одной из так называемых элементарных. М. Джексон выделил пять целей, к которым, с его точки зрения, сводится большинство целей программ. Эти цели отражают шаблоны отношений, реализуемых программами чаще всего. Именно такие цели описываются элементарными схемами. Например, элементарная схема TRM определяет функцию преобразования системой заданных входных данных. Остальные элементарные схемы можно найти, например, в работе [6]. Множество типов таких целей М. Джексон оставил открытым.

Чтобы продемонстрировать возможность определения темпоральных отношений между событиями с помощью методики М. Джексона, зададим цель обработки события вызова лифта системой управления лифтом на языке схем целей. Вызов поступает из кабины лифта или с площадки этажа. Схема этой цели в терминологии М. Джексона представлена на рис. 1. Требование обработки вызова можно сформулировать так: «Всегда, если поступил вызов, то лифт когда-нибудь придет на данный этаж и остановится».

На схемах требование формулируется коротко на естественном языке в овале с пунктирной границей. Требование задается к главному объекту схемы — «машине». Машиной М. Джексон



■ Рис. 1. Схема цели обработки вызова

назвал функцию будущей программной системы в соответствии с данной целью. Машина задается на схеме прямоугольником с двумя вертикальными чертами (*Контроллер движения лифта*). Машина не обязана реализовываться как отдельный модуль системы. Чтобы реализовать цель, машина взаимодействует с другими объектами внешнего мира. Они определяются на схеме в прямоугольниках. На рис. 1 к объектам, кроме машины, относится и *Внешний домен*, объединяющий пользователей, двигатель, шахту лифта, этажи здания.

При реализации цели объекты связываются друг с другом по интерфейсам. Интерфейсы изображаются на схеме прямыми сплошными линиями. Через интерфейсы объекты обмениваются феноменами. Феноменами в методике М. Джексона могут быть статические отношения, параметры, константы, события. Феномены определяются на интерфейсах надписями в формате <Объект:><феномен>. На рис. 1, например, «КД: двигаться» означает, что «Контроллер движения» дает команду двигателю лифта начать движение.

На пунктирных прямых линиях, связывающих объекты и требования, установлены конкретные значения параметров, которые входят в развернутую формулировку требования. Пунктирная линия со стрелкой направлена к объекту, который ограничивается требованием. Например, на рис. 1 требование определено относительно следующих булевых переменных: *callAnywhere(n)* — этаж *n* вызван (с площадки или из кабины лифта), *atFloor(n)* — лифт находится на этаже *n*, *doorOpen* — дверь лифта открыта. Развернутое требование к ограничению пропуски этажа хотелось бы задать следующим образом: «Всегда, если поступил вызов *callAnywhere(n)*, то когда-нибудь в будущем лифт доедет *atFloor(n)* до этого этажа и откроет двери *doorOpen*».

Согласно оригинальной методике, схема на рис. 1 уже есть элементарная схема Required Behaviour (RBV). Элементарная схема RBV определяет требуемое поведение системы перечислением множества событий (на рис. 1 — *callAnywhere(n)*, *atFloor(n)* и *doorOpen*). Разные темпоральные отношения, заданные на одинаковом множестве событий, не будут различаться на схемах целей по методике М. Джексона. Он рекомендует такие отношения определять отдельно, вне схем, другими средствами, например системой переходов [6].

Далее в этой работе методика схем целей будет модифицирована так, чтобы с ее помощью задавать контекстные темпоральные требования. В частности, мы будем использовать в качестве феноменов только события и цепочки событий, а также фиксировать темпоральные отношения между событиями.

Определение контекстных требований на языке LTL

Сформируем контекстные требования к поведению программной реагирующей системы как требования к причинно-следственным отношениям между системой и окружением.

Пропозициональная LTL является расширением классической логики высказываний. Ее формулы строятся на множестве атомарных высказываний $p \in AP$ при помощи булевых операций и темпоральных операторов U (Until), X (NextTime) по следующей грамматике:

$$\phi ::= p \mid \neg\phi \mid \phi \vee \psi \mid X\phi \mid \phi U \psi.$$

Кроме того, для сокращения записи вводятся дополнительные темпоральные операторы — оператор будущего F и глобальный оператор G :

$$F\phi = \top U \phi; \quad G\phi = \neg F\neg\phi,$$

где $\top = \text{true}$.

Семантика формул LTL задается на бесконечных словах [1], представляющих вычисления реагирующей системы.

Определим два типа темпоральных отношений причины и следствия — отношения отклика и предшествования. Рассмотрим сначала элементарные отношения, когда причиной-стимулом является одно событие и следствием-реакцией также одно событие. Мы будем отслеживать источники событий при формировании отношений и требований. Важно, чтобы события контекста входили либо только в причину, либо только в следствие.

