

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Том 27

2021

№ 12

САПР

КОМПЬЮТЕРНАЯ ГРАФИКА

МЕТОДЫ ПРОГРАММИРОВАНИЯ

ОПЕРАЦИОННЫЕ СИСТЕМЫ И СРЕДЫ

ТЕЛЕКОММУНИКАЦИИ
И ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

НЕЙРОСЕТИ И
НЕЙРОКОМПЬЮТЕРЫ

СТРУКТУРНЫЙ СИНТЕЗ

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ
СИСТЕМЫ

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

ОПТИМИЗАЦИЯ И МОДЕЛИРОВАНИЕ

ИТ В ОБРАЗОВАНИИ

ГИС

Рисунок к статье И. В. Бычкова, С. А. Горского, А. Г. Феоктистова, Р. О. Костромина

«ПОДДЕРЖКА ВЫЧИСЛЕНИЙ В РАСПРЕДЕЛЕННЫХ СРЕДАХ НА ОСНОВЕ НЕПРЕРЫВНОЙ ИНТЕГРАЦИИ»

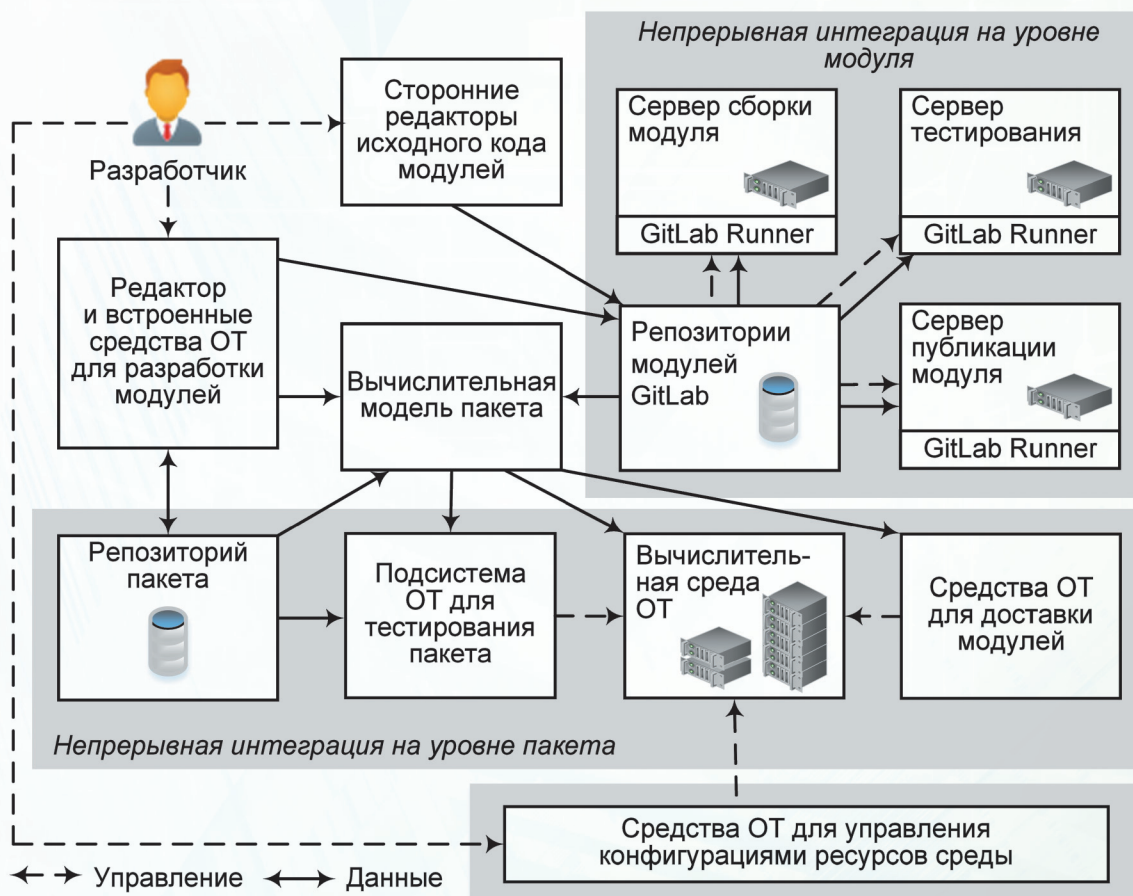


Рис. 1. Схема непрерывной интеграции в ОТ

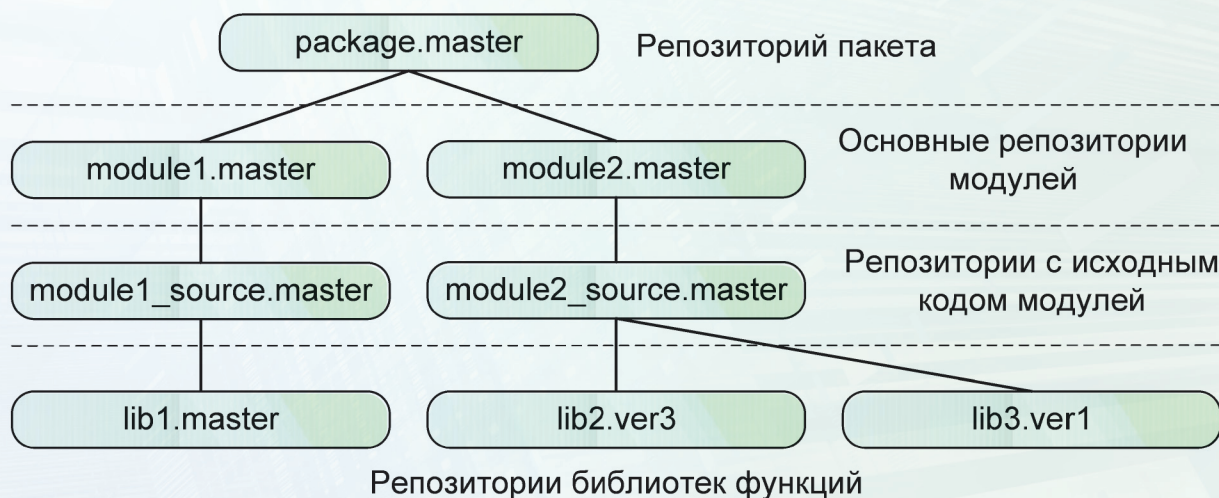


Рис. 2. Пример дерева зависимостей репозитория

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 27
2021
№ 12

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

DOI 10.17587/issn.1684-6400

УЧРЕДИТЕЛЬ
Издательство "Новые технологии"

СОДЕРЖАНИЕ

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

Бычков И. В., Горский С. А., Феоктистов А. Г., Костромин Р. О. Поддержка вычислений в распределенных средах на основе непрерывной интеграции 619

Левченко Н. Н., Змеев Д. Н. Динамический контроль целостности вычислительного процесса в параллельной потоковой вычислительной системе . 625

Тарасов В. Н., Бахарева Н. Ф. Математическая модель задержки в сетях связи на основе СМО с запаздыванием во времени 634

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

Мезенцев Ю. А., Короткова Ю. Л., Эстрайх И. В. Эффективный алгоритм решения прикладной задачи оптимизации расписаний параллельно-последовательной системы 642

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ

Мартынов В. В., Ширяев О. В. Обеспечение комплексной безопасности в рамках построения цифровой архитектуры предприятий 651

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ЭКОНОМИКЕ, УПРАВЛЕНИИ И ПРОИЗВОДСТВЕ

Медведев А. В., Ярещенко Д. И. Идентификация и управление многомерными процессами с запаздыванием в условиях неполной информации 658

Указатель статей, опубликованных в журнале "Информационные технологии" в 2021 г. 668

Главный редактор:

СТЕМПОВСКИЙ А. Л.,
акад. РАН, д. т. н., проф.

Зам. главного редактора:

ИВАННИКОВ А. Д., д. т. н., проф.
ФИЛИМОНОВ Н. Б., д. т. н., с.н.с.

Редакционный совет:

БЫЧКОВ И. В., акад. РАН, д. т. н.
ЖУРАВЛЕВ Ю. И.,
акад. РАН, д. ф.-м. н., проф.
КУЛЕШОВ А. П.,
акад. РАН, д. т. н., проф.
ПОПКОВ Ю. С.,
акад. РАН, д. т. н., проф.
РУСАКОВ С. Г.,
чл.-корр. РАН, д. т. н., проф.
РЯБОВ Г. Г.,
чл.-корр. РАН, д. т. н., проф.
СОЙФЕР В. А.,
акад. РАН, д. т. н., проф.
СОКОЛОВ И. А.,
акад. РАН, д. т. н., проф.
СУЕТИН Н. В., д. ф.-м. н., проф.
ЧАПЛЫГИН Ю. А.,
акад. РАН, д. т. н., проф.
ШАХНОВ В. А.,
чл.-корр. РАН, д. т. н., проф.
ШОКИН Ю. И.,
акад. РАН, д. т. н., проф.
ЮСУПОВ Р. М.,
чл.-корр. РАН, д. т. н., проф.

Редакционная коллегия:

АВДОШИН С. М., к. т. н., доц.
АНТОНОВ Б. И.
БАРСКИЙ А. Б., д. т. н., проф.
ВАСЕНИН В. А., д. ф.-м. н., проф.
ВАСИЛЬЕВ В. И., д. т. н., проф.
ВИШНЕКОВ А. В., д. т. н., проф.
ДИМИТРИЕНКО Ю. И., д. ф.-м. н., проф.
ДОМРАЧЕВ В. Г., д. т. н., проф.
ЗАБОРОВСКИЙ В. С., д. т. н., проф.
ЗАРУБИН В. С., д. т. н., проф.
КАРПЕНКО А. П., д. ф.-м. н., проф.
КОЛИН К. К., д. т. н., проф.
КУЛАГИН В. П., д. т. н., проф.
КУРЕЙЧИК В. В., д. т. н., проф.
ЛВОВИЧ Я. Е., д. т. н., проф.
МАРТЫНОВ В. В., д. т. н., проф.
МИХАЙЛОВ Б. М., д. т. н., проф.
НЕЧАЕВ В. В., к. т. н., проф.
ПОЛЕШУК О. М., д. т. н., проф.
ПРОХОРОВ С. А., д. т. н., проф.
САКСОНОВ Е. А., д. т. н., проф.
СОКОЛОВ Б. В., д. т. н., проф.
СОЛОВЬЕВ Р. А., д. т. н., в. н. с.
ТИМОНИНА Е. Е., д. т. н., проф.
УСКОВ В. Л., к. т. н. (США)
ФИНЬКО О. А., д. т. н., проф.
ФОМИЧЕВ В. А., д. т. н., проф.
ШИЛОВ В. В., к. т. н., доц.

Редакция:

БЕЗМЕНОВА М. Ю.

Информация о журнале доступна по сети Internet по адресу <http://novtex.ru/IT>.
Журнал включен в систему Российского индекса научного цитирования и базу данных RSCI на платформе Web of Science.
Журнал входит в Перечень научных журналов, в которых по рекомендации ВАК РФ должны быть опубликованы научные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

INFORMATION TECHNOLOGIES INFORMACIONNYYE TEHNOLOGII

Vol. 27
2021
No. 12

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

DOI 10.17587/issn.1684-6400

ISSN 1684-6400

CONTENTS

COMPUTING SYSTEMS AND NETWORKS

Bychkov I. V., Gorsky S. A., Feoktistov A. G., Kostromin R. O. Support for Computing in Distributed Environments Based on Continuous Integration 619

Levchenko N. N., Zmejev D. N. Dynamic Control of Computation Consistency in the Parallel Dataflow Computing System 625

Tarasov V. N., Bakhareva N. F. Mathematical Model of the Delay in Communication Networks Based on QS with a Time Lag 634

MODELING AND OPTIMIZATION

Mezentsev Yu. A., Korotkova Yu. L., Estrach I. V. An Efficient Algorithm for Solving the Applied Problem Scheduling Optimization of a Parallel-Sequential System 642

INFORMATION SECURITY

Martynov V. V., Shiryayev O. V. Ensuring Integrated Security as Part of Building Digital Architecture for Energy Companies 651

INFORMATION TECHNOLOGIES IN ECONOMY, MANAGEMENT AND PRODUCTION

Medvedev A. V., Yareshchenko D. I. Identification and Control of Multidimensional Delayed Processes in Conditions of Incomplete Information 658

Editor-in-Chief:

Stempkovsky A. L., Member of RAS,
Dr. Sci. (Tech.), Prof.

Deputy Editor-in-Chief:

Ivannikov A. D., Dr. Sci. (Tech.), Prof.
Filimonov N. B., Dr. Sci. (Tech.), Prof.

Chairman:

Bychkov I. V., Member of RAS,
Dr. Sci. (Tech.), Prof.
Zhuravljov Yu. I., Member of RAS,
Dr. Sci. (Phys.-Math.), Prof.
Kuleshov A. P., Member of RAS,
Dr. Sci. (Tech.), Prof.
Popkov Yu. S., Member of RAS,
Dr. Sci. (Tech.), Prof.
Rusakov S. G., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Ryabov G. G., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Soifer V. A., Member of RAS,
Dr. Sci. (Tech.), Prof.
Sokolov I. A., Member of RAS,
Dr. Sci. (Phys.-Math.), Prof.
Suetin N. V.,
Dr. Sci. (Phys.-Math.), Prof.
Chaplygin Yu. A., Member of RAS,
Dr. Sci. (Tech.), Prof.
Shakhnov V. A., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Shokin Yu. I., Member of RAS,
Dr. Sci. (Tech.), Prof.
Yusupov R. M., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.

Editorial Board Members:

Avdoshin S. M., Cand. Sci. (Tech.), Ass. Prof.
Antonov B. I.
Barsky A. B., Dr. Sci. (Tech.), Prof.
Vasenin V. A., Dr. Sci. (Phys.-Math.), Prof.
Vasiliev V. I., Dr. Sci. (Tech.), Prof.
Vishnekov A. V., Dr. Sci. (Tech.), Prof.
Dimitrienko Yu. I., Dr. Sci. (Phys.-Math.), Prof.
Domrachev V. G., Dr. Sci. (Tech.), Prof.
Zaborovsky V. S., Dr. Sci. (Tech.), Prof.
Zarubin V. S., Dr. Sci. (Tech.), Prof.
Karpenko A. P., Dr. Sci. (Phys.-Math.), Prof.
Kolin K. K., Dr. Sci. (Tech.)
Kulagin V. P., Dr. Sci. (Tech.), Prof.
Kureichik V. V., Dr. Sci. (Tech.), Prof.
Ljvovich Ya. E., Dr. Sci. (Tech.), Prof.
Martynov V. V., Dr. Sci. (Tech.), Prof.
Mikhailov B. M., Dr. Sci. (Tech.), Prof.
Nechaev V. V., Cand. Sci. (Tech.), Ass. Prof.
Poleschuk O. M., Dr. Sci. (Tech.), Prof.
Prokhorov S. A., Dr. Sci. (Tech.), Prof.
Saksonov E. A., Dr. Sci. (Tech.), Prof.
Sokolov B. V., Dr. Sci. (Tech.)
Solovyev R. A., Dr. Sci. (Tech.)
Timonina E. E., Dr. Sci. (Tech.), Prof.
Uskov V. L. (USA), Dr. Sci. (Tech.)
Finko O. A., Dr. Sci. (Tech.), Prof.
Fomichev V. A., Dr. Sci. (Tech.), Prof.
Shilov V. V., Cand. Sci. (Tech.), Ass. Prof.

Editors:

Bezmenova M. Yu.

Complete Internet version of the journal at site: <http://novtex.ru/IT>.

According to the decision of the Higher Certifying Commission of the Ministry of Education of Russian Federation, the journal is inscribed in "The List of the Leading Scientific Journals and Editions wherein Main Scientific Results of Theses for Doctor's or Candidate's Degrees Should Be Published"

И. В. Бычков, академик РАН, д-р техн. наук, проф., e-mail: idstu@icc.ru,

С. А. Горский, канд. техн. наук, ст. науч. сотр., e-mail: gorsky@icc.ru,

А. Г. Феоктистов, канд. техн. наук, доц., e-mail: agf@icc.ru,

Р. О. Костромин, мл. науч. сотр., e-mail: kostromin@icc.ru,

Институт динамики систем и теории управления им. В. М. Матросова СО РАН, г. Иркутск

Поддержка вычислений в распределенных средах на основе непрерывной интеграции*

Предложен новый подход к разработке программного обеспечения для распределенных вычислительных сред. Его целью является применение непрерывной интеграции как прикладного, так и системного программного обеспечения в процессе разработки распределенных пакетов прикладных программ на основе модульного подхода. Практическая значимость результатов исследования заключается в уменьшении числа ошибок и сбоев при разработке и применении пакетов.

Ключевые слова: распределенные вычисления, программное обеспечение, пакеты, непрерывная интеграция, GitLab

Введение

В процессе проведения крупномасштабных экспериментов при решении сложных задач математического моделирования нередко используются приложения, базирующиеся на выполнении научных рабочих процессов (workflows) в различных распределенных средах [1]. При этом среды могут включать разнородные вычислительные ресурсы.

Приложения такого вида коррелируют с понятием пакета прикладных программ, архитектура которого становится распределенной из-за использования распределенной среды [2]. Возникают проблемы обеспечения отказоустойчивости в процессе выполнения научных рабочих процессов с учетом изменения как прикладных программ, так и ресурсов среды [3, 4].

Однако поддержка непрерывной интеграции программного обеспечения (ПО) в подобных

приложениях до сих пор является нетривиальной задачей [5]. Проблемы доставки и тестирования ПО для интегрированных научных рабочих процессов обсуждаются в работе [6].

Результаты анализа инструментальных средств [7–9], применение которых в той или иной степени возможно при разработке приложений, выявляют проблемы при реализации непрерывной интеграции прикладного и системного ПО для разнородных сред [10, 11]. Кроме того, отдельно взятый инструмент часто не решает всех проблем, а совместное применение нескольких инструментов ограничено использованием различных форматов данных, программных протоколов и интерфейсов, что может привести к нарушению процесса непрерывной интеграции в будущем [12].

Таким образом, для улучшения качества ПО и сокращения времени его разработки, подготовки и проведения экспериментов должны быть существенно расширены функциональные возможности непрерывной интеграции ПО в инструментальных средствах для их разработки и применения.

В статье рассмотрен новый подход к поддержке непрерывной интеграции прикладного и системного ПО в инструментальном ком-

*Исследование выполнено при поддержке Министерства науки и высшего образования Российской Федерации, проект № 121032400051-9 "Технологии разработки и анализа предметно-ориентированных интеллектуальных систем группового управления в недетерминированных распределенных средах", а также Российского фонда фундаментальных исследований, проект № 19-07-00097.

плексе Orlando Tools (OT) [13]. Представлены аспекты взаимодействия с внешними системами непрерывной интеграции. Приведен базовый динамический алгоритм непрерывной интеграции, реализованный в ОТ. Рассмотрены вопросы конфигурирования ресурсов среды.

1. Основные понятия

Рассмотрим используемые в статье неформальные понятия объектов и процессов непрерывной интеграции.

Распределенная вычислительная среда (РВС) объединяет множество разнородных ресурсов (персональных компьютеров, высокопроизводительных серверов и кластеров, компонентов облачных платформ).

Распределенный пакет прикладных программ (РППП) — это комплекс взаимосвязанных прикладных *модулей*, а также программных и языковых *средств системного обеспечения* их совместного использования в процессе автоматизации решения определенного класса задач в РВС.

Модуль представляет собой прикладную программу или сервис, снабженные спецификациями, которые отражают информацию о назначении, способах применения, форматах входных и выходных параметров, а также другие необходимые для вычислительного процесса сведения.

В статье *средства системного обеспечения* РППП представлены компонентами ОТ: веб-интерфейсом пользователя, конструктором описания пакета, базой знаний, исполнительной подсистемой, средствами непрерывной интеграции и базой расчетных данных. ОТ устанавливается на выделенный сервер. Языковые средства ОТ поддерживают описание пакета как в текстовом, так и в графическом режимах. Они обеспечивают формирование спецификаций модулей, описание их взаимосвязей в терминах абстрактных параметров и операций, построение схем решения задач (workflows) и задание условий совместного выполнения модулей в распределенной среде в виде набора продукций. Сервер ОТ поддерживает подключение ресурсов РВС с помощью специальных API. Описание пакета представляет собой его *вычислительную модель*.