Отношение, когда по стимулу s контекста когда-нибудь в будущем система должна обеспечить реакцию p , назовем отношением отклика. Оно формализуется на языке LTL так:

$$Resp(s, p) = s \rightarrow Fp.$$

Отношение отклика часто встречается в требованиях к программным системам логического управления, например: «Всегда, если пользователь вызвал лифт, то когда-нибудь в будущем лифт прибудет на этаж вызова».

Отношение отклика становится нереализуемым, если переставить источники стимула и реакции: если источником стимула выступает система, а источником реакции — окружение, то такое отношение отклика требует гарантии специфического поведения окружения. Но система не может «заставить» окружение выполнить то или иное действие.

Рассмотрим другое отношение — отношение предшествования: если когда-нибудь в будущем

возникнет реакция p , то до этого момента возник стимул s . Этот тип отношения на языке LTL формулируется так:

$$Prec(s, p) = \neg p \wedge Fp \rightarrow \neg p U s.$$

Событие контекста может участвовать в отношении предшествования либо как стимул, либо как реакция.

Для единообразного масштабирования и общности формул будем ограничивать появление стимула в отношениях отклика специальным событием t и появление реакции в том же отношении при помощи события v . В отношениях предшествования при помощи события t ограничивается только появление реакции, в частности, во введенных ранее элементарных отношениях $v = T$ и $t = T$. Зафиксируем это явно, переписав формулы элементарных отношений:

$$Resp(s, p, t, v) = s \rightarrow v U p;$$

$$Prec(s, p, t) = \neg p \rightarrow (t U p \rightarrow \neg p U s).$$

Далее, давая примеры возможных формулировок требований на естественном языке, будем для простоты полагать $t = T$, $v = T$.

Чтобы сформировать контекстные требования, необходимо учесть отношения отклика и предшествования в структурах взаимодействий, отражающих различные фазы функционирования программных систем логического управления. Такие фазы выбраны из практики работы систем, например, их можно найти в работах [7, 8].

Наиболее часто встречается глобальная фаза: отношение должно гарантироваться на протяжении всей работы системы в каждом состоянии. Также часто распространена регулярная фаза, когда отношение необходимо требовать каждый раз в некотором режиме работы системы, ограниченном специфическими событиями. Начальная и финальная фазы появляются реже. Типичным примером начальной фазы является следующая ситуация. При запуске системы пользователю требуется установить некоторые настройки, которые во время работы системы не меняются. В финальной фазе система переходит по некоторым параметрам в невозвратный режим и в этом режиме должна особым образом обрабатывать события контекста.

При задании контекстных требований каждой фазы на языке LTL мы абстрагируемся от семантики и сосредоточимся на формальном описании взаимодействий с помощью последовательностей событий.

Требование к глобальной фазе задает выполнение отношения отклика или предшествования в каждом состоянии поведения системы. На естественном языке требование к глобальной фазе для

отношения отклика можно описать так: «Всегда, если появился стимул контекста, то когда-нибудь в будущем возникнет реакция системы»; для отношения предшествования: «Всегда реакция возникает непосредственно после появления стимула». Например, приведенное выше требование обработки лифтом вызовов должно выполняться в глобальной фазе. Формализуются требования глобальной фазы следующим образом:

$$global(s, p, t, v; \varphi) = G \varphi(s, p, t, v).$$

Здесь и далее в формулах фаз формула φ — это формула либо $Resp$, либо $Prec$. Чтобы первый блок событий не состоял из непрерывного блока реакций, формулу глобальной фазы с отношением предшествований необходимо умножить на $\neg p$.

Требование выполнения отношения отклика (предшествования) в финальной фазе можно сформулировать так: «После первого события, определяющего финальную фазу, всегда гарантируется, что по стимулу когда-нибудь в будущем наступит реакция», или формально

$$fin(s, p, t, v; q; \varphi) = F q \rightarrow \neg q U (q \wedge global(s, p, t, v; \varphi)),$$

где событие q фиксирует начало финальной фазы.

Требование отношения отклика в начальной фазе естественно сформулировать так: «Каждый стимул, поданный до конца начальной фазы, будет когда-нибудь обработан»; а требование с отношением предшествования: «Каждой реакции, поданной до конца начальной фазы, непосредственно предшествует стимул». Запишем эти требования формально:

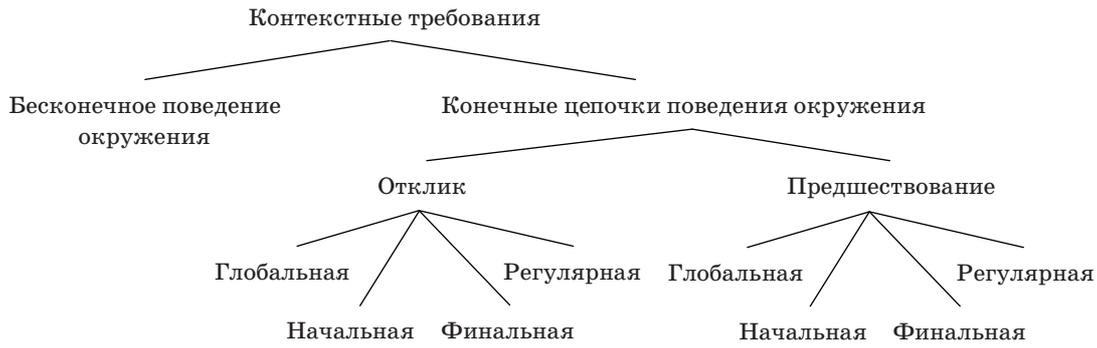
$$start(s, p, \neg r \wedge t, v; r; \varphi) = \varphi(s, p, \neg r \wedge t, v) U r,$$

где событие r фиксирует завершение начальной фазы. Чтобы избежать первого блока событий реакций, формулу с отношением предшествования также необходимо предварить $\neg p$. В начальной фазе влияние ограничения событием t становится существенным. При формировании контекстного требования начальной фазы к исходному ограничению t добавляется ограничение наличия начальной фазы $\neg r$.

Требование отношения отклика в регулярной фазе будет таково: «Всегда, если установлено событие q и когда-нибудь в будущем произойдет событие r , то на каждый стимул, поступивший до события r , когда-нибудь будет выполнена реакция». Запишем LTL-формулу требования для обоих отношений:

$$reg(s, p, \neg r \wedge t, v; q, r; \varphi) =$$

$$= G \left(q \rightarrow \left(F r \rightarrow start(s, p, \neg r \wedge t, v; r; \varphi) \right) \right),$$



■ Рис. 2. Классификация контекстных требований

где начало и завершение режима определяются событиями q и r соответственно.

Эти требования построены в предположении, что переменные s, p, q, r, t, v независимы. В общем случае соответствующие события могут задаваться булевыми выражениями и быть зависимыми. Тогда для проверки адекватности требований обязателен дополнительный анализ.

Определенные выше требования заданы относительно конечных цепочек событий с отношениями отклика и предшествования. Требования, формируемые в условиях бесконечного поведения контекста, должны быть представлены формулами LTL следующего вида:

$$\Psi \rightarrow \Phi,$$

где Φ задает одно из контекстных требований к конкретной фазе функционирования системы, определенной выше, а Ψ характеризует требование к бесконечному поведению контекста. В частности, известные классические требования с условиями справедливости в нашей терминологии являются требованиями с бесконечным поведением контекста.

В рамках предложенной структуры контекстные требования легко масштабируются на случай более сложных отношений отклика и предшествования, когда стимул и (или) реакция задаются цепочками событий: $s_1, s_2, \dots, s_m, p_1, p_2, \dots, p_n$ с соответствующими ограничениями t_i, v_j . Для этого нужно определить расширенный вариант отношений отклика и предшествования на языке LTL:

$$\begin{aligned} \Psi_1(s_1, \dots, s_m, t_1, \dots, t_m) &= \\ &= s_1 \wedge t_2 U(\dots s_{m-1} \wedge (t_m U s_m) \dots); \\ \Psi_2(p_1, \dots, p_n, v_1, \dots, v_n) &= \\ &= v_1 U(p_1 \wedge \dots v_{n-1} U(p_{n-1} \wedge v_n U p_n) \dots); \\ \Psi_3(s_1, \dots, s_m, p_1, \dots, p_n, t_1, \dots, t_n) &= \\ &= t_1 U(p_1 \wedge t_2 U(\dots p_{n-1} \wedge (t_n U p_n) \dots)) \rightarrow \\ &\rightarrow \neg p_1 U(s_1 \wedge \neg p_1 U(\dots (\neg p_1 U s_m) \dots)); \end{aligned}$$

$$\begin{aligned} Resp(s_1, \dots, s_m, p_1, \dots, p_n, t_1, \dots, t_m, v_1, \dots, v_n) &= \\ &= \Psi_1(s_1, \dots, s_m, t_1, \dots, t_m) \rightarrow \\ &\rightarrow t_2 U(s_2 \wedge \dots (t_m U(s_m \wedge \Psi_2(p_1, \dots, p_n, v_1, \dots, v_n))) \dots); \\ Prec(s_1, \dots, s_m, p_1, \dots, p_n, t_1, \dots, t_n) &= \\ &= \neg p_1 \rightarrow \Psi_3(s_1, \dots, s_m, p_1, \dots, p_n, t_1, \dots, t_n). \end{aligned}$$

Формулы контекстных требований задаются по шаблонам, указанным выше, в соответствии с конкретной фазой эксплуатации системы. Классификация введенных контекстных требований на языке LTL представлена на рис. 2.