Под *непрерывной интеграцией*, *доставкой и развертыванием* ПО понимается автоматизация процессов, связанных с разработкой,

сборкой, тестированием, доставкой и установкой модулей пакета и его самого в целом. Для краткости в статье используется обобщающий термин *непрерывная интеграция*.

Основными объектами процесса непрерывной интеграции являются модули, схемы решения задач и пакеты, а также репозитории для их хранения и ресурсы, на которых они размещаются.

Сборка и тестирование ПО осуществляется с помощью компонента GitLab Runner системы GitLab [9] на специальных серверах. *Сервер сборки или тестирования* представляет собой выделенный или невыделенный вычислительный ресурс РВС с установленным на нем компонентом GitLab Runner, а также программным обеспечением, необходимым для выполнения этих процессов применительно к модулям РППП или схемам решения задач, которые могут определять различные комбинации модулей.

2. Реализация непрерывной интеграции в ОТ

Создание и применение РППП представляют собой сложные итерационные процессы, которые в общем случае включают следующие основные этапы [2]:

- разработку и/или модификацию модулей пакета;
- разработку и/или модификацию вычислительной модели пакета;
- компиляцию, отладку и тестирование модулей в узлах среды с использованием GitLab Runner;
- размещение скомпилированных версий модулей в репозиториях и/или установку модулей на узлы;
- отладку и тестирование схем решения задач в ОТ с использованием новых или модифицированных версий модулей в комбинации с существующими модулями.

Так как перечисленные выше этапы многократно выполняются силами разработчиков пакетов, на которых возлагаются все накладные расходы по созданию и сопровождению ПО, актуальность автоматизации выполнения работ в рамках этих этапов не вызывает сомнений.

Общая схема выполнения непрерывной интеграции в ОТ приведена на рис. 1 (см. вторую сторону обложки). Передача данных и управление осуществляются с помощью специально разработанных прикладных программных протоколов, предоставляющих функции получения, обработки и хранения исходного и

бинарного кодов модулей, их конфигураций, тестовых исходных данных и результатов экспериментов.

Данные протоколы базируются на использовании языка JavaScript Object Notation (JSON). Разработаны специализированные структуры данных на этом языке для хранения, обработки и передачи необходимой системной информации. Использование JSON в качестве базового языка упрощает автоматизацию обработки данных, передаваемых между различными системами, применяемыми в процессе непрерывной интеграции.

Средства ОТ непрерывной интеграции ПО дополнены новыми функциями для реализации следующих системных операций:

- создания репозиториях всех типов;
- описания репозиториях в рамках вычислительной модели РППП;
- модификации вычислительной модели пакета на основе информации из репозиториях;
- установки модулей на вычислительные ресурсы среды;
- тестирования модулей для проверки их работоспособности и определения вычислительных возможностей узлов среды;
- тестирования схем решения задач РППП;
- хранения информации о результатах тестирования и расчетов.

Процессы разработки РППП и описание схемы непрерывной интеграции ПО пакетов в ОТ детально представлены соответственно в работах [2] и [10]. Ниже приведен базовый динамический алгоритм непрерывной интеграции с использованием разных типов репозиториях, а также рассмотрены средства ОТ, предназначенные для управления конфигурациями ресурсов среды.

3. Динамический алгоритм непрерывной интеграции

В ОТ используются репозитории двух типов: репозитории модулей и репозитории пакетов.

Каждый модуль должен быть включен хотя бы в один репозиторий системы GitLab. Основной репозиторий модуля содержит исходный код модуля, информацию, необходимую для его сборки и тестирования, сведения об интерфейсе модуля и инструкции по его установке на ресурсах, а также ссылки на дополнительные репозитории, в которых, как правило, хранятся исходный код для различных библиотек функций, используемых при ком-

пиляции модуля, а также правила их сборки, тестирования и размещения. Подключение дополнительных репозиториях необходимо в том случае, если разработчик модуля хочет включить создание вспомогательных библиотек в единую цепочку непрерывной интеграции.

Репозиторий пакета, разрабатываемый в ОТ, включает описание пакета по версиям со ссылками на репозитории модулей, а также тестовые данные для схем решения задач. Посредством таких ссылок строится дерево зависимостей репозиториях (рис. 2, см. вторую сторону обложки).

Для контроля и учета изменений ПО пакетов разработан динамический алгоритм непрерывной интеграции. Данный алгоритм периодически запускается с дискретным шагом времени τ для конкретного пакета. Шаг τ может быть индивидуально настроен для каждого пакета.

Пусть A — это множество системных атрибутов, представляющих собой характеристики объектов и процесса непрерывной интеграции, а $B \subset A$ — это множество булевых переменных, принимающих значения из множества $\{0, 1, \theta\}$. Символ θ означает, что значение переменной не определено. Обозначим Q — множество логических выражений, составленных из переменных множества B и логических операторов. Будем считать, что логическое выражение $q \in Q$ не определено, если не определена хотя бы одна переменная, входящая в него. Введем множество O операций непрерывной интеграции (действий по обработке изменений в пакете). Каждой i -й операции соответствуют два множества системных параметров $O_i^{inputs} \subset A$ и $O_i^{outputs} \subset A$, представляющих соответственно исходные данные и результаты ее выполнения, $O_i^{outputs} \cap B \neq \emptyset$. Определим условия применения каждой i -й операции с помощью продукции вида $p_i : q_i \Rightarrow o_i$. Таким образом, все множество P продукций определяет условия применения операций из O . Каждой продукции назначен приоритет.

В качестве системных параметров могут выступать атрибуты репозиториях и их коммитов, списки репозиториях, статус процесса непрерывной интеграции пакета, характеристики конвейера непрерывной интеграции пакета, описание пакета из репозитория, тесты для пакета, тестовые ресурсы, сообщения о сбоях и ошибках, результаты тестов и др. Множество булевых переменных включает признаки коммита, нового или активного конвейера Gitlab, завершения конвейера, наличия сбоев и ошибок в конвейере.

ре, активности в репозиториях пакета, завершения разбора описания пакета, наличия тестовых ресурсов с установленными модулями, корректного завершения всех тестов и других условий.

Типовыми операциями являются получение информации о репозиториях, запуск и завершение обработки репозитория, запуск и завершение конвейеров непрерывной интеграции пакетов в основных репозиториях, запуск и завершение конвейеров в дополнительных репозиториях, получение и разбор описаний пакетов, установка модулей и пакетов на тестовые и целевые ресурсы, запуск тестов модулей и пакетов, сохранение изменений в базе данных, вывод сообщений в лог, отправка сообщений разработчикам и другие действия.

Средства ОТ, реализующие такие операции, существенно расширяют функциональные возможности используемых сторонних систем непрерывной интеграции. Ниже приведены основные шаги динамического алгоритма непрерывной интеграции.

Алгоритм.

```

1  for  $i = \overline{1, k}$  do
2    Определение значений логических параметров, определяющих состояние  $i$ -го репозитория, с помощью системы мониторинга.
3    Формирование множества  $P^{cur} = \{p_j^{cur} : q_j \in \overline{0, 1}\}$  продуктов,  $P^{cur} \subset P$ .
4    if  $P^{cur} = \emptyset$  then
5      Завершение работы с  $i$ -м репозиторием. Переход на шаг 14.
6    else
7      Формирование множества  $O^{cur} = \{o_l^{cur} : p_l^{cur} \in P^{cur} \wedge q_l = 1\}$  операций непрерывной интеграции,  $O^{cur} \subset O$ .
8      if  $O^{cur} = \emptyset$  then
9        Завершение работы с  $i$ -м репозиторием. Переход на шаг 14.
10     else
11       Выполнение операций непрерывной интеграции из множества  $O^{cur}$ , включая операции завершения работы алгоритма в случае некорректного завершения конвейера непрерывной интеграции для  $i$ -го репозитория или взаимосвязанного с ним репозитория и отправки сообщений о сбоях и ошибках процесса непрерывной интеграции. Переход на шаг 2.
12     end if
13   end if
14    $k = k + 1$ 
15 end do
16 Завершение работы алгоритма.
```

В алгоритме осуществляется итерационная смена двух основных фаз работы с репозиторием. В рамках первой фазы формируется множество продуктов, готовых к выполнению. Затем определяется множество операций, которые можно применить в процессе непрерывной интеграции.

Во время второй фазы эти операции выполняются, что приводит к изменению значений булевых переменных и, соответственно, логических выражений, определяющих состояние объектов и процессов непрерывной интеграции. Далее происходит возврат к первой фазе.

Работа с репозиторием завершается, когда нет ни одной готовой к выполнению продукции или после очередной итерации выполнения продукции невозможно применить ни одну операцию.

В результате работы рассмотренного выше алгоритма непрерывной интеграции разработчик получает готовый к использованию РППП или детальную информацию об отказах и ошибках, выявленных в процессе непрерывной интеграции ПО пакета.

4. Средства управления конфигурациями ресурсов

Автоматизация процессов формирования среды с последующей доставкой модулей и данных на ее ресурсы, а также их конфигурирование является актуальной задачей для систем управления научными рабочими процессами [14]. В рамках данного исследования такая автоматизация осуществляется средствами ОТ. Они направлены на повышение степени управляемости процессов настройки и сопровождения ресурсов, а также на обеспечение надежности их работы. Их реализация базируется на использовании мультиагентных технологий. Кроме того, они предоставляют возможность подготовки и запуска виртуальных машин для выполнения модулей пакетов с помощью оригинальной гипервизорной оболочки [15]. Задания с такими виртуальными машинами могут запускаться из общих очередей ресурсов центров коллективного пользования. Известные средства аналогичного назначения [16–18] не обладают такими способностями.

Схема управления конфигурациями ресурсов показана на рис. 3 (см. третью сторону обложки).

Разработчик специфицирует требования к конфигурации ресурсов и сохраняет их в ре-

позитории пакета. Агент трансляции передает сформированные спецификации на управляемые ресурсы. Спецификация представляет собой набор команд, исполняемых ресурсными агентами. В процессе функционирования ресурсов мониторинг и модификация их конфигураций осуществляется вышеупомянутыми ресурсными агентами в соответствии с заданными требованиями.

В качестве базовой платформы для управления конфигурациями выбрана система Ansible [16]. Ее функционал расширен дополнительными функциями, реализуемыми ОТ. Такой выбор обусловлен следующими факторами: поддержкой широкого спектра популярных операционных систем; использованием императивной конфигурации ресурсов, которая поддерживает достаточно полный контроль процесса установки ПО; использованием лицензии GPLv3+, предполагающей бесплатное распространение программы; поддержкой автоматического обновления конфигураций ресурсов сервером при получении им новых версий спецификаций этих ресурсов (доставка конфигураций осуществляется путем применения метода Push); управление ресурсами с помощью протокола SSH, обеспечивающего управление устройствами, поддерживающими данный протокол. Результаты сравнительного анализа процесса подготовки экспериментальной РВС показывают существенные преимущества управления конфигурациями ресурсов в ОТ с использованием Ansible по сравнению с другими системами аналогичного назначения [19].

На рис. 4 (см. третью сторону обложки) приведено время конфигурации 10 узлов предметно-ориентированной РВС, организованной для поддержки принятия решений по обеспечению живучести систем энергетики на основе комбинаторного моделирования [20].

Было оценено время выполнения восьми этапов конфигурирования узлов в ручном режиме и с использованием средств ОТ. В рамках вышеупомянутых этапов выполняются следующие операции: установка необходимого ПО на ресурсы и конфигурация среды, запуск мультиагентной платформы, запуск агентов на ресурсах и их регистрация, распределение вычислительных работ, развертывание исполняемых файлов на ресурсах агентов, запуск приложения, запуск модулей, сбор и анализ результатов выполнения ПО. Использование средств ОТ позволило сократить время конфигурирования узлов почти в 4 раза по сравнению с ручным режимом их настройки.

Заключение

Предложен новый подход к непрерывной интеграции программного обеспечения в процессе разработки РППП на основе модульного подхода. Средства ОТ, базирующиеся на применении представленного в статье динамического алгоритма непрерывной интеграции обеспечивают разработчикам возможность существенного сокращения трудозатрат на создание пакетов, отладку и тестирование прикладного ПО, а также конфигурирование разнородных ресурсов вычислительной среды.

Список литературы

1. Hu H., Li Z., Hu H., Chen J., Ge J., Li C., Chang V. Multi-objective scheduling for scientific workflow in multicloud environment // *Journal of Network and Computer Applications*. 2018. Vol. 114. P. 108–122.
2. Феоктистов А. Г., Сидоров И. А., Горский С. А. Автоматизация разработки и применения распределенных пакетов прикладных программ // *Проблемы информатики*. 2017. № 4. С. 61–78.
3. Sidorov I. A. Methods and Tools to Increase Fault Tolerance of High-Performance Computing Systems // *Proceedings of the 39th International Convention on information and communication technology, electronics and microelectronics (MIPRO-2016)*. Riejska: IEEE, 2016. P. 242–246.
4. Poola D., Salehi M. A., Ramamohanarao K., Buyya R. A taxonomy and survey of fault-tolerant workflow management systems in cloud and distributed computing environments // *Software architecture for big data and the cloud*. Morgan Kaufmann, 2017. P. 285–320.
5. Bungartz H. J., Reiz S., Uekermann B., Neumann P., Nagel W. E. Software for Exascale Computing-SPPEXA 2016-2019 // *Lecture Notes in Computational Science and Engineering*. 2020. Vol. 136. 624 p.
6. Möller S. et al. Robust cross-platform workflows: how technical and scientific communities collaborate to develop, test and share best practices for data analysis // *Data Science and Engineering*. 2017. Vol. 2, N. 3. C. 232–244.
7. Sochat V. Containershare: Open Source Registry to build, test, deploy with CircleCI // *The Journal of Open Source Software*. 2018. Vol. 3, N. 28. P. 1–3.
8. Soni M. Jenkins Essentials. Packt Publishing Ltd, 2015. 157 p.
9. Gruver G. Start and Scaling Devops in the Enterprise. BookBaby, 2016. 100 p.
10. Горский С. А., Костромин Р. О., Феоктистов А. Г., Бычков И. В. Инструментальный комплекс поддержки высокопроизводительных вычислений в предметно-ориентированных гетерогенных средах // *Информационные технологии и нанотехнологии: Сб. тр. VI Междунар. конф. и молодежной школы (ИТНТ-2020)*. Самара: Новая техника, 2020. Т. 4. Науки о данных. С. 582–589.
11. Shahin M., Babar M. A., Zhu L. Continuous integration, delivery and deployment: a systematic review on approaches, tools, challenges and practices // *IEEE Access*. 2017. Vol. 5. P. 3909–3943.
12. Bobrovskis S., Jurenoks A. A Survey of Continuous Integration, Continuous Delivery and Continuous Deployment // *BIR Workshops*. 2018. P. 314–322.
13. Tchernykh A., Feoktistov A., Gorsky S., Sidorov I., Kostromin R., Bychkov I., Basharina O., Alexandrov A., Rivera-

Rodriguez R. Orlando Tools: Development, Training, and Use of Scalable Applications in Heterogeneous Distributed Computing Environments // Communications in Computer and Information Science. 2019. Vol. 979. P. 265–279.

14. **Chakroborti D., Roy B., Nath S. S.** Designing for Recommending Intermediate States in A Scientific Workflow Management System // Joint Proceedings of the BIR 2018 Short Papers, Workshops and Doctoral Consortium co-located with 17th International Conference Perspectives in Business Informatics Research (BIR 2018). CEUR-WS Proceedings. 2018. Vol. 2218. P. 314–322.

15. **Feoktistov A., Sidorov I., Sergeev V., Kostromin R., Bogdanova V.** Virtualization of Heterogeneous HPC-clusters Based on OpenStack Platform // Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика. 2017. Т. 6, № 2. С. 37–48.

16. **Powerful** infrastructure automation and delivery. URL: <https://puppet.com/> (дата обращения: 30.03.21).

17. **Ansible** is Simple IT Automation. URL: <https://www.ansible.com/> (дата обращения: 30.03.21).

18. **SaltStack**. URL: <https://www.saltstack.com/> (дата обращения: 30.03.21).

19. **Kostromin R. O.** Survey of software configuration management tools of nodes in heterogeneous distributed computing environment // Proceedings of the 2nd International Workshop on Information, Computation, and Control Systems for Distributed Environments. CEUR-WS Proceedings. 2020. Vol. 2638. P. 156–165.

20. **Bychkov I., Oparin G., Feoktistov A., Sidorov I., Gorsky S., Kostromin R., Edelev A.** Subject-oriented computing environment for solving large-scale problems of energy security research // Journal of Physics: Conference Series. 2019. Vol. 1368. P. 052030-1-052030-12.

I. V. Bychkov, Academician of RAS, Professor, e-mail: idstu@icc.ru,
S. A. Gorsky, Ph.D., Senior Researcher, e-mail: gorskysergey@mail.ru,
A. G. Feoktistov, Ph.D., Assistant Professor, e-mail: agf@icc.ru,
R. O. Kostromin, Junior Researcher, e-mail: kostromin@icc.ru,
Matrosov Institute for System Dynamics and Control Theory of SB RAS,
Irkutsk, 664033, Russian Federation

Support for Computing in Distributed Environments Based on Continuous Integration

Nowadays, tools for designing scientific applications often do not implement the required continuous integration capabilities of the applied software. Therefore, such overheads as the application development time and experiment execution makespan are substantially increased. In this regard, we propose a new approach to developing scientific applications and carrying out experiments with them. It is based on applying continuous integration to both the applied and system software in developing distributed applied software packages with a modular architecture using the Orlando Tools framework. Within the proposed approach, we provide integrating the Orlando Tools subsystems with the GitLab system and automating the development of package modules. At the same time, Orlando Tools fully support constructing and testing problem-solving schemes (workflows) that combine package modules located on environment resources with different computational characteristics. To this end, Orlando Tools provides the necessary configuring and setting up of computational resources. The practical significance of our study is substantial reduction overheads needed to experiment fulfillments and increase of the resource use efficiency.

Keywords: distributed computing, software, packages, continuous integration, GitLab

DOI: 10.17587/it.27.619-625

References

1. **Hu H., Li Z., Hu H., Chen J., Ge J., Li C., Chang V.** Multi-objective scheduling for scientific workflow in multicloud environment, *Journal of Network and Computer Applications*, 2018, vol. 114, pp. 108–122.

2. **Feoktistov A., Sidorov I., Gorsky S.** Avtomatizatsiya razrabotki i primeneniya raspredelennykh paketov prikladnykh program [Automation of development and application of distributed applied software packages], *Problemy informatiki* [Problems of Informatics], 2017, no. 4, pp. 61–78.

3. **Sidorov I. A.** Methods and Tools to Increase Fault Tolerance of High-Performance Computing Systems, *Proceedings of the 39th International Convention on information and communication technology, electronics and microelectronics (MIPRO-2016)*, Riejska, IEEE, 2016, pp. 242–246.

4. **Poola D., Salehi M. A., Ramamohanarao K., Buyya R.** A taxonomy and survey of fault-tolerant workflow management systems in cloud and distributed computing environments, *Software architecture for big data and the cloud*, Morgan Kaufmann, 2017, pp. 285–320.

5. **Bungartz H. J., Reiz S., Uekermann B., Neumann P., Nagel W. E.** Software for Exascale Computing-SPPEXA 2016-2019, *Lecture Notes in Computational Science and Engineering*, 2020, vol. 136, 624 p.

6. **M@ller S. et al.** Robust cross-platform workflows: how technical and scientific communities collaborate to develop, test and share best practices for data analysis, *Data Science and Engineering*, 2017, vol. 2, no. 3, pp. 232–244.

7. **Sochat V.** Containershare: Open Source Registry to build, test, deploy with CircleCI, *The Journal of Open Source Software*, 2018, vol. 3, no. 28, pp. 1–3.