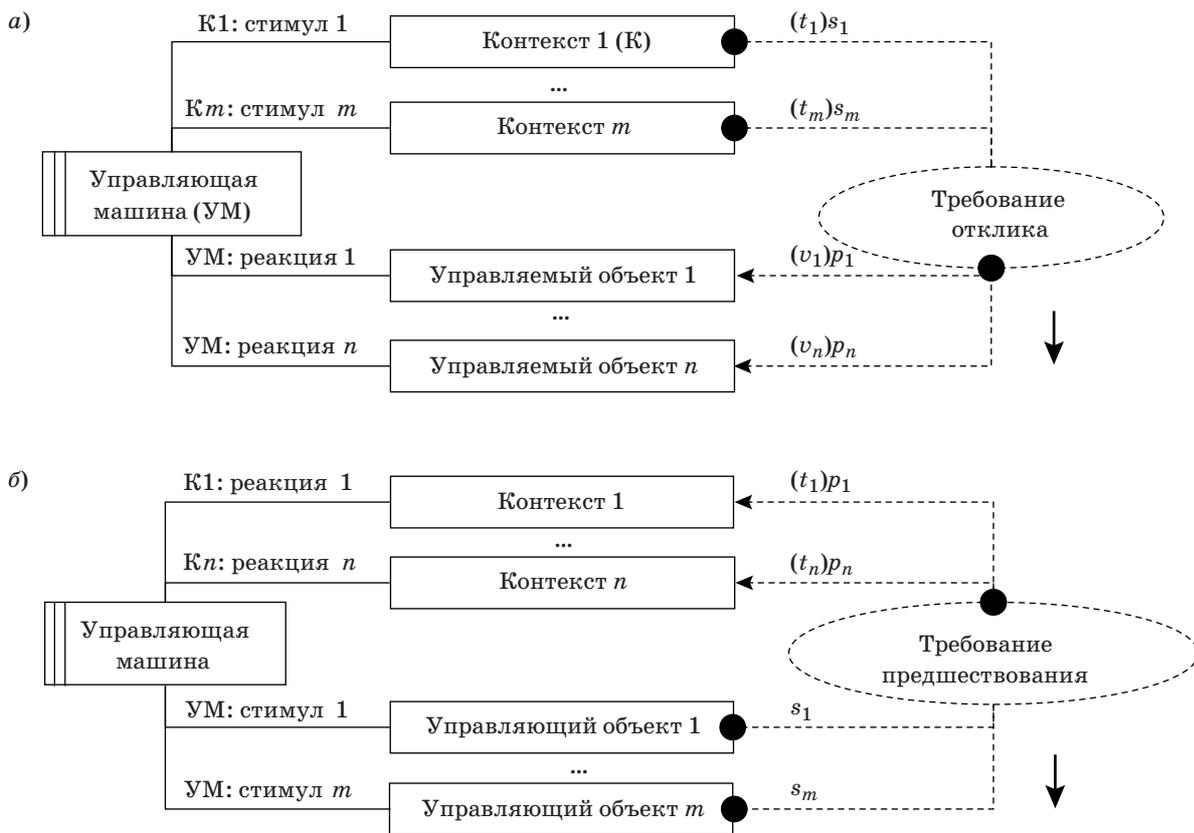
Модификация методики схем целей для формирования контекстных требований к реагирующим системам

В этом разделе будут разработаны элементарные схемы контекстных требований и определена процедура их формирования.

Цели, реализуемые контекстными требованиями (см. рис. 2), идентичны. Общую их структуру можно сформулировать так: «Система должна гарантировать выполнение (некоторого) отношения в (конкретной) фазе функционирования».

Чтобы упростить методику составления требований для таких целей, разобьем их на две взаимосвязанные цели. Цель отношения устанавливает необходимое отношение: «Система должна гарантировать выполнение (некоторого) отношения». Другая цель устанавливает контекстное требование: «Система должна гарантировать цель отношения в (конкретной) фазе функционирования». В соответствии с этими целями построим два типа элементарных схем.

Схемы отношений, следуя методике определения целей М. Джексона, будем представлять графически, однако семантика графических элементов и связей будет другой. Элементарные схемы отношений представлены на рис. 3, а, б. Объекты, инициирующие события-стимулы, связаны с требованием пунктирной прямой, а реак-

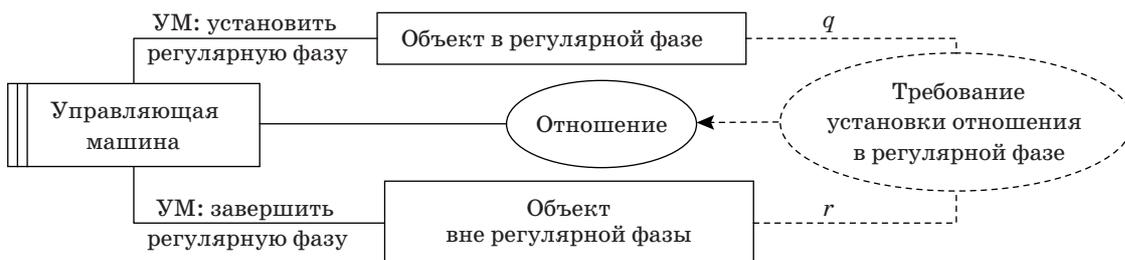


■ Рис. 3. Элементарные схемы отношений: а — отклика; б — предшествования

ции — прямой со стрелкой. Чтобы различать схемы отношений отклика от схем отношений предшествования, установлен порядок объектов, инициирующих стимулы и реакции. На графической схеме отношения отклика блок с объектами, инициирующими стимул, находится выше блока с объектами, инициирующими реакцию. На схеме отношения предшествования порядок следования блоков противоположный (см. рис. 3, б). Чтобы задавать последовательности стимулов s_1, \dots, s_m , введены точки сбора цепочки событий. Порядок следования стимулов определяется порядком следования объектов в соответствующем блоке сверху вниз. Если стимул s_k в цепочке стимулов следует раньше s_i , то объект, инициирующий

стимул s_k , расположен на схеме выше объекта, инициирующего s_i . Для реакций введены аналогичные обозначения.

Элементарные схемы контекстных требований задаются для определенных выше фаз функционирования. Эти схемы — иерархические, т. е. они ссылаются на элементарные схемы отношений. Ссылка обозначается овалом со сплошной границей (рис. 4). Иерархические схемы были использованы М. Джексоном в работе [6], хотя впоследствии не применялись. На элементарных схемах контекстных требований устанавливается порядок следования событий, определяющих фазы. На рис. 4 приведена схема для регулярной фазы, остальные строятся аналогично.



■ Рис. 4. Элементарные схемы контекстных требований выполнения отношения в регулярной фазе

Событие начала фазы q предшествует отношению, фаза завершается событием r , такой порядок восстанавливается по порядку расположения прямоугольников объектов, инициирующих соответствующие события, и ссылки на отношение (сверху вниз).

Теперь определим процедуру формирования контекстных требований при помощи схем целей для отношения отклика (для отношения предшествования процедура аналогична).

1. Когда в техническом задании либо в прочей доступной информации обнаруживается некоторое событие окружения, влияющее на поведение системы, то необходимо исследовать, не является ли оно реакцией или стимулом для действий системы.

2. Если событие окружения является стимулом и есть реакция системы, то необходимо зафиксировать цель гарантии соответствующего отношения отклика в элементарной схеме (см. рис. 3, а). Дополнительно отношение должно быть исследовано на влияние цепочки стимулов или реакций.

3. Необходимо проанализировать, в какой из фаз функционирования система должна гарантировать полученное отношение. Результат анализа фиксируется в элементарной схеме рабочей фазы (см. рис. 4).

4. По полученным элементарным схемам определяется LTL-формула, выражающая соответствующее контекстное требование.

Если согласно предложенной процедуре не удалось установить контекстное требование, то следует действовать по оригинальной методике схем целей. Одним из частных случаев, не описываемых ни одной из предложенных элементарных схем, будет требование с бесконечным поведением окружения.

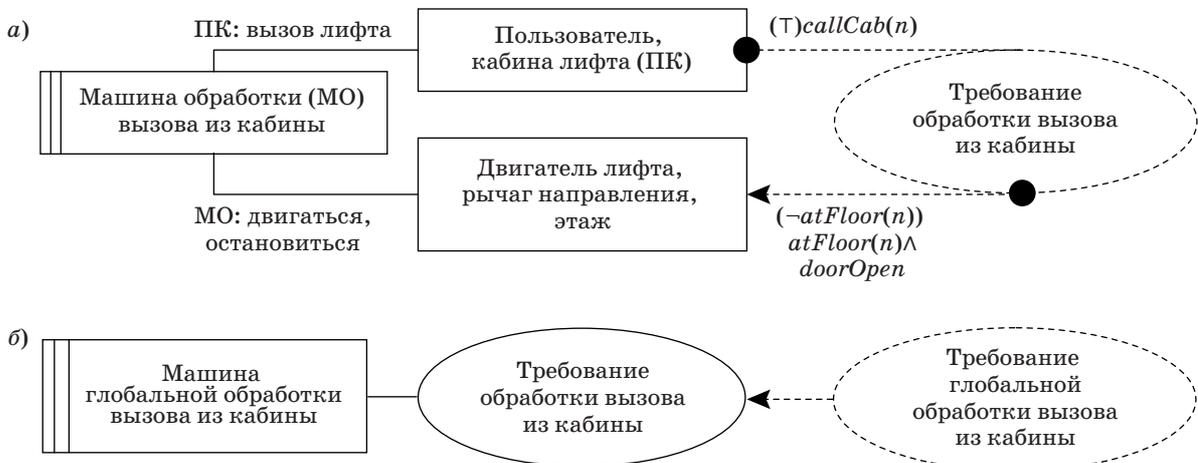
Продемонстрируем уточнение целей и построение LTL-формулы контекстного требования по

схеме для цели обработки вызовов (см. рис. 1) системы управления лифтом. Событие «ВД: вызов» на интерфейсе между «Внешним доменом» и «Контроллером движения» подается окружением. Оно стимулирует систему начать обслуживание, т. е. инициирует отношение отклика как стимул. Простейшей реакцией системы было бы прибытие лифта на этаж. Однако при поступлении вызова из кабины лифта эта реакция может быть уточнена.