8. **Soni M.** Jenkins Essentials, Packt Publishing Ltd, 2015, 157 p.
9. **Gruver G.** Start and Scaling Devops in the Enterprise. BookBaby, 2016, 100 p.
10. **Gorsky S. A., Kostromin R. O., Feoktistov A. G., Bychkov I. V.** Toolkit for supporting high-performance computing in subject-oriented heterogeneous environments, *Proceedings of the 6th International Conference on Information Technology and Nanotechnology (ITNT-2020)*, Samara, Novaya tekhnika, 2020, vol. 4, Data Science, pp. 582—589.
11. **Shahin M., Babar M. A., Zhu L.** Continuous integration, delivery and deployment: a systematic review on approaches, tools, challenges and practices, *IEEE Access*, 2017, vol. 5, pp. 3909—3943.
12. **Bobrovskis S., Jurenoks A.** A Survey of Continuous Integration, Continuous Delivery and Continuous Deployment, *BIR Workshops*, 2018, pp. 314—322.
13. **Tchernykh A., Feoktistov A., Gorsky S., Sidorov I., Kostromin R., Bychkov I., Basharina O., Alexandrov A., Rivera-Rodriguez R.** Orlando Tools: Development, Training, and Use of Scalable Applications in Heterogeneous Distributed Computing Environments, *Communications in Computer and Information Science*, 2019, vol. 979, pp. 265—279.
14. **Chakroborti D., Roy B., Nath S. S.** Designing for Recommending Intermediate States in A Scientific Workflow Management System, *Joint Proceedings of the BIR 2018 Short*

- Papers, Workshops and Doctoral Consortium co-located with 17th International Conference Perspectives in Business Informatics Research (BIR 2018)*, CEUR-WS Proceedings, 2018, vol. 2218, pp. 314—322.
15. **Feoktistov A., Sidorov I., Sergeev V., Kostromin R., Bogdanova V.** Virtualization of Heterogeneous HPC-clusters Based on OpenStack Platform, *Vestnik YUzhno-Ural'skogo gosudarstvennogo universiteta. Seriya: Vychislitel'naya matematika i informatika*, 2017, vol. 6, no. 2, pp. 37—48.
16. **Powerful** infrastructure automation and delivery, available at: <https://puppet.com/> (date of access: 30.03.21).
17. **Ansible** is Simple IT Automation, available at: <https://www.ansible.com/> (date of access: 30.03.21).
18. **SaltStack**, available at: <https://www.saltstack.com/> (date of access: 30.03.21).
19. **Kostromin R. O.** Survey of software configuration management tools of nodes in heterogeneous distributed computing environment, *Proceedings of the 2nd International Workshop on Information, Computation, and Control Systems for Distributed Environments*, CEUR-WS Proceedings, 2020, vol. 2638, pp. 156—165.
20. **Bychkov I., Oparin G., Feoktistov A., Sidorov I., Gorsky S., Kostromin R., Edelev A.** Subject-oriented computing environment for solving large-scale problems of energy security research, *Journal of Physics: Conference Series*, 2019, vol. 1368, pp. 052030-1-052030-12.

УДК 004.027

DOI: 10.17587/it.27.625-633

Н. Н. Левченко, канд. техн. наук, заведующий отделом, e-mail: nick@ippm.ru,
Д. Н. Змеев, науч. сотр., e-mail: zmejevdn@list.ru,
Институт проблем проектирования в микроэлектронике РАН, Москва

Динамический контроль целостности вычислительного процесса в параллельной потоковой вычислительной системе

При создании высокопроизводительных многопроцессорных вычислительных систем большое внимание уделяется вопросам обеспечения бесперебойного функционирования с точки зрения как программного, так и аппаратного обеспечения. В традиционных вычислительных системах решение данных вопросов возложено в основном на программное обеспечение. В статье рассматривается решение вопроса обеспечения динамического контроля целостности вычислительного процесса для параллельной потоковой вычислительной системы (ППВС), которая реализует потоковую модель вычислений с динамически формируемым контекстом. За счет особенностей ППВС предлагается реализовать данный вид контроля аппаратно, что позволит повысить его эффективность, поскольку вычислительный процесс будет контролироваться не только в статике, но и в динамике.

Ключевые слова: потоковая модель вычислений, процессор сопоставлений, аппаратный контроль целостности вычислений

Введение

Сегодня основные усилия создателей суперкомпьютеров (многопроцессорных высокопроизводительных вычислительных систем) направлены на преодоление барьера в квинтиллион (миллиард миллиардов) вещественных операций в секунду (10^{18} FLOPS). И этот момент по-настоящему близок, о чем сви-

детельствует динамика развития передовых вычислительных систем, входящих в список TOP500 [1]. Однако вычислительные системы, входящие в этот список, используют преимущественно процессоры фирмы Intel (94 %) [2], которые опираются на традиционную фон-неймановскую модель вычислений. Данная модель вычислений создавалась прежде всего для выполнения последовательных вычисле-

ний, что полностью устраивало и производителей, и программистов, пока развитие технологических норм производства позволяло существенно повышать производительность одного кристалла. С определенного же момента (с начала 2000-х годов) создатели микропроцессоров вынуждены были перейти к многоядерным кристаллам, поскольку усложнение одного ядра уже не обеспечивало желаемого повышения его производительности. В настоящее время для повышения производительности вычислительных систем используются различные методы и средства [3, 4].

С появлением многоядерных процессоров встал вопрос о том, как обеспечить загрузку "новых" аппаратных ресурсов. Существующее в то время программное обеспечение было в основном ориентировано на одноядерное (однопоточное) использование. Решением этой проблемы стало развитие параллельного программирования, при котором основная нагрузка возложена на разработчиков приложений, с небольшой поддержкой со стороны аппаратуры или вовсе без нее. Добавление так называемого человеческого фактора в распределение вычислений по доступным аппаратным ресурсам способствует появлению ошибок.

Одним из решений этой проблемы является переход к другим моделям вычислений, одной из которых является потоковая модель вычислений с динамически формируемым контекстом, которой имманентно присущ параллелизм вычислений. Поточковые модели вычислений были особенно популярны в 80-х годах прошлого века, однако из-за ряда причин не нашли своего применения [5–7]. Данная модель вычислений реализуется в архитектуре параллельной потоковой вычислительной системы (ППВС) "Буран", работу над которой ведут авторы [8]. При создании подобных систем высокой сложности вопросам обеспечения надежности [9–14] уделяется большое внимание, поскольку необходимо обеспечить нормальное функционирование вычислительной системы, несмотря на сбои, отказы и ошибки в программах.

Данная статья посвящена такому аспекту надежности, как динамический контроль целостности вычислительного процесса. Под *динамическим контролем целостности вычислительного процесса* понимается комплекс мер по обнаружению, корректировке и информированию пользователя о ситуациях, препятствующих нормальному прохождению вычислительного процесса, связанных с ошибками,

возникающими по вине программиста или аппаратуры, которые нельзя обнаружить на этапе компиляции, до запуска программы (на этапе статического контроля).

В традиционных вычислительных системах основной контроль за состоянием целостности вычислительного процесса возложен на программные средства, такие как операционная система и компилятор. Попытки переложить часть функций операционной системы на аппаратуру осуществлялись и ранее [15, 16], однако не имели успеха. Компилятор же способен контролировать возможные ошибки в программе исключительно в статике — без анализа изменяемых данных, к которым относятся входные данные, переменные и т. п. В динамике контроль за вычислительным процессом осуществляется на уровне операционной системы.

Например, в традиционных вычислительных системах при операциях обращения к ячейкам памяти у аппаратуры в большинстве своем отсутствует информация о состоянии этих ячеек, т. е. о том, содержится ли в них какое-либо данное или просто некий "мусор". В результате основной контроль за обращением к памяти возложен на операционные системы. Это, фактически, является накладными расходами на обеспечение вычислений.

Для преодоления этого недостатка в традиционных вычислительных системах некоторые исследователи предлагают использовать механизм full/empty bits [17], который можно назвать своеобразным аналогом ассоциативной памяти.

Целью данной работы является изучение возможности реализации динамического контроля целостности вычислительного процесса на аппаратном уровне в параллельной потоковой вычислительной системе.

1. Потоковая модель вычислений и архитектура, ее реализующая

Потоковая модель вычислений с динамически формируемым контекстом характеризуется активацией вычислительных квантов по готовности данных. Вычислительным квантом называется программа относительно небольшого размера (программный узел), которая активируется, как только на ее вход поступят все данные. Между собой эти кванты взаимодействуют с помощью сообщений — структуры данных, содержащих операнд, служебные поля

и контекст. Контекст однозначно определяет положение операнда в виртуальном адресном пространстве задачи.

Именно сопоставление сообщений и является основой потоковой модели вычислений с динамически формируемым контекстом. Сопоставлением называется процедура, в результате которой по заранее определенным правилам осуществляется поиск сообщений с "совпадающими" признаками. В результате положительного поиска (при совпадении) образуется пакет — структура данных, готовая к обработке. Пакет является полностью автономной структурой данных, для обработки которой не требуется запрос каких-либо дополнительных данных, а следовательно, отсутствует необходимость приостановки вычислительного процесса. Обработка пакета происходит в соответствии с последовательностью команд программного узла, адрес которого присутствует в самом пакете. В результате обработки пакета образуются новые сообщения, которые либо посылаются на другие узлы, либо в качестве результата вычислений выдаются на ХОСТ-машину. Адреса узлов и контекст новых сообщений вычисляются непосредственно в программе узла, т. е. программа выполняется в парадигме "раздачи". В парадигме "раздачи" производитель каждого нового операнда знает, кому это сообщение предназначается, и самостоятельно обеспечивает рассылку по нужным адресатам. Получатель же операнда просто ожидает его прихода и ничего не знает о его источнике. Данная парадигма программирования является более экономной в плане числа обменов сообщениями — одно сообщение на каждое использование операнда. В парадигме "сбора", которая в основном используется в традиционной модели вычислений, напротив, только потребитель данных знает, какие данные ему нужны и где их взять, и сам их запрашивает, указывая имя переменной (адрес в памяти). Кроме того, работа в данной парадигме значительно усложняет аппаратуре предсказание того, где и когда потребуются те или иные данные.

Потоковая модель вычислений с динамически формируемым контекстом воплощена в параллельном языке программирования DFL. Программа на этом языке представляет собой набор программных узлов, состоящих из заголовка узла и программного кода узла. Активация узла происходит тогда и только тогда, когда на все входы одного узла с определенным именем и контекстом придут все требуемые данные. Важной особенностью парал-

лельного языка является управление уровнем параллелизма программы и требуемыми для выполнения узла ресурсами (объемом ассоциативной памяти ключей, размером буферов и т. п.) посредством изменения размера программного узла (чем меньше размер программного узла, тем больше уровень параллелизма и тем меньше требуется аппаратных ресурсов для выполнения этого узла).

Потоковая модель вычислений с динамически формируемым контекстом реализуется в архитектуре ППВС "Буран".

ППВС является многоядерной масштабируемой вычислительной системой. Между вычислительными ядрами в системе передаются единицы информации в виде токенов. Токеном называется структура данных, в состав которой входит передаваемый операнд, ключ с контекстом и адресом программного узла, маска и ряд других специальных признаков. Коммутация между ядрами осуществляется на основе значения номера ядра, вырабатываемого блоком хэширования на основе настраиваемой функции распределения вычислений, входным параметром которой является контекст токена. Группа вычислительных ядер в рамках одного кристалла называется вычислительным модулем.

В состав вычислительного модуля входят следующие узлы и блоки:

- процессор сопоставлений, выполняющий сопоставление поступающих на вход токенов по определенным правилам и формирующий пакеты, готовые к активации на любом свободном исполнительном устройстве;
- исполнительное устройство (ИУ), выполняющее обработку пакетов и формирующее новые токены в соответствии с программой узла;
- блок хэш-функций, выполняющий на основе выбранной пользователем хэш-функции преобразование контекста токена в номер вычислительного ядра, в которое должен поступить сформированный токен;
- внутренний коммутатор токенов, выполняющий передачу токенов либо во внешнюю коммуникационную сеть, либо в процессор сопоставлений своего вычислительного модуля;
- внутренний коммутатор пакетов, выполняющий функцию аппаратного балансера, передавая пакеты на любое свободное в данный момент исполнительное устройство вычислительного модуля.

Более подробно модель вычислений и архитектура ППВС описаны в работах [18—21].

2. Архитектура процессора сопоставлений

Наиболее эффективным образом механизм аппаратного сопоставления токенов осуществляется с помощью аппаратной ассоциативной памяти, которая является одним из основных узлов процессора сопоставлений (ПС). Проведенные эксперименты подтверждают, что использование аппаратной ассоциативной памяти (а не ее микропрограммная эмуляция) позволяет существенно повысить реальную производительность системы на широком круге задач.

В настоящее время в мире возобновился интерес исследователей к ассоциативной памяти [23, 24], а также к вопросам повышения ее отказоустойчивости [25].

В процессоре сопоставлений ППВС применяется тернарная ассоциативная память — этот тип ассоциативной памяти позволяет проводить поиск по содержимому, используя для поиска ключ и маску, которая маскирует отдельные биты или байты ключа (замаскированные биты считаются безусловно совпавшими).

В том случае, если на момент прихода "парный" токен отсутствует, его ключ и маска "записываются" в процессор сопоставлений. При "совпадении" ключей токена происходит формирование пакета, который выдается на исполнение в ИУ и, в общем случае, проис-

ходит удаление этих токенов из процессора сопоставлений. Токен, который приходит на вход ПС, называется "верхним", а тот, который ожидает в ПС, — "нижним".

Базовая архитектура процессора сопоставлений (см. рисунок) включает следующие компоненты:

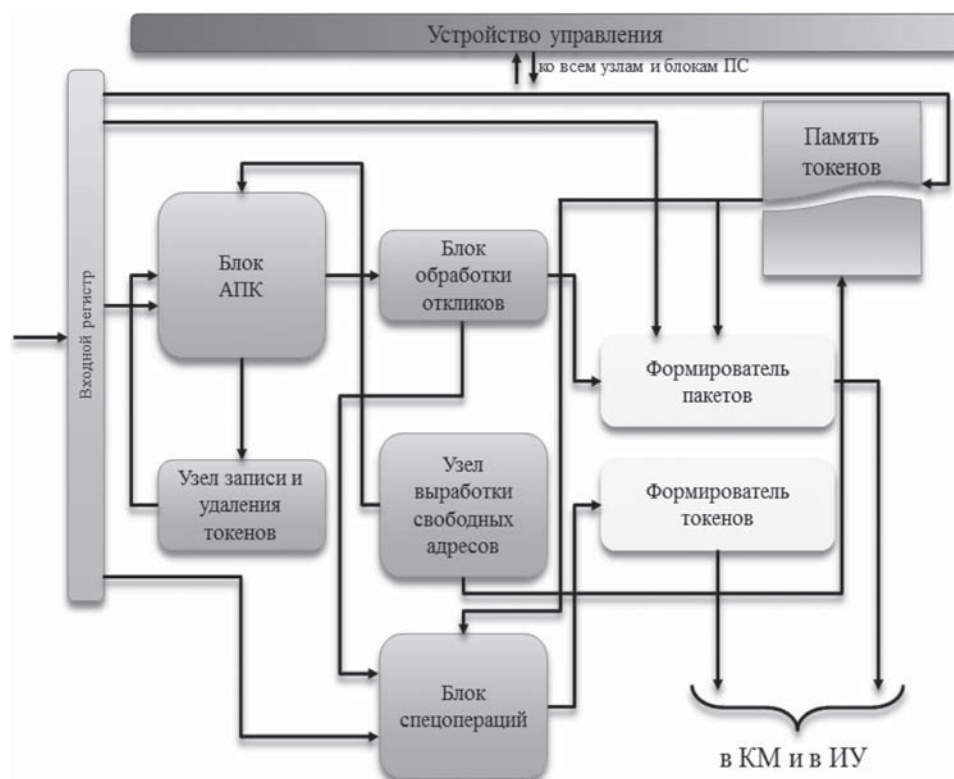
- входной регистр. Служит для хранения поступившего токена до момента его удаления или "записи";
- устройство управления (УУ). Анализирует служебные поля токенов и вырабатывает управляющие сигналы ("Запись", "Стирание", "Поиск", "Формирование пакета" и др.) для остальных узлов ПС. Для токенов, у которых кратность (определяет число взаимодействий токенов друг с другом) больше единицы, этот блок осуществляет расчет нового значения поля "кратность" после взаимодействия;
- ассоциативная память ключей (АПК). Сохраняет ключи и маски пришедших ранее токенов. Выполняет сопоставление ключей хранящихся и поступивших на вход ПС токенов;
- память дескрипторов (ПД). Содержит адреса токенов (дескрипторы), хранящихся в памяти токенов (ПТ), а также значения ряда полей токена для ускорения доступа к ним со стороны устройства управления;

• память токенов (ПТ) — массив памяти, в которой хранятся токены целиком, включая поля данных;

• формирователь пакетов (ФП). При совпадении ключей двух токенов формируется пакет, содержащий данные двух токенов. Пакет направляется на исполнительное устройство;

• блок специальных операций (БСО). Выполняет обработку сопоставившихся токенов, без образования пакетов и, соответственно, использования ИУ.

В ППВС некоторые функции, присущие операционным системам, реализованы на аппаратном уровне. Например, в традиционных вычислительных системах оперативная па-



Блок-схема процессора сопоставлений

мять выполняет роль исключительно хранения данных. В ППВС процессор сопоставлений (фактически аналог оперативной памяти в традиционной вычислительной системе) выполняет роль организатора вычислений и управляет ходом вычислительного процесса. Поскольку ПС является ключевым блоком ППВС, то и организацию динамического контроля целостности вычислительного процесса целесообразно и эффективно реализовать именно в ПС.

3. Средства контроля и диагностики процессора сопоставлений

Для динамического контроля вычислительного процесса в процессоре сопоставлений созданы средства, позволяющие обнаружить ошибку в ходе выполнения программы в точках контроля, своевременно ее зафиксировать, при возможности устранить, а также обеспечить выдачу информации о месте и условиях возникновения этой ошибки.

Под контролем ошибок в ходе выполнения программы имеется в виду комплекс мер, отвечающих за:

- предотвращение возникновения критических ситуаций (которые могут привести к некорректной работе системы и, как следствие, некорректным результатам вычислений), вызванных некорректными входными данными или ошибками в логике программы;
- корректное прекращение (при необходимости) работы системы и удаление "остатков" задачи при обнаружении критических ошибок;
- информирование пользователя вычислительной системы о характере и причинах возникновения ошибки.

Аппаратные средства контроля ошибок в ходе выполнения программы, реализованные в ПС, обеспечивают наблюдение как за ошибками программиста (такими как обращение в недействительную область виртуальной памяти программы, посылка некорректных типов токенов и т. п.), так и за возможными ошибками аппаратуры (некорректное сопоставление токенов, нарушение этапов и т. п.) и включают в себя:

- контроль этапов. На каждой стадии обработки "верхнего" токена осуществляется контроль номера этапа этого токена в соответствии с таблицей активных этапов. "Нижние" токены, которые сопоставились с "верхним", также проверяются на активность этапа;

- контроль целостности адресного пространства. Функции ОС, которые в традиционных вычислительных системах обеспечивают работу с виртуальным адресным пространством, в ППВС поддерживаются АПК на аппаратном уровне. Выполняется контроль ячеек ассоциативной памяти ключей — контролируется доступ к пустым ячейкам АПК путем сравнения содержимого регистра совпадений АПК (который заполняется по результатам сопоставления "верхнего" и "нижних" токенов) с содержимым регистра активных ячеек АПК. В отличие от традиционных ВС в ППВС у аппаратуры ПС имеется информация о том, содержит ли ячейка АПК ключ токена или она "пустая". Помимо контроля эта информация обеспечивает оптимизацию процесса сопоставлений токенов (что позволяет снизить энергопотребление ассоциативной памяти) при приходе нового токена на вход ПС, поскольку сопоставление токенов выполняется только с ключами "нижних" токенов, а не со всеми ячейками АПК;
- контроль кратности токенов. Данный вид контроля применяется при стандартной работе с токенами, использующими "механизм" кратности. Этот механизм позволяет вместо многократной (N) посылки одного и того же токена, посылать один токен с кратностью N . Проверка осуществляется непосредственно в устройстве управления ПС или в блоке специальных операций ПС. При любом изменении кратности как "верхнего", так и "нижнего" токенов происходит проверка выхода за границы допустимых значений;
- контроль маски. Отвечает за проверку наличия номера маски, содержащейся в токене, в памяти масок ПС. Номер маски в токене служит для уменьшения разрядности токена — вместо поразрядной маски, которая дает заметные преимущества программисту в плане гибкости организации вычислительного пространства задачи, передается только номер маски. Сама маска предварительно записывается в память масок процессора сопоставлений, что позволяет в девять раз уменьшить размер поля "маска" в структуре токена;
- контроль сопоставления токенов. Состоит из нескольких стадий. Во-первых, непосредственно в АПК происходит проверка на запрещенные комбинации полей токенов (более подробно описаны ниже). Во-вторых,

в устройстве управления ПС осуществляется проверка и сравнение различных полей "верхнего" и "нижнего" (извлеченных из памяти дескрипторов) токенов. Фактически, выполняется повтор операции сопоставления в АПК для того, чтобы убедиться, что данные в АПК не претерпели изменений;

- контроль формирования пакетов. Осуществляется проверка на соответствие полей токенов, участвующих в процессе формирования пакета. В формирователь пакетов "нижний" токен поступает из памяти токенов (а не из АПК), а "верхний" токен — с входного регистра. Проверка позволяет убедиться в правильности данных, которые находятся как в АПК, так и в памяти дескрипторов;
- контроль блока специальных операций. Блок специальных операций предназначен для выполнения простых арифметических и логических операций над данными токенов без образования пакетов и, соответственно, без использования ИУ. В результате выполнения операций происходит изменение уже имеющихся токенов. Новые токены или пакеты образуются только по завершению серии операций, например, при окончании суммирования набора данных. В данном блоке происходит как проверка запрещенных комбинаций (в соответствии с системой команд ПС) полей токенов при их взаимодействии друг с другом, так и стандартная проверка на ошибки, которая выполняется в функциональных арифметико-логических устройствах;
- контроль целостности адресного пространства задачи. В ППВС на аппаратном уровне реализовано разделение адресного пространства задач друг от друга. Несмотря на то, что у пользователя отсутствует возможность создать токен в программе или подать его извне с адресом, соответствующим другой программе, данный контроль осуществляется для предотвращения возможного появления аппаратного сбоя, вызванного различными внешними факторами.

Аппаратный динамический контроль целостности вычислительного процесса в ППВС позволяет более полно контролировать выполнение программы, поскольку он не ограничен статическим анализом, как компиляторы в традиционных системах. Важную роль в таком контроле играют запрещенные комбинации токенов, поскольку в ППВС они являются основным средством передачи сообщений и

синхронизации между программными узлами. Остановимся на этом подробнее.

Взаимодействие двух "стандартных" токенов с бесконечной кратностью. "Стандартные" токены являются основой вычислительного процесса, чья функция заключается в формировании пакета из двух данных, принадлежащих сопоставившимся токенам. При этом при взаимодействии двух токенов число формируемых пакетов равно значению их минимальной кратности. Это условие делает невозможным сопоставление токенов с бесконечными кратностями, так как формирование бесконечного числа пакетов априори приведет к "зависанию" вычислительной системы или, по крайней мере, к бесконечному выполнению "ненужных" вычислений.

Взаимодействие токенов управляющего класса друг с другом. К таким токенам относятся токены "Проверка", "Стирание", "Проверка с сохранением", "Программная косвенность" и "Аппаратная косвенность":

- назначение и функция токена "Проверка" — это контроль наличия определенного токена в АПК процессора сопоставлений и формирование пакета даже при отсутствии сопоставлений (в этом случае в пакете сообщается о том, что искомого токена в АПК на момент прихода токена "Проверка" не было). Так как по логике работы, заложенной в токен "Проверка", он не сохраняется в АПК, любое его взаимодействие с токенами управляющего класса может привести к нарушению этой логики — он может быть удален или перенаправлен на другой узел до выполнения своей основной задачи;
- взаимодействие токена "Проверка с сохранением" с токенами управляющего класса. Токен "Проверка с сохранением" отличается от токена "Проверка" тем, что после первоначальной проверки памяти на наличие определенного токена он не только формирует пакет, но и оставляет свою копию в АПК, продолжая "отслеживать" поступление токенов с определенным ключом. Таким образом, назначением данного токена является контроль адресного пространства задачи путем "перехвата" стандартных токенов по заданным координатам. Для выполнения этого функционала данный токен не должен удаляться или перемещаться, меняя свой ключ. Именно этим обусловлен запрет на указанные взаимодействия;
- взаимодействие токенов "Косвенность" друг с другом. Косвенность — это механизм "пе-

рехвата" данных, направленных для обработки в определенный программный узел, и перенаправление их для обработки в другой программный узел. Данный механизм хорошо зарекомендовал себя для контроля "границ" в сеточных задачах. Этот механизм позволяет самой программе при наступлении определенного состояния изменить стандартный ход выполнения вычислительного процесса. Токен "Программная косвенность" при "перехвате" другого токена формирует пакет с данным из перехваченного токена и отправляет его на обработку в ИУ. При аппаратной косвенности в "перехваченном" токене меняется номер программного узла, и обновленный токен направляется на повторную обработку в процессор сопоставлений, минуя ИУ. Исходя из описанного назначения при взаимодействии двух токенов "косвенность" отсутствует данное, которое требуется перенаправить на другой программный узел, а также не ясно, какой из двух токенов является более приоритетным;

- взаимодействие токенов "Стирание" и "Чистка". Удаление данных из памяти процессора сопоставлений происходит двумя способами: естественным и принудительным. Естественное удаление токена происходит при достижении его кратности нулевого значения. При принудительном удалении указывается конкретный токен или группа токенов, которые должны быть удалены из памяти процессора сопоставлений вне зависимости от значения их кратности. Этот механизм поддерживается двумя командами "Стирание" и "Чистка". Их отличие заключается в том, что токен "Стирание" после поиска и обработки взаимодействий при ненулевой кратности записывается в память (а токен "Чистка" — нет) и продолжает выполнять свою роль (очистки памяти), не допуская сохранения в ней вновь пришедших токенов, а также их взаимодействия с другими токенами (в виду более высокого приоритета). Взаимодействие токенов "Стирание" и "Чистка" между собой, а также двух токенов "Стирание" противоречит описанному функционалу;
- бесконечная кратность "Стартового" токена. Назначение "Стартового" токена — инициализация вычислений. При поступлении на вход процессора сопоставлений он, минуя этап поиска сопоставлений в ассоциативной памяти ключей, инициирует

формирование пакета с одним данным, которое он и содержит. Число формируемых пакетов равняется значению кратности токена. "Стартовый" токен не записывается в память. Бесконечная кратность при такой логике обработки токена приводит к формированию бесконечного числа пакетов, что является ошибкой, поэтому данная комбинация полей токена является запрещенной и контролируется на аппаратном уровне.

При обнаружении описанных выше ситуаций возникают соответствующие флаги ошибки. Токены, которые участвовали в зафиксированной ситуации, выдаются с признаком ошибки на ХОСТ-машину для дальнейшего анализа и устранения ошибки программистом. Вместе с ними выдается и содержимое регистра ошибок ПС.

Заключение

При создании высокопроизводительных вычислительных систем большое внимание уделяется вопросам обеспечения бесперебойности работы и контролю целостности вычислительного процесса. В традиционных вычислительных системах данные вопросы решаются в основном на уровне программного обеспечения, такого как ОС или оптимизирующие компиляторы.

ППВС, работа над которой ведется в ИППМ РАН, реализует потоковую модель вычислений с динамически формируемым контекстом. В ППВС часть функций, присущих операционным системам, реализована на аппаратном уровне. ПС (аналог оперативной памяти в традиционной вычислительной системе) выполняет роль организатора вычислений и управляет ходом вычислительного процесса, и поэтому динамический контроль целостности вычислительного процесса реализован в ПС, что позволяет оптимизировать аппаратные затраты на его организацию и повысить эффективность его работы.

Аппаратный динамический контроль целостности вычислительного процесса в ППВС позволяет контролировать выполнение программы более полно, чем это позволяют компиляторы в традиционных системах, ограниченные статическим анализом программы, и своевременно обнаруживать ошибки и сбои, возникшие как по вине программиста, так и вследствие сбоя аппаратуры.

Список литературы

1. **Top500**. URL: www.top500.org (дата обращения: 01.07.21).
2. **Monroe D.** Fugaku takes the lead // *Communications of the ACM*. January 2021. Vol. 64, N. 1. P. 16–18. DOI: 10.1145/3433954.
3. **Бобков С. Г.** Пути и методы повышения производительности микропроцессоров // *Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС)*. 2020. Вып. 4. С. 127–133. DOI: 10.31114/2078-7707-2020-4-127-133.
4. **Бобков С. Г., Басаев А. С.** Методы и средства аппаратного обеспечения высокопроизводительных микропроцессорных систем. Москва: Рекламно-издательский центр "Техносфера", 2021. 264 с.
5. **Böhm A. P. W.** Dataflow and hybrid dataflow architecture summary // *Parallel computer systems*, Rebecca Koskela and Margaret Simmons (Eds.). ACM, New York, NY, USA, 1990. P. 281–286. DOI: 10.1145/100215.100286.
6. **Lee B., Hurson A. R.** Issues in Dataflow Computing // *Advances in computers*. 1993. Vol. 37. P. 285–333.
7. **Silc J., Robic B., Ungerer T.** Asynchrony in parallel computing: From dataflow to multithreading // *Parallel and Distributed Computing Practices*. 1998. Vol. 1, N. 1. P. 3–30.
8. **Левченко Н. Н., Окунев А. С., Стемпковский А. Л.** Преимущества потоковой модели вычислений // *Проблемы разработки перспективных микро- и наноэлектронных систем*. 2018. Вып. 3. С. 24–30. DOI: 10.31114/2078-7707-2018-3-24-30.
9. **Тельпухов Д. В.** Разработка методов генетического синтеза сбоеустойчивых комбинационных схем // *Проблемы разработки перспективных микро- и наноэлектронных систем*. 2018. Вып. 1. С. 45–49. DOI: 10.31114/2078-7707-2018-1-45-49.
10. **Gavrilov S. V., Gurov S. I., Zhukova T. D., Rukhlov V. S., Ryzhova D. I., Tel'pukhov D. V.** Methods to increase fault tolerance of combinational integrated microcircuits by redundancy coding // *Computational Mathematics and Modeling*. 2017. Vol. 28, N. 3. P. 400–406.
11. **Stempkovskiy A., Telpukhov D., Gurov S., Zhukova T., Demeneva A.** R-code for concurrent error detection and correction in the logic circuits // *Proceedings of the 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering, ElConRus 2018*. DOI: 10.1109/ElConRus.2018.8317365.
12. **Stempkovsky A., Telpukhov D., Solovyev R.** Synthesis of Approximate Combinational Circuits based on Logic Regression Approach // *2020 IEEE East-West Design & Test Symposium (EWDTS)*. Varna, Bulgaria, Sept 4–7. 2020. P. 1–5, DOI: 10.1109/EWDTS50664.2020.9224981.
13. **Stempkovskii A. L., Tel'pukhov D. V., Zhukova T. D., Demeneva A. I., Nadolenko V. V., Gurov S. I.** Synthesis of a Concurrent Error Detection Circuit Based on the Spectral R-Code with the Partitioning of Outputs into Groups // *Russian Microelectronics*. 2019. Vol. 48. P. 240–249. DOI: 10.1134/S1063739719040097.
14. **Solovyev R. A., Stempkovsky A. L., Telpukhov D. V.** Study of Fault Tolerance Methods for Hardware Implementations of Convolutional Neural Networks // *Optical Memory and Neural Networks*. 2019. Vol. 28. P. 82–88. DOI: 10.3103/S1060992X19020103.
15. **Johnson D.** The Intel 432: A VLSI Architecture for Fault-Tolerant Computer Systems // *Computer*. August 1984. Vol. 17, N. 8. P. 40–48. DOI: 10.1109/MC.1984.1659216.
16. **Johnson D., Budde D., Carson D., Peterson C.** Intel iAPX 432: VLSI building blocks for a fault-tolerant computer // *AFIPS '83: Proceedings of the May 16-19, 1983, national computer conference*. ACM, New York, NY, USA, 1983. P. 531–537. DOI: 10.1145/1500676.1500743.
17. **Lustig D., Martonosi M.** Reducing GPU offload latency via finegrained CPU-GPU synchronization // *Proceedings of the 2013 IEEE 19th International Symposium on High Performance Computer Architecture (HPCA)*. 2013. P. 354–365. DOI: 10.1109/HPCA.2013.6522332.
18. **Змеев Д. Н., Левченко Н. Н., Окунев А. С., Стемпковский А. Л.** Влияние особенностей модели вычислений и архитектуры на надежность параллельной потоковой вычислительной системы // *Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС)*. 2020. Вып. 1. С. 64–69. DOI: 10.31114/2078-7707-2020-1-64-69.
19. **Бобков С. Г., Левченко Н. Н., Окунев А. С.** Параллельный потоковый процессор на новых архитектурных принципах для решения широкого круга задач // *Наноиндустрия*. 2020. Т. 13, № S5-2(102). С. 285–296. DOI: 10.22184/1993-8578.2020.13.S5.285.296.
20. **Стемпковский А. Л., Левченко Н. Н., Окунев А. С., Цветков В. В.** Параллельная потоковая вычислительная система — дальнейшее развитие архитектуры и структурной организации вычислительной системы с автоматическим распределением ресурсов // *Информационные технологии*. 2008. № 10. С. 2–8.
21. **Климов А. В., Левченко Н. Н., Окунев А. С., Стемпковский А. Л.** Суперкомпьютеры, иерархия памяти и потоковая модель вычислений // *Программные системы: теория и приложения: электронный научный журнал*. 2014. Т. 5, № 1(19). С. 15–36.
22. **Змеев Д. Н., Климов А. В., Левченко Н. Н., Окунев А. С., Стемпковский А. Л.** Потоковая модель вычислений как парадигма программирования будущего // *Информатика и ее применения*. 2015. Т. 9, Вып. 4. С. 29–36.
23. **Karthik S., Karthick D., Sanjaya M. V., Madhav R.** Design and Implementation of a Low Power Ternary Content Addressable Memory (TCAM) // *Proceedings of 2020 International SoC Design Conference (ISOC)*. 2020. P. 15–16. DOI: 10.1109/ISOC50952.2020.9333092.
24. **Choi W., Lee K., Park J.** Low Cost Ternary Content Addressable Memory Using Adaptive Matchline Discharging Scheme // *Proceedings of 2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. 2018. P. 1–4. DOI: 10.1109/ISCAS.2018.8351461.
25. **Antonyuk A. V., Stenin V. Y.** The Fault Tolerant CMOS Logical Element of Matching for a Content-Addressable Memory // *Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2018)*, Kazan, Russia, Sept 14–17, 2018. P. 1–4. DOI: 10.1109/EWDTS.2018.8524634.

N. N. Levchenko, Head of Department, e-mail: nick@ippm.ru,

D. N. Zmejev, Researcher, e-mail: zmejevdn@list.ru,

Institute for Design Problems in Microelectronics of Russian Academy of Sciences,
Moscow, Russian Federation

Dynamic Control of Computation Consistency in the Parallel Dataflow Computing System

When developing high-performance multiprocessor computing systems, much attention is paid to ensuring uninterrupted operation, both in terms of hardware and software. In traditional computing systems, software is the main focus in addressing these issues. The article discusses the solution to the issue of ensuring uninterrupted operation for the parallel dataflow computing system (PDCS), which implements the dataflow computational model with a dynamically formed context. Due to the features of the PDCS, it is proposed to implement this type of control in hardware, which will increase its efficiency, since the computational process will be controlled in dynamics, and not only in statics.

Keywords: dataflow, computational model, matching processor, hardware computation control

DOI: 10.17587/it.27.625-633

References

1. **Top500**, available at: www.top500.org (date of access: 01.07.21).
2. **Monroe D.** Fugaku takes the lead, *Communications of the ACM*, January 2021, vol. 64, no. 1, pp. 16–18, DOI:10.1145/3433954.
3. **Bobkov S. G.** Directions and methods for improving the performance of microprocessors, *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem (MJeS)*, 2020, no. 4, pp. 127–133, DOI: 10.31114/2078-7707-2020-4-127-133. (in Russian).
4. **Bobkov S. G., Basaev A. S.** *Hardware methods and tools for highperformance microprocessor systems*, Moscow, Reklamno-izdatel'skij centr "Tehnosfera", 2021, 264 p. (in Russian).
5. **Böhm A. P. W.** Dataflow and hybrid dataflow architecture summary, *Parallel computer systems*, Rebecca Koskela and Margaret Simmons (Eds.), ACM, New York, NY, USA, 1990, pp. 281–286, DOI:10.1145/100215.100286.
6. **Lee B., Hurson A. R.** Issues in Dataflow Computing, *Advances in computers*, 1993, vol. 37, pp. 285–333.
7. **Silc J., Robic B., Ungerer T.** Asynchrony in parallel computing: From dataflow to multithreading, *Parallel and Distributed Computing Practices*, 1998, vol. 1, no. 1, pp. 3–30.
8. **Levchenko N. N., Okunev A. S., Stempkovskij A. L.** Advantages of Dataflow Computing Model, *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem (MJeS)*, 2018, iss. 3, pp. 24–30, DOI:10.31114/2078-7707-2018-3-24-30 (in Russian).
9. **Tel'puhov D. V.** Development of Methods for Genetic Synthesis of Fault-Tolerant Logic circuits, *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem (MJeS)*, 2018, iss. 1, pp. 45–49. DOI: 10.31114/2078-7707-2018-1-45-49 (in Russian).
10. **Gavrilov S. V., Gurov S. I., Zhukova T. D., Rukhlov V. S., Ryzhova D. I., Tel'pukhov D. V.** Methods to increase fault tolerance of combinational integrated microcircuits by redundancy coding, *Computational Mathematics and Modeling*, 2017, vol. 28, no. 3, pp. 400–406.
11. **Stempkovskiy A., Telpukhov D., Gurov S., Zhukova T., Demeneva A.** R-code for concurrent error detection and correction in the logic circuits, *Proceedings of the 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering, ElConRus 2018*, DOI: 10.1109/ElConRus.2018.8317365.
12. **Stempkovsky A., Telpukhov D., Solovyev R.** Synthesis of Approximate Combinational Circuits based on Logic Regression Approach, *2020 IEEE East-West Design & Test Symposium (EWDTS)*, Varna, Bulgaria, Sept 4-7, 2020, pp. 1–5, DOI: 10.1109/EWDTS50664.2020.9224981.
13. **Stempkovskii A. L., Tel'pukhov D. V., Zhukova T. D., Demeneva A. I., Nadolenko V. V., Gurov S. I.** Synthesis of a Concurrent Error Detection Circuit Based on the Spectral R-Code with the Partitioning of Outputs into Groups, *Russian Microelectronics*, 2019, vol. 48, pp. 240–249, DOI:10.1134/S1063739719040097.
14. **Solovyev R. A., Stempkovsky A. L., Telpukhov D. V.** Study of Fault Tolerance Methods for Hardware Implementations of Convolutional Neural Networks, *Optical Memory and Neural Networks*, 2019, vol. 28, pp. 82–88.
15. **Johnson D.** The Intel 432: A VLSI Architecture for Fault-Tolerant Computer Systems, *Computer*, August 1984, vol. 17, no. 8, pp. 40–48.
16. **Johnson D., Budde D., Carson D., Peterson C.** Intel iAPX 432: VLSI building blocks for a fault-tolerant computer, *AFIPS '83: Proceedings of the May 16-19, 1983, national computer conference*, ACM, New York, NY, USA, 1983, pp. 531–537.
17. **Lustig D., Martonosi M.** Reducing GPU offload latency via finegrained CPU-GPU synchronization, *Proceedings of the 2013 IEEE 19th International Symposium on High Performance Computer Architecture (HPCA)*, 2013, pp. 354–365.
18. **Zmeev D. N., Levchenko N. N., Okunev A. S., Stempkovskij A. L.** Impact of features of the computing model and architecture on the reliability of the parallel dataflow computing system, *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem (MJeS)*, 2020, iss. 1, pp. 64–69, DOI: 10.31114/2078-7707-2020-1-64-69 (in Russian).
19. **Bobkov S. G., Levchenko N. N., Okunev A. S.** Parallel dataflow processor based on new architectural principles for solving a wide range of tasks, *Nanoindustrya*, 2020, vol. 13, no. S5-2(102), pp. 285–296, DOI: 10.22184/1993-8578.2020.13.5s.285.296 (in Russian).
20. **Stempkovskij A. L., Levchenko N. N., Okunev A. S., Cvetkov V. V.** Parallel Dataflow Computing System the Further Development of Architecture and the Structural Organization of the Computing System with Automatic Distribution of Resources, *Zhurnal "INFORMACIONNYYE TEHNOLOGII"*, 2008, no. 10, pp. 2–7 (in Russian).
21. **Klimov A. V., Levchenko N. N., Okunev A. S., Stempkovskij A. L.** Supercomputers, memory hierarchy and dataflow computation model, *Programmye sistemy: teoriya i prilozheniya: elektronnyj nauchnyj zhurnal*, 2014, vol. 5, no. 1 (19), pp. 15–36 (in Russian).
22. **Zmeev D. N., Klimov A. V., Levchenko N. N., Okunev A. S., Stempkovskij A. L.** Dataflow computing model as a paradigm of future mainstream of software development, *Informatika i ejo primeneniya*, 2015, vol. 9, iss. 4, pp. 29–36 (in Russian).
23. **Karthik S., Karthick D., Sanjaya M. V., Madhav R.** Design and Implementation of a Low Power Ternary Content Addressable Memory (TCAM), *Proceedings of 2020 International SoC Design Conference (ISOC)*, 2020, pp. 15–16, DOI: 10.1109/ISOC50952.2020.9333092.
24. **Choi W., Lee K., Park J.** Low Cost Ternary Content Addressable Memory Using Adaptive Matchline Discharging Scheme, *Proceedings of 2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2018, pp. 1–4, DOI: 10.1109/ISCAS.2018.8351461.
25. **Antonyuk A. V., Stenin V. Y.** The Fault Tolerant CMOS Logical Element of Matching for a Content-Addressable Memory, *Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2018)*, Kazan, Russia, Sept 14–17, 2018, pp. 1–4, DOI: 10.1109/EWDTS.2018.8524634.