Именно нажав кнопку этажа в кабине лифта, пользователь предполагает, что лифт, подъезжая к этому этажу, остановится, не проехав мимо. То есть реакция системы управления на вызов из кабины состоит в последовательности событий «лифт не на этаже вызова» до тех пор, пока наступят события «лифт на этаже вызова, и двери открыты». Отношение отклика должно выполняться в каждом состоянии системы. На рис. 5 определены схемы обработки вызова из кабины лифта для отношения отклика (рис. 5, а) и для глобальной фазы (рис. 5, б) с учетом переменных, введенных ранее.

Получим LTL-формулу по схеме на рис. 5. Элементарная схема отношения отклика на рис. 5, а содержит один стимул $callCab(n)$ между объектом «Пользователь, кабина лифта» и требованием, и одну реакцию $atFloor(n) \wedge doorOpen$ между требованием и объектом «Двигатель лифта». Реакция ограничена последовательностью событий $\neg atFloor(n)$. Таким образом, имеются все данные для построения LTL-формулы отношения отклика $Resp(callcab(n), atfloor(n) \wedge dooropen, T, \neg atfloor(n))$. Поскольку отношение задано в глобальной фазе (см. рис. 5, б), то требование обработки вызова из кабины лифта задается формулой *global*:

$$G(callCab(n) \rightarrow \rightarrow \neg atFloor(n) U (atFloor(n) \wedge doorOpen))$$



■ Рис. 5. Схема: а — отношения отклика для обработки вызова; б — глобальной обработки вызова из кабины лифта

Представленная в этом разделе модификация методики схем целей позволяет решить задачу систематизации построения формальных темпоральных требований к причинным зависимостям программных систем логического управления.

Сравнительный анализ контекстных требований

Рассмотрим, как контекстные требования соотносятся с другими шаблонами темпоральных взаимодействий, описанными в литературе, а также оценим влияние контекстных требований на корректность двух реализованных систем с известным техническим заданием.

Для апробации описанной выше методики составления контекстных требований были проанализированы системы управления лифтом и энергоснабжением судна [7, 9]. Они выбраны потому, что для них были доступны как техническое задание, так и полный текст программного кода, реализующего системы управления. Программный код тщательно протестирован разработчиками этих систем. Система управления лифтом использовалась Д. Кнудом для обучения разработке сопрограмм, для нее разработан интерпретатор. Система управления энергоснабжением судна была установлена на вновь построенное судно «Вайгач», которое отправилось на ходовые испытания в Белое море, поэтому имелась возможность проверить адекватность найденных нами ошибок в коде программ.

Структура спецификации контекстных требований рассматриваемых систем такова. Из 13 контекстных требований к системе управления лифтом одно требование включает условие с бесконечной цепочкой контекста, тогда как основная часть требования не задает связи с контекстом, 3 требования задают отношение предшествования, а 9 — отношение отклика.

При составлении требований к системе управления энергоснабжением судна [7] из 36 контекстных требований 16 требований — это контекстные требования с отношением предшествования, 20 требований — контекстные требования с отношением отклика. В цепочки событий в требованиях к системе управления энергоснабжением судна входило до 9 событий-реакций (стимулов). Все требования с отношением отклика включают условие с бесконечными цепочками контекста. В результате LTL-формулы требований содержали до 10 условий справедливости.

По программным кодам каждой из двух систем управления были построены модели на языке Promela и произведена верификация с помощью верификатора SPIN [10] относительно разработанных спецификаций. В результате проверки удалось выявить несколько ошибок.

В системе управления лифтом одна ошибка связана с нарушением контекстных требований отклика, в частности, обработки вызова. Ошибка является следствием того, что в некоторых ситуациях лифт не может покинуть этаж вызова. Эта ошибка демонстрирует невозможность применить разработанную Д. Кнудом программу [9] в качестве реальной системы управления лифтом. Для обучения разработке сопрограмм в код модели лифта в работе [9] искусственно введены такие ограничения на последовательности событий, которые в реальности из-за недетерминированности чередования процессов реализовать невозможно.

В системе управления энергоснабжением судна при верификации найдено три критические ошибки. Одна из ошибок связана с нарушением требования выхода из защищенного режима. В результате очень редкого возможного поведения датчиков двигателя система бесконечно часто переключается из рабочего в защищенный режим, полностью блокируя управление. Для судна «Вайгач» ошибки были подтверждены в процессе ходовых испытаний, которые проводились параллельно и независимо от формальной верификации системы управления энергоснабжением.

Отличие контекстных требований от других работ, также определяющих шаблоны темпоральных отношений реагирующих систем, состоит в отсутствии у последних семантики функционального назначения этих систем.

В работе [11] впервые предложена классификация шаблонов темпоральных взаимодействий с практическими целями. Классификация [11] составлялась, чтобы решить задачу перехода от описания темпоральных требований на естественном языке в формулы LTL, STL и в регулярные выражения. Эти шаблоны выбирались исходя из наиболее часто встречающихся структур требований в доступных источниках. Привязка требований к функциональности системы в работе [11] не проводилась. Тем не менее значение этой работы достаточно велико. Большинство из последующих работ по практическим классификациям темпоральных требований, например [12–14], лишь расширяют результаты работы [11] (для вероятностных, количественных и прочих зависимостей).

Если контекстное требование нашей классификации будет задано на естественном языке в соответствии с шаблонами [11, 15], то оно может быть транслировано в LTL-формулу по методике работы [11]. Однако некоторые контекстные требования не могут быть определены при помощи шаблонов [11], в частности, если длина цепочки событий реакции или контекста более двух. Отметим также, что работа [11] содержит несколько ошибок трансляции словесной формулировки

требований в формулы темпоральной логики. Если такую трансляцию проводить по предложенной в данной работе методике формирования контекстных требований, то ошибок можно избежать. Из 217 требований, рассмотренных в работе [11], 83 требования относятся по структуре связей к классу контекстных требований с отношением отклика, 13 требований — к классу контекстных требований с отношением предшествования.

Заключение

В работе предложен новый класс требований к поведению реагирующих программных систем — класс контекстных требований, определяемых через отношения отклика и предшествования между событиями, возникающими в разрабатываемой программной системе и ее

окружении. Контекстные требования формально определены на языке LTL. Для структуризации информации об отношениях при составлении контекстных требований модифицирована известная методика схем целей, предложенная М. Джексоном. По предложенным элементарным схемам контекстных требований можно построить LTL-формулы контекстных требований.

Сравнительный анализ показал, что контекстные требования структурно являются расширением наиболее употребляемых шаблонов темпоральных требований, описанных в литературе.

Предложенная методика позволяет систематически составлять контекстные требования из технического задания и дополнительной информации о разрабатываемой системе. Как показал опыт верификации двух программных систем управления, невыполнение таких требований может быть критическим для приложений.

Литература

1. Карпов Ю. Г. Model checking. Верификация параллельных и распределенных программных систем. — СПб.: БХВ-Петербург, 2010. — 560 с.
2. Lamport L. Proving the Correctness of Multiprocess Programs // IEEE Trans. Softw. Eng. 1977. Vol. 3. N 2. P. 125–143.
3. Manna Z., Pnueli A. A Hierarchy of Temporal Properties // Proc. of the Ninth Annual ACM Symp. on PODC. 1990. P. 377–410.
4. Cerná I., Pelánek R. Relating Hierarchy of Linear Temporal Properties to Model Checking // Proc. of MFCS'2003. LNCS. 2003. Vol. 2747. P. 318–327.
5. IEEE Std 830-1998. IEEE Recommended Practice for Software Requirements Specifications. — N. Y.: IEEE, Inc. 1998. — 31 p.
6. Jackson M. A. Problem Analysis Using Small Problem Frames // South African Computer J. 1999. Vol. 22. P. 47–60.
7. Система КСУ ТС. Руководство по эксплуатации КЛГИ.657122.001 РЭ / ЦНИИ СЭТ. — СПб., 2008. — 41 с.
8. Alspaugh T. A. et al. Software Requirements for the A-7E Aircraft // NRL Report. 1989. N 9194. — 450 p.
9. Кнут Д. Искусство программирования. Т. 1. Основные алгоритмы. — М.: Вильямс, 2010. — 720 с.
10. Verifying Multi-threaded Software with Spin. <http://spinroot.com/spin/whatispin.html> (дата обращения: 21.12.2013).
11. Dwyer M. B., Avrunin G. S., Corbett J. C. Patterns in Property Specifications for Finite-State Verification // Proc. of ICSE '99. P. 411–420.
12. Konrad S., Cheng B. H. C. Real-Time Specification Patterns // Proc. of ICSE '05. P. 372–381.
13. Post A., Menzel I., Podelski A. Applying Restricted English Grammar on Automotive Requirements: Does it Work? A Case Study // Proc. of the REFSQ'11. 2011. P. 166–180.
14. Ramezani E., Fahland D., van Dongen B. F., van der Aalst W. M. P. Diagnostic Information for Compliance Checking of Temporal Compliance Requirements // Proc. of the CAiSE. 2013. P. 304–320.
15. Property Pattern Mappings for LTL. <http://patterns.projects.cis.ksu.edu/documentation/patterns/ltl.shtml> (дата обращения: 21.12.2013).