В. Н. Тарасов, д-р техн. наук, проф., зав. каф., e-mail: veniamin_tarasov@mail.ru,

Н. Ф. Бахарева, д-р техн. наук, проф., зав. каф., e-mail: nadin1956_04@inbox.ru,

Поволжский государственный университет телекоммуникаций и информатики, г. Самара

Математическая модель задержки в сетях связи на основе СМО с запаздыванием во времени

При математическом моделировании современных компьютерных сетей, сетей телекоммуникаций, транспортных потоков, логистики и многих других систем широко используются методы теории массового обслуживания. В свою очередь, в исследованиях систем массового обслуживания (СМО) G/G/1 с произвольными законами распределений интервалов между соседними требованиями входящего потока и времени их обслуживания часто применяется метод спектрального разложения (МСР) решения интегрального уравнения Линдли. В основе этого метода лежит поиск нулей и полюсов построенного спектрального разложения в виде некоторой дробно-рациональной функции с привлечением численных методов для определения корней многочленов. При этом коэффициенты многочлена в числителе разложения выражаются через неизвестные параметры используемых для описания СМО законов распределений. В случае исследования телетрафика обычно эти неизвестные параметры законов распределений могут быть определены через числовые характеристики интервалов между пакетами трафика методом моментов. Целью данной статьи является представление принципиально новой математической модели системы, образованной двумя потоками со сдвинутыми вправо законами распределений. Такое возможно только для тех законов распределений вероятностей, функции плотности которых преобразуемы по Лапласу. Основные преимущества таких систем (назовем их системами с запаздыванием во времени) в том, что они обеспечивают меньшее время задержки в очереди по сравнению с обычными системами, а также расширяют диапазон параметров трафика. В статье приведены полученные результаты по средней задержке требований в очереди для системы с экспоненциальным и гиперэрланговским распределениями, алгоритм расчета средней задержки и результаты вычислительных экспериментов в пакете MathCAD.

Ключевые слова: система массового обслуживания, средняя задержка, интегральное уравнение Линдли, преобразование Лапласа, метод спектрального разложения

Введение

При рассмотрении систем G/G/1 будем опираться на метод спектрального разложения решения интегрального уравнения Линдли, который сыграл большую роль при их исследовании в теории массового обслуживания. Наиболее доступно этот метод на конкретных примерах изложен в монографии по теории массового обслуживания [1]. Этот метод также использован в работах [2–5].

Настоящая статья посвящена анализу систем массового обслуживания (СМО) с запаздыванием во времени, т. е. системы, образованной сдвинутыми вправо от нуля экспоненциальным и гиперэрланговским распределениями.

В работе [6] приведены результаты по исследованию системы M/M/1 с запаздыванием во времени со сдвинутыми вправо от нулевой точки экспоненциальными входными распределениями, которую в отличие от классической системы обозначим M⁻/M⁻/1. Эти результаты получены методом спектрального разложения. Здесь и далее верхний индекс "—" будет означать операцию сдвига закона распределения.

В работе [6] показано, что средняя задержка требования в очереди в системе M⁻/M⁻/1 меньше, чем в классической системе M/M/1 при одинаковом коэффициенте загрузки за счет того, что коэффициенты вариации времен поступления c_λ и обслуживания c_μ становятся меньше единицы из-за ввода параметра запаздывания $t_0 > 0$ в законы распределения. Таким образом, операция сдвига закона распределения трансформирует классическую систему M/M/1 в немарковскую систему G/G/1.

Результаты работы [6] с применением методов работы [1] позволили перенести метод спектрального разложения на системы $H_2^-/H_2^-/1$, $H_2^-/M^-/1$, $M^-/H_2^-/1$, $E_2^-/E_2^-/1$, $E_2^-/M^-/1$, $M^-/E_2^-/1$ и $HE_2^-/HE_2^-/1$ с запаздыванием [7–11]. Рассматриваемая в статье СМО, образованная известными сдвинутыми законами распределений M, HE₂ относится к типу G/G/1.

Аппроксимативные методы для законов распределений подробно описаны в работах [12, 13], а последние наиболее интересные исследования по СМО представлены в статьях [14–25].

В теории массового обслуживания исследования систем G/G/1 актуальны в связи с тем, что они активно используются в современной теории телетрафика, в моделировании транспортных потоков и т. п. Как известно, нельзя получить решения по средней задержке для таких систем в конечном виде для общего случая. При использовании метода спектрального разложения для определения средней задержки будем придерживаться классического подхода теории массового обслуживания [1]. По методике спектрального разложения необходимо найти закон распределения задержки для рассматриваемой системы через спектральное разложение вида

$$F_{\lambda}^*(-s)F_{\mu}^*(s) - 1 = \omega_+(s)/\omega_-(s),$$

где $F_{\lambda}^*(s)$, $F_{\mu}^*(s)$ — преобразования Лапласа функций плотности распределения интервалов входного потока $f_{\lambda}(t)$ и времени обслуживания $f_{\mu}(t)$ соответственно; $\omega_+(s)$ и $\omega_-(s)$ — некоторые дробно-рациональные функции от s , которые должны быть сконструированы с учетом специальных условий [1].

Постановка и решение задачи

Рассмотрим систему со сдвинутыми законами распределений интервалов между требованиями входного потока и времени обслуживания, заданными функциями плотности вида

$$f_{\lambda}(t) = \lambda e^{-\lambda(t-t_0)}; \quad (1)$$

$$f_{\mu}(t) = 4q\mu_1^2(t-t_0)e^{-2\mu_1(t-t_0)} + 4(1-q)\mu_2^2(t-t_0)e^{-2\mu_2(t-t_0)}. \quad (2)$$

Такую СМО в отличие от обычной системы M/HE₂/1 обозначим M⁻/HE₂⁻/1. Функция плотности распределения вероятностей (2) является гиперэрланговским законом распределения второго порядка общего вида. Ставится задача определения средней задержки требований в очереди в определенной выше системе с выводом ее математической модели и проведения вычислительных экспериментов. Для решения поставленной задачи докажем предварительно следующую теорему.

Теорема. Спектральное разложение

$$F_{\lambda}^*(-s)F_{\mu}^*(s) - 1 = \omega_+(s)/\omega_-(s)$$

решения интегрального уравнения Линдли для системы M⁻/HE₂⁻/1 со сдвинутыми рас-

пределениями (1) и (2) полностью совпадает со спектральным разложением для обычной системы M/HE₂/1 и имеет вид

$$\frac{\omega_+(s)}{\omega_-(s)} = \left(\frac{\lambda}{\lambda-s}\right) \left[q \left(\frac{2\mu_1}{s+2\mu_1} \right)^2 + (1-q) \left(\frac{2\mu_2}{s+2\mu_2} \right) \right] - 1. \quad (3)$$

Доказательство.

Запишем преобразования Лапласа функций (1) и (2):

$$F_{\lambda}^*(s) = \frac{\lambda}{s+\lambda} e^{-t_0 s};$$

$$F_{\mu}^*(s) = \left[q \left(\frac{2\mu_1}{s+2\mu_1} \right)^2 + (1-q) \left(\frac{2\mu_2}{s+2\mu_2} \right)^2 \right] e^{-t_0 s}.$$

Здесь мы использовали теорему о запаздывании в преобразованиях Лапласа.

Тогда спектральное разложение решения интегрального уравнения Линдли для системы M⁻/HE₂⁻/1 примет следующий вид:

$$F_{\lambda}^*(-s)F_{\mu}^*(s) - 1 = \frac{\lambda}{\lambda-s} e^{t_0 s} \times$$

$$\times \left[q \left(\frac{2\mu_1}{s+2\mu_1} \right)^2 + (1-q) \left(\frac{2\mu_2}{s+2\mu_2} \right)^2 \right] e^{-t_0 s} - 1 = \quad (4)$$

$$= \frac{\lambda}{\lambda-s} \left[q \left(\frac{2\mu_1}{s+2\mu_1} \right)^2 + (1-q) \left(\frac{2\mu_2}{s+2\mu_2} \right)^2 \right] - 1.$$

Как видно, экспоненты со степенями с противоположными знаками обнуляются, и операция сдвига тем самым нивелируется. Таким образом, убеждаемся, что такое разложение инвариантно к операции сдвига в законах распределения. Последнее выражение идентично (3). Теорема доказана.

Опираясь на теорему, продолжим разложение. Выражение, стоящее в квадратных скобках, представим в виде

$$\left[q \left(\frac{2\mu_1}{s+2\mu_1} \right)^2 + (1-q) \left(\frac{2\mu_2}{s+2\mu_2} \right)^2 \right] =$$

$$= \frac{q(16\mu_1^2\mu_2^2 + 16\mu_1^2\mu_2 s + 4\mu_1^2 s^2)}{(s+2\mu_1)^2 (s+2\mu_2)^2} +$$

$$+ \frac{(1-q)(16\mu_1^2\mu_2^2 + 16\mu_1\mu_2^2s + 4\mu_2^2s^2)}{(s+2\mu_1)^2(s+2\mu_2)^2} =$$

$$= \frac{b_0 + b_1s + b_2s^2}{(s+2\mu_1)^2(s+2\mu_2)^2},$$

где вводимые промежуточные параметры имеют вид

$$b_0 = 16\mu_1^2\mu_2^2, b_1 = 16\mu_1\mu_2[q\mu_1 + (1-q)\mu_2],$$

$$b_2 = 4[q\mu_1^2 + (1-q)\mu_2^2].$$

Продолжая разложение, получим:

$$\frac{\omega_+(s)}{\omega_-(s)} =$$

$$= \frac{\lambda(b_0 + b_1s + b_2s^2) - (2\mu_1 + s)^2(2\mu_2 + s)^2(\lambda - s)}{(2\mu_1 + s)^2(2\mu_2 + s)^2(\lambda - s)} =$$

$$= \frac{s(s^4 + d_3s^3 + d_2s^2 + d_1s + d_0)}{(2\mu_1 + s)^2(2\mu_2 + s)^2(\lambda - s)} = \quad (5)$$

$$= \frac{s(s + \sigma_1)(s + \sigma_2)(s + \sigma_3)(s + \sigma_4)}{(2\mu_1 + s)^2(2\mu_2 + s)^2(\lambda - s)},$$

где коэффициенты многочлена четвертой степени в числителе выражения

$$d_0 = b_1\lambda + 16\mu_1\mu_2[\mu_1\mu_2 - \lambda(\mu_1 + \mu_2)],$$

$$d_1 = 16\mu_1\mu_2(\mu_1 + \mu_2) - 4\lambda(\mu_1^2 + 4\mu_1\mu_2 + \mu_2^2) + b_2\lambda,$$

$$d_2 = 4(\mu_1^2 + \mu_2^2) + 16\mu_1\mu_2 - 4\lambda(\mu_1 + \mu_2),$$

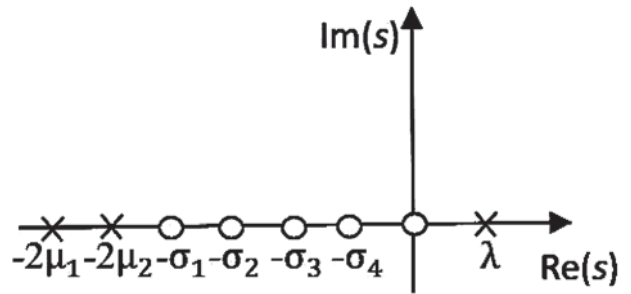
$$d_3 = 4(\mu_1 + \mu_2) - \lambda$$

положительны. Эти коэффициенты сформированы с помощью символьных операций MathCAD. Многочлен четвертой степени в случае стабильной системы

$$s^4 + d_3s^3 + d_2s^2 + d_1s + d_0 \quad (6)$$

с положительными коэффициентами имеет четыре действительных отрицательных корня, либо два действительных отрицательных корня и два комплексно-сопряженных корня с отрицательными вещественными частями. Тогда на основе доказанной теоремы окончательно спектральное разложение решения интегрального уравнения Линдли для системы М/НЕ₂/1 можно записать в виде

$$\frac{\omega_+(s)}{\omega_-(s)} = \frac{s(s + \sigma_1)(s + \sigma_2)(s + \sigma_3)(s + \sigma_4)}{(2\mu_1 + s)^2(2\mu_2 + s)^2(\lambda - s)}, \quad (7)$$



Нули и полюсы функции $\omega_+(s)/\omega_-(s)$ для систем М/НЕ₂/1 и М⁺/НЕ₂⁺/1

где для удобства отрицательные корни многочлена (6) обозначены $-\sigma_1, -\sigma_2, -\sigma_3, -\sigma_4$. На рисунке показана комплексная s -плоскость с нулями и полюсами функции $\omega_+(s)/\omega_-(s)$, где полюсы отмечены крестиками, а нули — кружками. Нули и полюсы нужны для построения функций $\omega_+(s)$ и $\omega_-(s)$ в отдельности.

Далее с учетом условий положений МСР [1] строим функции $\omega_+(s)$ и $\omega_-(s)$:

$$\omega_+(s) = \frac{s(s + \sigma_1)(s + \sigma_2)(s + \sigma_3)(s + \sigma_4)}{(2\mu_1 + s)^2(2\mu_2 + s)^2},$$

$$\omega_-(s) = \lambda - s.$$

Константа спектрального разложения

$$K = \lim_{s \rightarrow 0} \frac{\omega_+(s)}{s} =$$

$$= \lim_{s \rightarrow 0} \frac{(s + \sigma_1)(s + \sigma_2)(s + \sigma_3)(s + \sigma_4)}{(2\mu_1 + s)^2(2\mu_2 + s)^2} =$$

$$= \frac{\sigma_1\sigma_2\sigma_3\sigma_4}{16\mu_1^2\mu_2^2}.$$

Эта константа определяет вероятность того, что поступающее в систему требование заставит ее свободной.

Далее по методике спектрального разложения строим функцию

$$\Phi_+(s) = \frac{K}{\omega_+(s)} =$$

$$= \frac{\sigma_1\sigma_2\sigma_3\sigma_4(2\mu_1 + s)^2(2\mu_2 + s)^2}{16\mu_1^2\mu_2^2s(s + \sigma_1)(s + \sigma_2)(s + \sigma_3)(s + \sigma_4)}.$$

Отсюда преобразование Лапласа функции плотности задержки требований в очереди $W^*(s) = s\Phi_+(s)$ будет равно

$$W^*(s) = \frac{\sigma_1\sigma_2\sigma_3\sigma_4(2\mu_1 + s)^2(2\mu_2 + s)^2}{16\mu_1^2\mu_2^2s(s + \sigma_1)(s + \sigma_2)(s + \sigma_3)(s + \sigma_4)}, \quad (8)$$

Числовые характеристики распределений

Распределение	Первый начальный момент	Второй начальный момент	Квадрат коэффициента вариации
M	$1/\lambda$	$2/\lambda^2$	1
HE ₂	$\frac{q}{\mu_1} + \frac{(1-q)}{\mu_2}$	$\frac{3q}{2\mu_1^2} + \frac{3(1-q)}{2\mu_2^2}$	$\frac{\mu_1^2 - 2q\mu_2(\mu_1 - \mu_2) + q(1-2q)(\mu_1 - \mu_2)^2}{2[(1-q)\mu_1 + q\mu_2]^2}$
M ⁻	$\frac{1}{\lambda} + t_0$	$2\left(\frac{1}{\lambda^2} + \frac{t_0}{\lambda}\right) + t_0^2$	$\frac{1}{(1 + \lambda t_0)^2}$
HE ₂ ⁻	$\frac{q}{\mu_1} + \frac{(1-q)}{\mu_2} + t_0$	$t_0^2 + 2t_0\left[\frac{q}{\mu_1} + \frac{(1-q)}{\mu_2}\right] + \frac{3q}{2\mu_1^2} + \frac{3(1-q)}{2\mu_2^2}$	$\frac{\mu_1^2 - 2q\mu_2(\mu_1 - \mu_2) + q(1-2q)(\mu_1 - \mu_2)^2}{2[t_0\mu_1\mu_2 - q(\mu_1 - \mu_2) + \mu_1]^2}$

Таблица 2

Параметры сдвинутых распределений, полученные методом моментов

Распределение	Функция плотности	Параметры распределений		
M ⁻	$\lambda e^{-\lambda(t-t_0)}$	$\lambda = \frac{1}{\bar{\tau}_\lambda - t_0}$		
HE ₂ ⁻	$4q\mu_1^2(t-t_0)e^{-2\mu_1(t-t_0)} + 4(1-q)\mu_2^2(t-t_0)e^{-2\mu_2(t-t_0)}$	$q = \frac{1}{2} + \sqrt{\frac{1}{4} - \frac{3(\bar{\tau}_\mu - t_0)^2}{8[(\bar{\tau}_\mu - t_0)^2 + c_\mu^2 \bar{\tau}_\mu^2]}}$	$\mu_1 = \frac{2q}{(\bar{\tau}_\mu - t_0)}$	$\mu_2 = \frac{2(1-q)}{(\bar{\tau}_\mu - t_0)}$

а средняя задержка $\bar{W} = -\frac{dW^*(s)}{ds}$ в точке $s = 0$ будет равна

$$\bar{W} = \frac{1}{\sigma_1} + \frac{1}{\sigma_2} + \frac{1}{\sigma_3} + \frac{1}{\sigma_4} - \frac{1}{\mu_1} - \frac{1}{\mu_2}. \quad (9)$$

Формула (9) в силу доказанной теоремы внешне одинакова для обычной системы M/HE₂/1 и для системы M⁻/HE₂⁻/1, но для последней системы входящие в нее параметры будут измененными вследствие операции сдвига числовыми характеристиками (табл. 1) и, как следствие, параметрами распределений (1) и (2) (табл. 2).