UDC 004-4

A Methodology of Eliciting Context Requirements to Program Logic Control Systems

Shoshmina I. V.^a, Professor Assistant, ishoshmina@den.icc.spbstu.ru

^aSaint-Petersburg State Polytechnical University, 21, Polytechnicheskaya St., 194021, Saint-Petersburg, Russian Federation

Purpose: Modern approaches to eliciting formal requirements to software behavior do not provide enough ways to structure incomplete information about the future system. The paper deals with one of the essential stages in formal verification of software logical control systems, namely elicitation of temporal requirements. **Results:** We proposed a technique for eliciting some temporal

requirements to logical control software behavior, by modifying problem frames approach. This technique is based on arranging the temporal relations between events within the system and its operational context. In particular, we considered chains of stimuli and reactions depending on a source which induces them. In the paper, these temporal relations are called context requirements. Formally, context requirements have been described in terms of linear temporal logic. We extended the problem frame notation to define ordered event chains and event blocks, and also added two elementary problem frames, which are temporal relation frame and operational phase frame. The temporal relation frame can specify problems which require that the system answers by a reactions chain to a stimulus chain locally. The operational phase frame extends the local behavior determined by the temporal relation frame to the global scope, depending on the system phase restrictions. Our elementary problem frames can be translated into linear temporal logic formulas. The traditional problem frame development procedure has been modified to find out context requirements. **Practical relevance:** The technique has been demonstrated on the examples of eliciting requirement specifications to two logical control systems (an elevator controller and a vessel power controller).

Keywords — Reactive Systems, Requirement Specification, Linear Temporal Logic, Model Checking, Goal-Oriented Methods of Requirement Development, Problem Frame Approach.

References

1. Karpov Yu. G. *Model checking. Verifikatsiia parallel'nykh i raspredeleennykh programmykh sistem* [Model checking. Verification of Parallel and Distributed Software Systems]. Saint-Petersburg, BKhHV-Peterburg Publ., 2010. 560 p. (In Russian).
2. Lamport L. Proving the Correctness of Multiprocess Programs. *IEEE Transactions on Software Engineering*, 1977, vol. 3, no. 2, pp. 125–143.
3. Manna Z., Pnueli A. A Hierarchy of Temporal Properties. *Proc. of the Ninth Annual ACM Symposium on PODC*, 1990, pp. 377–410.
4. Cerná I. and Pelánek R. Relating Hierarchy of Linear Temporal Properties to Model Checking. *Proc. of the MFCS'2003*, LNCS, 2003, vol. 2747, pp. 318–327.
5. IEEE Std 830-1998. IEEE Recommended Practice for Software Requirements Specifications. New York, IEEE, Inc., 1998. 31 p.
6. Jackson M. A. Problem Analysis Using Small Problem Frames. *South African Computer Journal*, 1999, vol. 22, pp. 47–60.
7. *Sistema KSU TS. Rukovodstvo po Eksploatatsii KLG1.657122.001RE* [System ICS TF. Operating Instruction]. TsNII SET Publ., 2008. 41 p. (In Russian).
8. Alspaugh T. A., Faulk S. R., Britton K. H., Parker R. A., Parnas D. L., Shore J. E. Software Requirements for the A-7E Aircraft. *NRL Report*, 1989, no. 9194. 450 p.
9. Knuth D. E. *The Art of Computer Programming*. Vol. 1. *Fundamental Algorithms*. Addison-Wesley Longman, Inc., 1998. 672 p.
10. *Verifying Multi-Threaded Software with Spin*. Available at: <http://spinroot.com/spin/whatispin.html> (accessed 21 December 2013).
11. Dwyer M. B., Avrunin G. S., Corbett J. C. Patterns in Property Specifications for Finite-State Verification. *Proc. of the ICSE '99*, 1999, pp. 411–420.
12. Konrad S., Cheng B. H. C. Real-Time Specification Patterns. *Proc. of the ICSE '05*, 2005, pp. 372–381.
13. Post A., Menzel I., Podelski A. Applying Restricted English Grammar on Automotive Requirements: Does it Work? A Case Study. *Proc. of the REFSQ'11*, 2011, pp. 166–180.
14. Ramezani E., Fahland D., van Dongen B. F., van der Aalst W. M. P. Diagnostic Information for Compliance Checking of Temporal Compliance Requirements. *Proc. of the CAiSE*, 2013, pp. 304–320.
15. *Property Pattern Mappings for LTL*. Available at: <http://patterns.projects.cis.ksu.edu/documentation/patterns/ltl.shtml> (accessed 21 December 2013).

УВАЖАЕМЫЕ АВТОРЫ!

Национальная электронная библиотека (НЭБ) продолжает работу по реализации проекта SCIENCE INDEX. После того как Вы регистрируетесь на сайте НЭБ (<http://elibrary.ru/defaultx.asp>), будет создана Ваша личная страничка, содержание которой составят не только Ваши персональные данные, но и перечень всех Ваших печатных трудов, имеющих в базе данных НЭБ, включая диссертации, патенты и тезисы к конференциям, а также сравнительные индексы цитирования: РИНЦ (Российский индекс научного цитирования), h (индекс Хирша) от Web of Science и h от Scopus. После создания базового варианта Вашей персональной страницы Вы получите код доступа, который позволит Вам редактировать информацию, помогая создавать максимально объективную картину Вашей научной активности и цитирования Ваших трудов.