Определение числовых характеристик законов распределения (1) и (2)

Сначала определим числовые характеристики интервала между поступлениями требований и времени обслуживания. Для этого воспользуемся преобразованием Лапласа функций плотности (1) и (2).

Значение первой производной функции $F_\lambda^*(s)$ со знаком минус в точке $s = 0$ равно

$$-\left.\frac{dF_\lambda^*(s)}{ds}\right|_{s=0} = \frac{1}{\lambda} + t_0.$$

Отсюда среднее значение интервалов между соседними требованиями будет равно

$$\bar{\tau}_\lambda = \lambda^{-1} + t_0. \quad (10)$$

Значение второй производной функции $F_\lambda^*(s)$ в точке $s = 0$ дает второй начальный момент интервала поступления

$$\bar{\tau}_\lambda^2 = t_0^2 + 2\left(\frac{t_0}{\lambda} + \frac{1}{\lambda^2}\right). \quad (11)$$

Отсюда определим коэффициент вариации интервалов поступления:

$$c_\lambda = (1 + \lambda t_0)^{-1}. \quad (12)$$

Для определения числовых характеристик распределения (2) воспользуемся преобразованием Лапласа $F_\mu^*(s)$, приведенным выше. Тогда среднее время обслуживания в системе равно

$$\bar{\tau}_\mu = q\mu_1^{-1} + (1-q)\mu_2^{-1} + t_0, \quad (13)$$

второй начальный момент времени обслуживания равен

$$\bar{\tau}_\mu^2 = t_0^2 + 2t_0\left[\frac{q}{\mu_1} + \frac{(1-q)}{\mu_2}\right] + \frac{3q}{2\mu_1^2} + \frac{3(1-q)}{2\mu_2^2}. \quad (14)$$

Отсюда квадрат коэффициента вариации времени обслуживания будет равен

$$c_{\mu}^2 = \frac{\mu_1^2 - 2q\mu_2(\mu_1 - \mu_2) + q(1 - 2q)(\mu_1 - \mu_2)^2}{2[\mu_1 - q(\mu_1 - \mu_2) + t_0\mu_1\mu_2]^2}. \quad (15)$$

Для удобства восприятия материала статьи числовые характеристики обычных и сдвинутых распределений (1) и (2) помещены в табл. 1. Эти числовые характеристики определены с использованием свойства преобразования Лапласа соответствующей функции плотности производить моменты. Заметим, что коэффициенты вариации $c_{\lambda} < 1$ и $c_{\mu} > 0$ при параметре сдвига $t_0 > 0$, в то время как для обычной системы $M/HE_2/1$ $c_{\lambda} = 1$ и $c_{\mu} > 1$. Следовательно, ввод параметра сдвига в законы распределения существенно расширяет диапазоны параметров трафика. Таким образом, очевидно, что система $M^-/HE_2^-/1$ относится к типу $G/G/1$.

Определение параметров законов распределения (1) и (2)

Рассматривая равенства (4)–(9) как форму записи метода моментов, найдем неизвестные параметры распределений (1) и (2) λ , μ_1 , μ_2 , q , t_0 через найденные выше числовые характеристики.

Из соотношения (10) получим значение $t_0 = \bar{\tau}_{\lambda} - \lambda^{-1}$ и, подставив его в (12), найдем параметр распределения (1) $\lambda = 1/c_{\lambda}\bar{\tau}_{\lambda}$. Тогда параметр сдвига будет связан с параметрами поступлений условием

$$t_0 = \bar{\tau}_{\lambda}(1 - c_{\lambda}). \quad (16)$$

Выражение (16) будет определять диапазон изменения параметра сдвига $t_0 \in (0, 1)$.

Нахождение параметров распределения (2) μ_1 , μ_2 , q будет аналогичным нахождению этих параметров для распределения без учета сдвига. Теперь исходя из вида уравнений (13) положим

$$\mu_1 = 2q/(\bar{\tau}_{\mu} - t_0), \mu_2 = 2(1 - q)/(\bar{\tau}_{\mu} - t_0), \quad (17)$$

и потребуем выполнения условия (15). Подставив решения (17) в выражение (15), решаем полученное уравнение четвертой степени относительно параметра q с учетом условия $0 < q < 1$, т. е. не принимая во внимание тривиальные решения $q = 0$ и $q = 1$, и получим

$$q = \frac{1}{2} + \sqrt{\frac{1}{4} - \frac{3(\bar{\tau}_{\mu} - t_0)^2}{8[(\bar{\tau}_{\mu} - t_0)^2 + c_{\mu}^2\bar{\tau}_{\mu}^2]}}, \quad (18)$$

а затем определяем из выражений (17) подстановкой в них (18) параметры μ_1 , μ_2 .

Задавая значения числовых характеристик распределений (1) и (2) $\bar{\tau}_{\lambda}$, $\bar{\tau}_{\mu}$, c_{λ} , c_{μ} в качестве входных параметров системы, таким образом определяем известным методом моментов все неизвестные параметры распределений (1) и (2). Найденные таким образом параметры помещены в табл. 2.

Исследование влияния параметра сдвига на числовые характеристики законов распределений

Теперь рассмотрим влияние параметра сдвига на коэффициенты вариаций распределений. Для экспоненциального закона распределения интервалов между поступлениями требований параметр сдвига уменьшает коэффициент вариации c_{λ} в $(1 + \lambda t_0)$ раз.

Для обычного распределения HE_2 времени обслуживания квадрат коэффициента вариации, как следует из (15) при $t_0 = 0$, равен

$$c_{\mu}^2 = \frac{\mu_1^2 - 2q\mu_2(\mu_1 - \mu_2) + q(1 - 2q)(\mu_1 - \mu_2)^2}{2[\mu_1 - q(\mu_1 - \mu_2)]^2}.$$

Сравнивая последнее выражение с (15), убеждаемся, что параметр сдвига во времени $t_0 > 0$ уменьшает коэффициент вариации времени обслуживания в $1 + \frac{t_0\mu_1\mu_2}{[\mu_1(1 - q) + \mu_2q]}$ раз. Теперь, зная из теории массового обслуживания квадратичную зависимость средней задержки от коэффициентов вариаций интервалов поступлений и времени обслуживания, убеждаемся в том, что введение параметра сдвига в законы распределения уменьшает среднюю задержку требований в очереди в СМО.

Тогда алгоритм расчета среднего времени ожидания требований в очереди в системе $M^-/HE_2^-/1$ по формуле (9) в математическом пакете MathCAD сводится к следующим действиям. Задаем исходные данные $\bar{\tau}_{\lambda}$, $\bar{\tau}_{\mu}$, c_{λ} , c_{μ}

и значение параметра сдвига $t_0 \in (0, 1)$. При этом коэффициент загрузки системы определяется отношением средних значений временных интервалов $\rho = \bar{\tau}_{\mu}/\bar{\tau}_{\lambda}$. Через исходные данные определяем параметры распределений

Результаты экспериментов для СМО $M^-/HE_2^-/1$ и $M/HE_2/1$

Входные параметры		Средняя задержка							
ρ	c_λ	Для системы $M^-/HE_2^-/1$				Для системы $M/HE_2/1$			
		$c_\mu = 0,37$	$c_\mu = 1,05$	$c_\mu = 2,11$	$c_\mu = 4,21$	$c_\mu = 0,71$	$c_\mu = 2$	$c_\mu = 4$	$c_\mu = 8$
0,1	0,91	0,01	0,06	0,25	0,99	0,09	0,28	0,94	3,61
0,5	0,55	0,07	0,56	2,23	8,87	0,75	2,50	8,50	32,50
0,9	0,19	0,66	5,01	20,08	79,80	6,77	22,50	76,50	292,50

(1) и (2) λ , μ_1 , μ_2 , q по формулам табл. 2, а с использованием последних формируем коэффициенты уравнения (6) и находим его отрицательные корни $-\sigma_1$, $-\sigma_2$, $-\sigma_3$, $-\sigma_4$.

В табл. 3 приведены расчеты для системы $M^-/HE_2^-/1$ для случаев малой, средней и высокой нагрузок $\rho = 0,1$; $0,5$; $0,9$ при значении параметра сдвига $t_0 = 0,9$ и коэффициентах вариации времени обслуживания $c_\mu = 0,71$; 2 ; 4 ; 8 для системы $M/HE_2/1$. Этим значениям c_μ в системе $M^-/HE_2^-/1$ соответствуют приближенно значения $c_\mu = 0,37$; $1,05$; $2,11$; $4,21$ в связи с уменьшением коэффициента вариации согласно выражению (15) в результате введения параметра сдвига $t_0 > 0$.

Коэффициент вариации интервала между требованиями входящего потока c_λ при $t_0 = 0,9$ в зависимости от загрузки ρ принимает значения $c_\lambda = 0,91$; $0,55$; $0,19$. Таким образом, табл. 3 демонстрирует качественное и количественное влияние параметра сдвига $t_0 > 0$ на числовые характеристики распределений (1) и (2), а также на основную характеристику системы — среднюю задержку. Как и следовало ожидать, уменьшение коэффициентов вариации c_λ и c_μ влечет за собой уменьшение средней задержки в несколько раз. Коэффициент загрузки ρ в табл. 3 определяется отношением средних интервалов $\rho = \bar{\tau}_\mu / \bar{\tau}_\lambda$. Расчеты, приведенные в табл. 3, проведены для нормированного времени обслуживания $\bar{\tau}_\mu = 1$.

Выводы

В статье представлена математическая модель системы с запаздыванием во времени $M^-/HE_2^-/1$, полученная через спектральные разложения решения интегрального уравнения Линдли для систем $M/HE_2/1$ и $M^-/HE_2^-/1$. С помощью этих разложений выведена расчетная формула для средней задержки требований в очереди для указанных СМО в замкну-

той форме. Алгоритм расчета реализован в математическом пакете MathCAD. Результаты расчетов сравниваются для этих двух систем. Расчеты однозначно демонстрируют работоспособность представленной математической модели. Ее адекватность обеспечивается корректным использованием аппарата теории вероятностей и, в частности, метода спектрального разложения.

Результаты расчетов подтверждают тот факт, что диапазон изменения параметров у системы $M^-/HE_2^-/1$ со сдвинутыми входными распределениями гораздо шире, чем у обычной системы $M/HE_2/1$, и поэтому эти системы взаимно дополняют друг друга. Вместе с тем, система $M^-/HE_2^-/1$ с запаздыванием обеспечивает меньшую задержку в очереди по сравнению с обычной системой. Указанные системы с успехом могут быть применены в современной теории телетрафика. На взгляд авторов, подобные системы могут быть востребованы при анализе спутниковой передачи данных, включая дистанционное зондирование, в которых явно необходимо учитывать задержки передачи данных во времени.

Список литературы

1. Клейнрок Л. Теория массового обслуживания. Пер. с англ. под редакцией В. И. Неймана. М.: Машиностроение, 1979. 432 с.
2. Brannstrom N. A Queueing Theory analysis of wireless radio systems. Applied to HS-DSCH. Lulea university of technology. 2004. 79 p.
3. Whitt W. Approximating a point process by a renewal process: two basic methods // Operation Research. 1982. № 1. P. 125—147.
4. Myskja A. An improved heuristic approximation for the GI/GI/1 queue with bursty arrivals Teletraffic and datatraffic in a Period of Change // ITC-13. Elsevier Science Publishers. 1991. P. 683—688.
5. Бочаров П. П., Печинкин А. В. Теория массового обслуживания. М.: Изд-во РУДН, 1995. 529 с.
6. Тарасов В. Н., Бахарева Н. Ф., Блатов И. А. Анализ и расчет системы массового обслуживания с запаздыванием // Автоматика и телемеханика. 2015. № 11. С. 51—59.

7. **Тарасов В. Н.** Расширение класса систем массового обслуживания с запаздыванием // Автоматика и телемеханика. 2018. № 12. С. 57–70.
8. **Тарасов В. Н., Ахметшина Э. Г.** Среднее время ожидания в системе массового обслуживания H2/H2/1 с запаздыванием // Вестн. Сам. гос. техн. ун-та. Сер. Физ.-мат. Науки. 2018. № 4. С. 702–713.
9. **Тарасов В. Н., Бахарева Н. Ф., Ахметшина Э. Г.** Модели телетрафика на основе двойственных систем с запаздыванием с гиперэкспоненциальными и экспоненциальными распределениями // Информационные технологии. Т. 26, № 4. 2020. С. 195–202.
10. **Тарасов В. Н.** Анализ и сравнение двух систем массового обслуживания с гиперэрланговскими входными распределениями // Радиоэлектроника, информатика, управление. 2018. № 4 (47). С. 61–70.
11. **Тарасов В. Н., Липилина Л. В., Бахарева Н. Ф.** Автоматизация расчета характеристик систем массового обслуживания для широкого диапазона изменения их параметров // Информационные технологии. 2016. Т. 22, № 12. С. 952–957.
12. **Алиев Т. И.** Основы моделирования дискретных систем. СПб.: СПбГУ ИТМО, 2009. 363 с.
13. **Алиев Т. И.** Аппроксимация вероятностных распределений в моделях массового обслуживания // Научно-технический вестник информационных технологий, механики и оптики. 2013. № 2(84). С. 88–93.
14. **Novitzky S., Pender J., Rand R. H., Wesson E.** Limiting the oscillations in queues with delayed information through a novel type of delay announcement // Queueing Systems. 2020. Vol. 95. P. 281–330. DOI: <https://doi.org/10.1007/s11134-020-09657-9>.
15. **Novitzky S., Pender J., Rand R. H., Wesson E.** Nonlinear Dynamics in Queueing Theory: Determining the Size of Oscillations in Queues with Delay // SIAM J. Appl. Dyn. Syst., 2019. Vol. 18, N. 1. P. 279–311.
16. **Aras A. K., Chen X., Liu Y.** Many-server Gaussian limits for overloaded non-Markovian queues with customer abandonment // Queueing Systems. 2018. Vol. 89, N. 1. P. 81–125. DOI: <https://doi.org/10.1007/s11134-018-9575-0>.
17. **Jennings O. B., Pender J.** Comparisons of ticket and standard queues // Queueing Systems. 2016. Vol. 84, N. 1. P. 145–202. DOI: <https://doi.org/10.1007/s11134-016-9493-y>.
18. **Gromoll H., Terwilliger B., Zwart C.** Heavy traffic limit for a tandem queue with identical service times // Queueing Systems. 2018. Vol. 89, N. 3. P. 213–241. DOI: <https://doi.org/10.1007/s11134-017-9560-z>.
19. **Legros B.** M/G/1 queue with event-dependent arrival rates // Queueing Systems. 2018. Vol. 89, N. 3. P. 269–301. DOI: <https://doi.org/10.1007/s11134-017-9557-7>.
20. **Bazhba M., Blanchet J., Rhee C. H.** Queue with heavy-tailed Weibull service times // Queueing Systems. 2019. Vol. 93, N. 11. P. 1–32. DOI: <https://doi.org/10.1007/s11134-019-09640-z>.
21. **Adan I., D'Auria B., Kella O.** Special volume on 'Recent Developments in Queueing Theory' of the third ECQT conference // Queueing Systems. 2019. Vol. 93, N. 1. P. 1–190. DOI: <https://doi.org/10.1007/s11134-019-09630-1>.
22. **Adan I., D'Auria B., Kella O.** Special volume on 'Recent Developments in Queueing Theory' of the third ECQT conference: part 2 // Queueing Systems. 2019. P. 1–2. DOI: <https://doi.org/10.1007/s11134-019-09637-8>.
23. **Tibi D.** Martingales and buffer overflow for the symmetric shortest queue model // Queueing Systems. 2019. Vol. 93. P. 153–190. DOI: <https://doi.org/10.1007/s11134-019-09628-9>.
24. **Jacobovic R., Kella O.** Asymptotic independence of regenerative processes with a special dependence structure // Queueing Systems. 2019. Vol. 93. P. 139–152. DOI: <https://doi.org/10.1007/s11134-019-09606-1>.
25. **Wang L., Kulkarni V.** Fluid and diffusion models for a system of taxis and customers with delayed matching // Queueing Systems. 2020. Vol. 96. P. 101–131. DOI: <https://doi.org/10.1007/s11134-020-09659-7>.

V. N. Tarasov, Dr. of Tech. Sc., Professor, Head of Department of POUTS, e-mail: veniamin_tarasov@mail.ru,
N. F. Bakhareva, Dr. of Tech. Sc., Professor, Head of Department of ICT, e-mail: nadin1956_04@inbox.ru,
 Volga State University of Telecommunications and Informatics, Samara, 443010, Russian Federation

Mathematical Model of the Delay in Communication Networks Based on QS with a Time Lag

In the mathematical modeling of modern computer networks, telecommunication networks, traffic flows, logistics and many others, the methods of queueing theory are widely used. In turn, in studies of queueing systems (QS) G/G/1 with arbitrary distribution laws of intervals between adjacent requirements of the incoming flow and their service time, the spectral decomposition method (MSD) of solving the Lindley integral equation is often used. This method is based on the search for zeros and poles of the constructed spectral decomposition in the form of some fractional-rational function using numerical methods to determine the roots of polynomials. In this case, the coefficients of the polynomial in the numerator of the expansion are expressed through the unknown parameters of the distribution laws used to describe the QS. In the case of teletraffic research, usually these unknown parameters of the distribution laws can be determined through the numerical characteristics of the intervals between traffic packets by the method of moments. The purpose of this article is to present a fundamentally new mathematical model of a system formed by two flows with distribution laws shifted to the right. This is possible only for those probability distribution laws whose density functions are Laplace transformable. The main advantages of such systems, let us call them time lag systems, are that they provide less queue latency compared to conventional systems, and that they extend the range of traffic parameters. The article presents the results obtained on the average delay of requests in the queue for a system with exponential and hyper-Erlang distributions, an algorithm for calculating the average delay and the results of computational experiments in the Mathcad package.

Keywords: queueing system, average delay, Lindley integral equation, Laplace transform, spectral decomposition method

DOI: 10.17587/it.27.634-641

References

1. Kleinrock L. Queueing theory, Moscow, Mashinostroenie Publ., 1979, 432 p. (in Russian).
2. Brannstrom N. A. Queueing Theory analysis of wireless radio systems, *Applied to HS-DSCH*. Lulea university of technology, 2004, 79 p.
3. Whitt W. Approximating a point process by a renewal process: two basic methods, *Operation Research*, 1982, vol. 30, no. 1, pp. 125–147.
4. Myskja A. An improved heuristic approximation for the GI/GI/1 queue with bursty arrivals. Teletraffic and datatraffic in a Period of Change, *ITC-13, Elsevier Science Publishers*, 1991, pp. 683–688.
5. Bocharov P. P., Pechinkin A. V. Queueing Theory, Moscow, Publishing House of Peoples' Friendship University, 1995, 529 p. (in Russian).
6. Tarasov V. N., Bakhareva N. F., Blatov I. A. Analysis and calculation of queueing system with delay, *Automation and Remote Control*, 2015, vol. 52, no. 11, pp. 1945–1951. DOI: 10.1134/S0005117915110041.
7. Tarasov V. N. Extension of the Class of Queueing Systems with Delay, *Automation and Remote Control*, 2018, vol. 79, no. 12, pp. 2147–2157, DOI: 10.1134/S0005117918120056.
8. Tarasov V. N., Ahmetshina E. G. *Vestn. Sam. gos. tekhn. un-ta. Ser. Fiz.-mat. Nauki*, 2018, no. 4, pp. 702–713 (in Russian).
9. Tarasov V. N., Bakhareva N. F., Ahmetshina E. G. Teletraffic models based on dual systems with delay with hyperexponential and exponential distributions, *Informacionnyeologii*, 2020, vol. 26, no. 4, pp. 195–202 (in Russian).
10. Tarasov V. N. Analysis and comparison of two queueing systems with hypererlangian input distributions, *Radio Electronics Computer Science Control*, 2018, vol. 47, no. 4, pp. 61–70, DOI: 10.15588/1607-3274-2018-4-6 (in Russian).
11. Tarasov V. N., Bakhareva N. F., Lipilina L. V. *Informacionnye Technologii*, 2016, vol. 22, no. 12, pp. 952–957 (in Russian).
12. Aliev T. I. *Osnovy modelirovaniya diskretnykh sistem*, SPb., SPbGU ITMO, 2009, 363 p. (in Russian).
13. Aliev T. I. *Nauchno-tekhnicheskij vestnik informacionnykh tekhnologii, mekhaniki i optiki*, 2013, vol. 84, no. 2, pp. 88–93 (in Russian).
14. Novitzky S., Pender J., Rand R. H., Wesson E. Limiting the oscillations in queues with delayed information through a novel type of delay announcement, *Queueing Systems*, 2020, vol. 95, pp. 281–330, DOI: <https://doi.org/10.1007/s11134-020-09657-9>.
15. Novitzky S., Pender J., Rand R. H., Wesson E. Non-linear Dynamics in Queueing Theory: Determining the Size of Oscillations in Queues with Delay, *SIAM J. Appl. Dyn. Syst.*, 18-1 2019, vol. 18, no. 1, pp. 279–311, DOI: <https://doi.org/10.1137/18M1170637>.
16. Aras A. K., Chen X. & Liu Y. Many-server Gaussian limits for overloaded non-Markovian queues with customer abandonment, *Queueing Systems*, 2018, vol. 89, no. 1, pp. 81–125, DOI: <https://doi.org/10.1007/s11134-018-9575-0>.
17. Jennings O. B. & Pender J. Comparisons of ticket and standard queues, *Queueing Systems*, 2016, vol. 84, no. 1, pp. 145–202, DOI: <https://doi.org/10.1007/s11134-016-9493-y>.
18. Gromoll H. C., Terwilliger B. & Zwart B. Heavy traffic limit for a tandem queue with identical service times, *Queueing Systems*, 2018, vol. 89, no. 3, pp. 213–241, DOI: <https://doi.org/10.1007/s11134-017-9560-z>.
19. Legros B. M/G/1 queue with event-dependent arrival rates, *Queueing Systems*, 2018, vol. 89, no. 3, pp. 269–301, DOI: <https://doi.org/10.1007/s11134-017-9557-7>.
20. Bazhba M., Blanchet J., Rhee Ch. et al. Queue with heavy-tailed Weibull service times, *Queueing Systems*, 2019, vol. 93, no. 11, pp. 1–32, DOI: <https://doi.org/10.1007/s11134-019-09640-z/>.
21. Adan I., D'Auria B., Kella O. Special volume on 'Recent Developments in Queueing Theory' of the third ECQT conference, *Queueing Systems*, 2019, vol. 93, no. 1, pp. 1–190, DOI: <https://doi.org/10.1007/s11134-019-09630-1>.
22. Adan I., D'Auria B., Kella O. Special volume on 'Recent Developments in Queueing Theory' of the third ECQT conference: part 2, *Queueing Systems*, 2019, pp. 1–2, DOI: <https://doi.org/10.1007/s11134-019-09637-8>.
23. Tibi D. Martingales and buffer overflow for the symmetric shortest queue model, *Queueing Systems*, vol. 93, 2019, pp. 153–190, DOI: 10.1007/s11134-019-09628-9.
24. Jacobovic R., Kella O. Asymptotic independence of regenerative processes with a special dependence structure, *Queueing Systems*, vol. 93, 2019, pp. 139–152, DOI: 10.1007/s11134-019-09606-1.
25. Wang L., Kulkarni V. Fluid and diffusion models for a system of taxis and customers with delayed matching, *Queueing Systems*, vol. 96, 2020, pp. 101–131, DOI: 10.1007/s11134-020-09659-7.

Ю. А. Мезенцев, д-р техн. наук, проф., e-mail: mesyan@yandex.ru,

Ю. Л. Короткова, аспирант, e-mail: juliapetrulina@yandex.ru,

И. В. Эстрайх, ст. преподаватель, e-mail: ive7@yandex.ru,

Новосибирский государственный технический университет

Эффективный алгоритм решения прикладной задачи оптимизации расписаний параллельно-последовательной системы*

Рассматривается задача оптимального регулирования расписаний флота авиакомпании посредством переназначений воздушных судов на рейсы. Оптимальное регулирование расписаний заключается в таком их составлении или изменении, которое минимизирует потери системы из-за текущих нарушений. В качестве оценки потерь использовано суммарное отклонение корректируемого расписания от заданных графиков вылета воздушных судов. Показана принадлежность описываемой технологической системы к разряду параллельно-последовательных систем, задача управления которой является NP-трудной и не имеет эффективных алгоритмов точного решения. Дан краткий обзор подходов к решению ее модификаций и смежных задач управления. Приведена оригинальная формальная постановка, обоснована декомпозиция задачи и представлен алгоритм ее приближенного решения. Приведен иллюстративный пример и отражена статистика тестирования программных реализаций алгоритма декомпозиции и решения задачи регулирования расписания, доказывающие фактическую эффективность разработанного инструментария.

Ключевые слова: оптимизация расписаний, критерий минимальных отклонений, декомпозиция, эвристический алгоритм, поэтапное формирование расписаний

Введение

Проблемы эффективного управления сложными технологическими и организационными системами нередко порождают труднорешаемые задачи дискретной оптимизации. Одной из традиционных областей, в которой возникают такие задачи, опосредующих применение методов дискретной оптимизации, является управление движением авиапарков. Поскольку расписание — это основа производственного процесса любой авиакомпании, задачи, связанные с планированием и управлением расписаниями, являются наиболее актуальными и при этом наиболее сложными ввиду высокой размерности и необходимости учета целого ряда факторов.

В силу того, что процессы деятельности авиакомпании подвержены изменениям, вызываемым флуктуациями как внешней, так и внутренней среды, плановое расписание нуждается в постоянном управлении и контроле. Оперативное управление расписаниями авиакомпании предполагает внесение корректировок в текущее расписание на глубину от нескольких часов до нескольких суток исходя из совокупности ожидаемых или реализовавшихся событий и заключается в принятии решения о том, какой борт должен быть назначен на каждый рейс [1]. Такие задачи относятся к классу NP-трудных и, учитывая потребность авиакомпаний в получении моментального решения, требуют применения специальных алгоритмов быстрого приближенного решения. Это противоречие между потребностями и возможностью их удовлетворения порождает перманентную актуальность исследований

*Работа поддержана Российским фондом фундаментальных исследований, исследовательский проект № 19-37-90012\19, и финансовой поддержке Министерства науки и высшего образования РФ в рамках Госзадания, проект FSUN-2020-0009.

темы на протяжении длительного времени [2–22].

В работах [2, 3] рассмотрены подходы к оптимизации назначений воздушных судов (ВС) по рейсам авиакомпании, предложены оригинальные постановки задачи оптимизации расписания по критерию минимизации задержек. Их непосредственным развитием является материал данной статьи.

Справку по ретроспективе исследований в данном направлении также можно получить в работах [3, 4]. Приводим краткое хронологическое перечисление работ по тематике. Проект системы поддержки принятия решений (СППР) авиадиспетчеров United Airlines, представленный в работе [5], основан на сетевых моделях (1993). Эти же идеи получили развитие в СППР, способной работать в режиме реального времени [6] (1997). В работе [7] применен эвристический алгоритм принятия решений о замене типа ВС (1996). В статье [8] осуществлено формирование графика рейсов после нарушения расписания при использовании симплекс-метода и лагранжевой релаксации (1996). Работа [9] посвящена отработке нарушений планового расписания при выполнении многоплечевых рейсов для парка с несколькими типами ВС (1996). Проблема регулярности в виде задачи целочисленного программирования с ЛП-релаксацией рассмотрена в статье [10] (1997). Жадный алгоритм решения задачи в близкой постановке применен в работе [11] (1998). В работах [12, 13] в тех же целях применялись метод ветвей и границ и ЛП-релаксация в сочетании с эвристиками (2000). В работе [14] предложен эвристический алгоритм синтеза графиков расписаний (2003). В статье [15] применен метод генерации столбцов, СППР при этом обладала достаточным быстродействием для работы в реальном времени. В работе [16] (2009) применены динамическое программирование и метод генерации столбцов для решения задачи синтеза графиков движения ВС с учетом планов по техническому обслуживанию и планируемых стыковок рейсов по критерию минимизации расходов. В статье [17] (2010) представлена задача смешанно-целочисленного программирования для одновременного разрешения вопросов восстановления расписания и пассажирских стыковок после сбойной ситуации. Решить задачу удалось лишь на небольшом тестовом наборе данных, который содержал данные по 13 самолетам двух типов. В работе [18] (2012) разработан эвристический алгоритм

управления расписаниями в сбойной ситуации, сочетающей перераспределение и ВС, и пассажиров с целью минимизировать эксплуатационные расходы и влияние на пассажиров. Подход к решению был улучшен в более поздних работах [19, 20] (2014, 2016). В работе [21] (2011) сформулирована задача целочисленного программирования с минимизацией задержек и расходов, связанных с отменами рейсов и обслуживанием пассажиров в сбойных ситуациях. Решение основано на ЛП-релаксации с последующим применением эвристического алгоритма. Алгоритм был протестирован на данных, которые включали 16 самолетов и 70 рейсов. В 2019 г. была опубликована статья [22] с подходом к решению комплексной задачи восстановления расписания и пассажирских маршрутов (при условии, что пассажир сможет сам выбрать вернуть билет или изменить маршрут). В качестве целевой функции предложена минимизация общих расходов, связанных с переназначением ВС по рейсам, и затрат, связанных с изменением маршрутов по пассажирам [22].

Анализ существующих подходов к решению рассматриваемой задачи приводит к выводам о том, что ни один из них не гарантирует получения оптимального решения даже при обособленном рассмотрении задачи переназначения ВС, а также об отсутствии алгоритма нахождения оптимальных назначений для каждого ВС с доказанной эффективностью и должным быстродействием. Все это подтверждает актуальность настоящего исследования.

1. Постановка задачи оптимального регулирования расписания

Потребность в регулировании расписаний возникает из-за появления непредвиденных задержек вылета рейсов. Прикладная задача оптимального оперативного регулирования текущих графиков вылета ВС повседневно актуальна для любой авиакомпании и требует многократного решения в течение суток в реальном масштабе времени. В отличие от общей задачи составления расписаний вылетов на относительно длительный период задача оптимального регулирования действующего расписания, с одной стороны, имеет меньшую размерность, но с другой стороны, для регулирования расписания устанавливаются значительно более жесткие временные рамки. Ее содержательную постановку можно описать следующим образом.

Рассмотрим множество запланированных рейсов авиакомпании, которые необходимо переназначить множеству ВС разных типов при известных (различных) длительностях всех операций (включая обслуживание в аэропортах и перелеты) таким образом, чтобы минимизировать суммарное отклонение времени выполнения всех рейсов от планового расписания.

На момент корректировки расписания известны времена выполнения и текущие задержки начала выполнения рейсов. Такие задержки различны для разных ВС и разных рейсов. Содержательно задачи составления расписаний вылетов на сезон и задача оптимального оперативного регулирования действующего расписания мало отличаются одна от другой. Отличия касаются в большей степени средств решения для обеспечения должной точности и быстродействия. Поэтому применяемая нами формальная постановка задачи и алгоритм решения, по сути, являются универсальными и для случая составления сезонного расписания, и для случая оперативного регулирования действующего расписания.

Формальная постановка задачи оптимального регулирования расписаний флота авиакомпании основана на работе [3]. Там же приведено обоснование и оценка трудоемкости алгоритмов решения подобной задачи. Нижеследующий материал развивает идеи, изложенные в упомянутой работе. Основанием для этого является модификация релаксированной постановки задачи с априорным заданием последовательностей обслуживания рейсов ВС [3]. Запишем эту модификацию в форме задачи смешанного целочисленного программирования (milp), снабдив рядом дополнений. Введем следующие обозначения:

- $t_{i,j}$ — заданное время обслуживания рейса i ВС j ; $T = \|t_{i,j}\|$, $j = \overline{1, J}$, $i = \overline{1, I}$. Кроме этого, пусть b_j и $\overline{b_j}$ — соответственно минимальное и максимальное числа рейсов, назначаемых ВС j . Здесь и далее $\|\cdot\|$ обозначает вектор, матрицу или тензор соответствующей контексту размерности;
- S_i — исходное расписание вылета рейса $i = \overline{1, I}$, пересчитанное в виде лага в минутах от начала суток (00:00 часов);
- $\tau_{i,j}^0$ — задержка начала выполнения i -го рейса воздушным судном j . Если упорядочить рейсы для каждого ВС авиакомпании по возрастанию $\tau_{i,j}^0$, то $T^0 = \|\tau_{i,j}^0\|$, $j = \overline{1, J}$, $i = \overline{1, I}$ (I — общее число рейсов, J — число ВС), можно интерпретировать как расписа-

ние на входе каждого из ВС. Более точно $\tau_{i,j}^0$ — это рассчитанные задержки вылета всех ВС на момент корректировки расписания от 00:00 часов с учетом расписания S_j , $i \in I_j$, и текущих задержек рейсов на всех шагах приведенного ниже алгоритма A_j ;

- $x_{i,j}$ — булевы переменные-назначения ВС j на рейс i , подлежащие определению;
- непрерывные переменные $C_{i,j} \geq \tau_{i,j}^0$ — время вылета рейса i при назначении на ВС j , C_i — время вылета рейса i , $i = \overline{1, I}$, независимо от назначенного на него борта (очевидно, поскольку на рейс назначается единственный борт, всегда выполняется соотношение $C_i = \sum_{j=1}^J C_{i,j}$). Тогда формулируемую задачу можно представить в следующем виде. Найти $x_{i,j}$, C_i , $C_{i,j}$, Δ при условиях:

$$\sum_{j=1}^J x_{i,j} = 1, i = \overline{1, I}; \quad (1.1)$$

$$\underline{b_j} \leq \sum_{i=1}^I x_{i,j} \leq \overline{b_j}, j = \overline{1, J}; \quad (1.2)$$

$$x_{i,j} = \begin{cases} 1, & \text{если рейс } i \text{ назначен ВС } j, \\ 0 & \text{в противном случае,} \end{cases} \quad i = \overline{1, I}, j = \overline{1, J}; \quad (1.3)$$

$$C_{i,j} \geq S_i, i = \overline{1, I}, j = \overline{1, J}; \quad (1.4)$$

$$C_{i,j} \geq \tau_{i,j}^0 x_{i,j}, i = \overline{1, I}, j = \overline{1, J}; \quad (1.5)$$

$$C_{i,j} + t_{i,j} x_{i,j} \leq C_{k,j}, i = \overline{1, I}, j = \overline{1, J}; \quad (1.6)$$

$$\sum_{j=1}^J \tau_{i,j}^0 x_{i,j} \leq C_i, i = \overline{1, I}; \quad (1.7)$$

$$\Delta = \sum_{i=1}^I \left(C_i - S_i \sum_{j=1}^J x_{i,j} \right) \rightarrow \min. \quad (1.8)$$

Анализ результатов применения формализации (1.1)–(1.8) на реальных данных выявил существенный недостаток постановки, неочевидный для сгенерированных исходных данных, на которых задача и алгоритмы ее решения тестировались. Значения начальных задержек $\tau_{i,j}^0$, считавшиеся постоянными для любых назначений [3], таковыми не оказались. Ситуация возникает из-за возможного наличия в расписании логически связанных последовательных цепочек рейсов. Причинно-следственные связи в таких цепочках опосредуют рекурсивную зависимость задержек $\tau_{i,j}^0$

от предшествующих назначений. Поэтому система "ВС—рейсы" не относится к разряду параллельных систем. Она является параллельно-последовательной, и для решения задач управления ею естественно применение соответствующих подходов.

Прямое раскрытие рекурсий, как правило, приводит к значительному росту (иногда на много порядков) размерностей за счет появления дополнительных логических переменных и ограничений (см., например, [2, 4]), что недопустимо для задачи оперативного управления с учетом принадлежности к классу NP. Поэтому для решения проблемы была применена неполная декомпозиция с релаксацией и быстрым эвристическим многошаговым алгоритмом. Изложим его суть.

2. Декомпозиция задачи и алгоритм пересчета расписания

Сформулируем подзадачу назначения единственного рейса на каждое из P_v воздушных судов на шаге v . Исходно имеем J — число ВС; j — номер ВС, $j = \overline{1, J}$; I — число рейсов; i — номер рейса, $i = \overline{1, I}$; S_i — исходное расписание вылета рейса $i = \overline{1, I}$; положим $I \geq J$; I_v — подмножество рейсов, переназначаемых на шаге v , $v = \overline{1, N}$; N — число подзадач и шагов алгоритма. Тогда номера рейсов на шаге v принадлежат I_v ($i \in I_v$), $\bigcap_{v=1}^N I_v = \emptyset$, это условие фактически означает, что назначения ВС на рейсы на шаге v не могут быть переназначены на последующих шагах. Обозначим Δ_v — суммарное отклонение расписаний на шаге v от исходного; C_i — пересчитанное время вылета рейса i на шаге v ($i \in I_v$).

Как и ранее, определим переменные назначений ВС на рейсы:

$$x_{i,j} = \begin{cases} 1, & \text{если рейс } i \\ & \text{назначен ВС } j \text{ на шаге } v, \\ 0 & \text{в противном случае,} \end{cases} \\ j = \overline{1, J}, i \in I_v.$$

С учетом абсолютной унимодулярности нижеприводимых условий булевы переменные можно заменить непрерывными с интервальными ограничениями: $0 \leq x_{i,j} \leq 1$. $j = \overline{1, J}$, $i \in I_v$. Тогда подзадача назначения единственного рейса на каждое из P_v воздушных судов на шаге v , $P_v \leq [J/2]$, $v = \overline{1, N}$, принимает вид:

найти $x_{i,j}$, C_i , Δ_v при условиях

$$0 \leq x_{i,j} \leq 1, j = \overline{1, J}, i \in I_v; \quad (2.1)$$

$$\sum_{j=1}^J x_{i,j} \leq 1, i \in I_v; \quad (2.2)$$

$$\sum_{i \in I_v} x_{i,j} \leq 1, j = \overline{1, J}; \quad (2.3)$$

$$\sum_{i \in I_v} \sum_{j=1}^J x_{i,j} \geq P_v, P_v \leq [J/2]; \quad (2.4)$$

$$\sum_{j=1}^J \tau_{i,j}^0 x_{i,j} \leq C_i, i \in I_v; \quad (2.5)$$

$$\Delta_v = \sum_{i \in I_v} \left(C_i - S_i \sum_{j=1}^J x_{i,j} \right) \rightarrow \min, \quad (2.6)$$

где $[\cdot]$ — целая часть числа, $\sum_{v=1}^N P_v = I$.

После любого шага v задержки $\tau_{i,j}^0$, $i \in I_v$, $j = \overline{1, J}$, должны быть пересчитаны.

Координирующая задача:

$$\sum_{v=1}^N P_v = I; \quad (2.7)$$

$$\Delta = \sum_{v=1}^N \Delta_v \rightarrow \min. \quad (2.8)$$

Совершенно очевидна полиномиальная разрешимость обеих задач ввиду отсутствия цикличности обращений к подзадачам (2.1)–(2.6) класса ЛП при решении координирующей задачи (2.7)–(2.8). Соответственно, пошаговый алгоритм приближенного решения задачи оптимального регулирования расписания вылетов ВС имеет полиномиальную трудоемкость. Ниже приведено его описание.

Декомпозиционный алгоритм оперативного оптимального регулирования расписания движения воздушных судов авиакомпании

Алгоритм A_v

1. Исходные данные — множества ВС $\{1, 2, \dots, J\}$ и рейсов $\{1, 2, \dots, I\}$, известно исходное расписание вылетов рейсов S_i , $i = \overline{1, I}$, и $t_{i,j}$ — время перелета, разворота и подготовки рейса i ВС j . Разбиение множества рейсов, упорядоченных по времени вылета S_i , $i = \overline{1, I}$, на последовательность подмножеств $\{I_1, I_2, \dots, I_v, \dots, I_N\}$ априорно не определено, однако можно заранее задать числа элементов этих подмножеств посредством последовательности $\{P_1, P_2, \dots, P_v, \dots, P_N\}$. Дополнительные условия такого разбиения $P_v \leq [J/2]$, $v = \overline{1, N}$, $[\cdot]$ — целая часть числа, $\sum_{v=1}^N P_v = I$.

2. Задаем номер шага $v := 0$.
 3. Увеличиваем номер шага $v := v + 1$. Вычисляем возможные задержки рейсов для всех ВС на шаге v $\tau_{i,j}^0$, $i \in I_v$, $j = \overline{1, J}$.

4. Решаем подзадачу (2.1)–(2.6). Фиксируем оптимальные назначения $x_{i,j}^*$, $j = \overline{1, J}$, $i \in I_v$, скорректированное расписание вылетов C_i^* , $i \in I_v$, и значение критерия Δ_v^* .

5. Проверка на окончание алгоритма: при выполнении условия $v = N$ переход к следующему пункту, иначе — возврат к п. 3.

6. Объединение назначений, полученных на всех предшествующих шагах $x_{i,j}^*$, $j = \overline{1, J}$, $i \in I_v$, $v = \overline{1, N}$, расписаний вылета, C_i^* , $i \in I_v$, $v = \overline{1, N}$, и вычисление общего времени отклонения от исходного расписания $\Delta = \sum_{v=1}^N \Delta_v^*$.

Конец алгоритма.

Совершенно очевидна зависимость конечного результата решения общей задачи оптимального регулирования расписания от исходного разбиения $\{I_1, I_2, \dots, I_v, \dots, I_N\}$ и, следовательно, от набора $\{P_1, P_2, \dots, P_v, \dots, P_N\}$. Вариантов выбора числа переназначаемых рейсов по шагам $\{P_1, P_2, \dots, P_v, \dots, P_N\}$ может быть сколь угодно много. В то же время вычислительные эксперименты показывают незначительность изменений значений критерия $\Delta = \sum_{v=1}^N \Delta_v$ при варьировании значений P_v для равных N . Заметим, что минимальное значение N зависит от максимальной длины выстраиваемых цепочек рейсов и предпочтительно должно быть

четным числом. Проиллюстрируем применение формализации (2.1)–(2.8) и алгоритма A_v на небольшом примере.

3. Иллюстративный пример и результаты тестирования

Применение описанного выше инструментария проиллюстрировано на поэтапном переформировании небольшого фрагмента суточного расписания авиакомпании.

Результаты применения A_v при пересчете фрагмента расписания по шагам сведены в табл. 1–4. Рассматриваются 10 рейсов и 11 бортов. Обозначения рейсов (столбцы таблиц) и бортов (строки) соответствуют отраслевым стандартам. Для рассматриваемого примера рациональное значение N равно 2, поскольку во временной интервал фрагмента расписания укладываются цепочки рейсов длиной не более 2. Соответственно алгоритм A_v реализуется за два шага. Поэтому задаваемые на входе алгоритма наборы числа рейсов по шагам $\{4, 6\}$, $\{5, 5\}$, $\{6, 4\}$. Последние два варианта приводят к одинаковым локально наилучшим результатам. Результаты расчетов для последнего варианта числа рейсов по шагам приведены в таблицах.

Табл. 1 и 2 содержат вычисляемые пошагово значения возможных задержек вылета в минутах каждого из бортов по каждому из маршрутов $\tau_{j,i}^0$, формально являющиеся элементами

Таблица 1

Задержки $\tau_{j,i}^0$, на шаге 1 ($i \in I_1$)

Борт	Рейс									
	KZN-DME	DME-LED	LED-DME	DME-ROV	ROV-DME	DME-KZN	KZN-DME	DME-KUF	KUF-DME	DME-KZN
VP-BHF	1055	1195	1385	1090	10000	1160	1205	1125	10000	1400
VP-BHG	1160	1090	1330	995	1280	1065	1300	1030	1285	1305
VP-BHI	1150	1090	1330	985	1270	1055	1290	1020	1275	1295
VP-BHJ	1160	1100	1340	995	1280	1065	1300	1030	1285	1305
VP-BHL	1155	1095	1335	980	1265	1060	1295	1025	1280	1300
VP-BHP	1155	1095	1335	990	1275	1060	1295	1025	1280	1300
VP-BHQ	1155	1095	1335	990	1275	1060	1295	1025	1280	1300
VP-BTN	1165	1105	1345	1000	1285	1060	1295	1035	1290	1310
VP-BTP	1167	1107	1347	1002	1287	1072	1307	1037	1292	1312
VP-BTS	1150	1090	1330	985	1270	1055	1290	1020	1275	1295
VP-BTU	1157	1097	1337	992	1277	1062	1297	1017	1272	1302

Таблица 2

Задержки $\tau_{j,i}^0$, на шаге 2 ($i \in I_2$)

Борт	Рейс			
	LED-DME	ROV-DME	KZN-DME	KUF-DME
VP-BHF	1385	10000	1305	10000
VP-BHG	1330	1380	1400	1385
VP-BHI	1430	1270	1295	1275
VP-BHJ	1340	1280	1300	1285
VP-BHL	1335	1265	1295	1280
VP-BHP	1335	1275	1295	1280
VP-BHQ	1335	1275	1295	1280
VP-BTN	1345	1285	1295	1290
VP-BTP	1347	1287	1307	1292
VP-BTS	1550	1370	1290	1375
VP-BTU	1337	1277	1297	1272

транспонированной матрицы $T^0 = \|\tau_{i,j}^0\|$. Время задержки $\tau_{i,j}^0$ вычисляется от начала суток с учетом времени появления в аэропорту вылета по действующему расписанию, фактической задержки на момент пересчета расписания, времени разворота и подготовки к вылету.

Перед каждым шагом $\tau_{i,j}^0$ пересчитываются с учетом состояния системы после предыдущего шага. Фоном в табл. 1 и 2 выделены задержки, соответствующие переназначениям бортов на рейсы.

Табл. 3 аккумулирует сделанные назначения бортов на рейсы $x_{i,j}^*$ за два шага алгоритма. Назначения первого шага выделены светлым, второго — более темным фоном.

В результате сформированы четыре двухзвенные цепочки рейсов, сопряженные во времени и пространстве. Еще два рейса, совместимые для стыковки по ВС, назначены разным бортам.

Таблица 3

Результирующие назначения

Борт	Рейс									
	KZN-DME	DME-LED	LED-DME	DME-ROV	ROV-DME	DME-KZN	KZN-DME	DME-KUF	KUF-DME	DME-KZN
VP-BHF	1	0	0	0	0	0	0	0	0	0
VP-BHG	0	1	1	0	0	0	0	0	0	0
VP-BHI	0	0	0	0	0	0	0	0	0	1
VP-BHJ	0	0	0	0	0	0	0	0	0	0
VP-BHL	0	0	0	1	1	0	0	0	0	0
VP-BHP	0	0	0	0	0	0	0	0	0	0
VP-BHQ	0	0	0	0	0	0	0	0	0	0
VP-BTN	0	0	0	0	0	0	0	0	0	0
VP-BTP	0	0	0	0	0	0	0	0	0	0
VP-BTS	0	0	0	0	0	1	1	0	0	0
VP-BTU	0	0	0	0	0	0	0	1	1	0

Таблица 4

Минимальные отклонения от расписания

Отклонения	Рейс									
	KZN-DME	DME-LED	LED-DME	DME-ROV	ROV-DME	DME-KZN	KZN-DME	DME-KUF	KUF-DME	DME-KZN
S_i	1015	1050	1195	945	1110	1015	1155	980	1130	1255
$\Delta_{v,i}$	40	40	135	35	155	40	135	37	142	40
C_i	1055	1090	1330	980	1265	1055	1290	1017	1272	1295

Табл. 4 содержит значения компонент критерия качества расписания Δ .

В результате минимальное суммарное отклонение нового расписания от исходного Δ^* составило 799 мин. Наибольший ущерб выявляется на последних этапах работы алгоритма (на втором для рассмотренного примера). Фоном выделены отклонения второго этапа. Кроме этого во все значения $\tau_{i,j}^0$ в программной реализации A_v включены получасовые резервы времени.

Поэтому для рассмотренного примера фактическое суммарное отклонение скорректированного расписания от исходного составит 499 мин.

Тестирование программной реализации алгоритма проводилось на данных суточного расписания авиакомпании. В расписании 90 рейсов и 86 бортов. По оперативным данным констатируется множество существенных сбоев в реализации исходного расписания и ряд относительно некритических задержек вылетов. В частности, 26 рейсов задержаны на время от 80 до 215 мин и 27 рейсов — от 15 до 75 мин.

В результате корректировок расписания удастся снизить суммарное отклонение от исходного расписания более чем на 5 %. Тестирование также подтвердило существенную зависимость качества пересчитываемого расписания от исходного разбиения $\{P_1, P_2, \dots, P_v, \dots, P_N\}$, тем самым выявив факт зависимости эффективности применения аппарата от наличия набора рациональных разбиений и, следовательно, от опыта диспетчера. Возникший при этом вопрос выбора более глубоко не анализировался, являясь отдельной темой исследования. Время счета одного варианта корректировки расписания для любого разбиения $\{P_1, P_2, \dots, P_v, \dots, P_N\}$ и исходных данных обозначенной выше размерности составляет несколько секунд при использовании персонального компьютера (ПК).

Заключение

Показана принадлежность описываемой технологической системы числу параллельно-последовательных систем, порождающих наиболее трудоемкие задачи теории расписаний.

Осуществлена постановка NP-трудной задачи теории расписаний, проведен анализ и тестирование свойств моделей и модификаций средств поиска приближений к оптимальным решениям.

Разработан программно-математический инструментальный составления и регулирования расписаний авиакомпании по критерию минимизации суммарных отклонений от заданных графиков вылета ВС. Разработан эвристический алгоритм, основанный на декомпозиции, линейной релаксации и быстром поэтапном решении, переводящий задачу в класс полиномиально разрешимых.

Тестирование инструментария на реальных данных подтвердило эффективность примененного подхода (постановок, алгоритма и программного обеспечения) к решению рассмотренной задачи регулирования расписания по заданному критерию и времени счета. Так, для одной из действующих российских авиакомпаний применение и тестирование прототипа ПО, реализующего представленный подход, показывает в среднем более чем двукратное улучшение критерия суммарных отклонений от действующего расписания Δ (2.8) и почти трехкратное снижение соответствующего показателя пунктуальности рейсов флота авиакомпании. Сравнение показателей действующей системы оперативного управления флотом авиакомпании (на примере суточного расписания системы из 90 рейсов и 85 бортов) с результатами применения представленного выше инструментария для корректировок сбоев показывает снижение Δ с 2405 до 938 мин. Время любого пересчета расписания и переназначений рейсов с применением алгоритма A_v для этого примера не превышает трех минут работы ПК с весьма средними характеристиками памяти и быстродействия. Подобными свойствами не обладает ни одна из действующих систем оперативного регулирования расписаний.

Список литературы

1. Rushmeier R. A., Hoffman K. L., Padberg M. Recent Advances in Exact Optimization of Airline Scheduling Problems // Technical Report. 1995. DOI: 10.1.1.37.6823.
2. Mezentsev Y. A., Estraykh I. V., Chubko N. Y. Implementation of an efficient parametric algorithm for optimal scheduling on parallel machines with release dates // Journal of Physics: Conference Series № 1333. 2019. P. 1—7. DOI: 10.1088/1742-6596/1333/2/022002.
3. Мезенцев Ю. А., Короткова Ю. Л., Эстрайх И. В. Задача и инструменты оптимального регулирования расписаний флота авиакомпании // Информационные технологии. 2020. Т. 26, № 8. С. 450—459. DOI: 10.17587/it.26.450-459.
4. Avdeenko T. V., Mezentsev Y. A., Estraykh I. V. Heuristic approach to unrelated parallel machines scheduling under availability and resource constraints // IFAC-PapersOnline. 2017. Vol. 50, Iss. 1. P. 13096—13101. DOI: 10.1016/j.ifacol.2017.08.1998.
5. Jarrah A. I. Z., Yu G., Krishnamurthy N., Rakshit A. A Decision Support Framework for Airline Flight Cancellations

and Delays // Transportation Science. 1993. Vol. 27, N. 3, P. 266—280. DOI: 10.1287/trsc.27.3.266.

6. **Cao J. M., Kanafi A.** Real-time decision support for integration of airline flight cancellations and delays. Part i: Mathematical formulation // Transportation Planning and Technology. 1997. Vol. 20. P. 183—199. DOI: 10.1080/03081069708717588.

7. **Talluri K. T.** Swapping Applications in a Daily Airline Fleet Assignment // Transportation Science. 1996. Vol. 30, N. 3. P. 237—248. DOI: 10.1287/trsc.30.3.237.

8. **Yan S., Yang D-H.** A decision support framework for handling schedule perturbation // Transportation Research B. 1996. Vol. 30, N. 6. P. 405—419. DOI: 10.1016/0191-2615(96)00013-6.

9. **Yan S., Tu Y. P.** Multifleet Routing and Multistop Flight Scheduling for Schedule Perturbation // European Journal of Operational Research. 1997. Vol. 103, N. 1. P. 155—169. DOI: 10.1016/S0377-2217(96)00260-3.

10. **Lou S., Yu G.** On the Airline Schedule Perturbation Problem Caused by the Ground Delay Program // Transportation Science. 1997. Vol. 31, N. 4. P. 298—311. DOI: 10.1287/trsc.31.4.298.

11. **Arguello M. F., Bard J. F., Yu G.** A GRASP for Aircraft Routing in Response to Grounding and Delays // Journal of Combinatorial Optimization. 1997. Vol. 5. P. 211—228. DOI: 10.1023/A:1009772208981.

12. **Bard J. F., Yu G., Arguello M. F.** Optimizing Aircraft Routings in Response to Groundings and Delays // IEE Transactions. 2000. Vol. 33, P. 931—947. DOI: 10.1023/A:1010987008497.

13. **Thengvall B. G., Bard J. F., Yu G.** Balancing User Preferences for Aircraft Schedule Recovery During Irregular Operations // IEE Transactions. 2000. Vol. 32. P. 181—193. DOI: 10.1023/A:1007618928820.

14. **Rosenberger J. M., Johnson E. L., Nemhauser G. L.** Re-routing Aircraft for Airline Recovery // Transportation Science. 2003. Vol. 37, N. 4. P. 408—421. DOI: 10.1287/trsc.37.4.408.23271.

15. **Andersson T., Varbrand P.** The Flight Perturbation Problem // Transportation Planning and Technology. 2004. Vol. 27, N. 2. P. 91—118. DOI:10.1080/0308106042000218195.

16. **Eggenberg N., Salani M., Bierlaire M.** Constraint-specific recovery network for solving airline recovery problems // Computers and Operations Research. 2010. Vol. 37, N. 6. P. 1014—1026. DOI: 10.1016/j.cor.2009.08.006.

17. **Jafari N., Zegordi S. H.** Simultaneous recovery model for aircraft and passengers // Journal of the Franklin Institute. 2010. Vol. 348, N. 7. P. 20—22. DOI:10.1016/j.jfranklin.2010.03.012.

18. **Bisaillon S., Cordeau J.-F., Laporte G., Pasin F.** A large neighbourhood search heuristic for the aircraft and passenger recovery problem // 4OR. 2011. Vol. 9, N. 2. P. 139—157. DOI: 10.1007/s10288-010-0145-5.

19. **Sinclair K., Cordeau J.-F., Laporte G.** Improvements to a large neighborhood search heuristic for an integrated aircraft and passenger recovery problem // European Journal of Operational Research. 2014. Vol. 233, N. 1, P. 234—245. DOI: 10.1016/j.ejor.2013.08.034.

20. **Sinclair K., Cordeau J.-F., Laporte G.** A column generation post-optimization heuristic for the integrated aircraft and passenger recovery problem // Computers & Operations Research. 2016. Vol. 65. P. 42—52. DOI: 10.1016/j.cor.2015.06.014.

21. **Hu Y., Xu B., Gao M., Chi H.** The study on the aircraft recovery with consideration of passenger transiting // The Fifth International Conference on Management Science and Engineering Management. Macau, China: World Academic Press, World Academic Union. 2011. P. 3—7.

22. **Yang T., Hu Y.** Considering Passenger Preferences in Integrated Postdisruption Recoveries of Aircraft and Passengers // Mathematical Problems in Engineering. 2019. Vol. 9. P. 1—19. DOI:10.1155/2019/9523610.

Yu. A. Mezentsev, Dr. of Tech. Sc., Professor, e-mail: mesyan@yandex.ru,

Yu. L. Korotkova, Postgraduate Student, e-mail: juliapetrulina@yandex.ru,

I. V. Estrach, Senior Lecturer, e-mail: ive7@yandex.ru,

Novosibirsk State Technical University, Novosibirsk, Russian Federation

An Efficient Algorithm for Solving the Applied Problem Scheduling Optimization of a Parallel-Sequential System

The problem of optimal regulation of airline fleet schedules by reassigning aircraft to flights is considered. The optimal regulation of schedules is to create or change them in such a way that minimizes system losses due to current violations. As an estimate of losses, the total deviation of the adjusted schedule from the specified departure schedules of aircraft is used. It is shown that the described technological system belongs to the category of parallel-sequential systems. Accordingly, the considered system control problem is NP-hard and does not have effective algorithms for exact solution. A brief overview of approaches to solving its modifications and related fleet management tasks is given. The original formal formulation is given, the decomposition of the problem is justified, and an algorithm for its approximate solution is presented. An illustrative example is given and comparative statistics of testing software implementations of the decomposition algorithm of the schedule control problem on real data are reflected, proving the actual effectiveness of the developed tools.

Keywords: optimal schedule control, minimum deviation criterion, decomposition, heuristic algorithm, step-by-step schedule formation

Acknowledgements: The work is supported by the Russian Foundation for Basic Research, Research Project No. 19-37-90012\19; and financial support from the Ministry of Science and Higher Education of the Russian Federation under the State Task, Project FSUN-2020-0009.

DOI: 10.17587/it.27.642-650

References

1. Rushmeier R. A., Hoffman K. L., Padberg M. Recent Advances in Exact Optimization of Airline Scheduling Problems, Technical Report, 1995, DOI: 10.1.1.37.6823.
2. Mezentsev Y. A., Estraykh I. V., Chubko N. Y. Implementation of an efficient parametric algorithm for optimal scheduling on parallel machines with release dates, *Journal of Physics: Conference Series* № 1333, 2019, pp. 1–7, DOI: 10.1088/1742-6596/1333/2/022002.
3. Mezentsev Y. A., Korotkova Yu. L., Estraykh I. V. Problem and tools for optimal regulation of airline fleet schedules, *Informazionen Tekhnologii*, 2020, vol. 26, no. 8, pp. 450–459, DOI: 10.17587/it.26.450–459 (in Russian).
4. Avdeenko T. V., Mezentsev Y. A., Estraykh I. V. Heuristic approach to unrelated parallel machines scheduling under availability and resource constraints, *IFAC-PapersOnline*, 2017, vol. 50, iss. 1, pp. 13096–13101, DOI: 10.1016/j.ifacol.2017.08.1998.
5. Jarrah A. I. Z., Yu G., Krishnamurthy N., Rakshit A. A Decision Support Framework for Airline Flight Cancellations and Delays, *Transportation Science*, 1993, vol. 27, no. 3, pp. 266–280, DOI: 10.1287/trsc.27.3.266.
6. Cao J. M., Kanafi A. Real-time decision support for integration of airline flight cancellations and delays. part i: Mathematical formulation, *Transportation Planning and Technology*, 1997, vol. 20, pp. 183–199, DOI: 10.1080/03081069708717588.
7. Talluri K. T. Swapping Applications in a Daily Airline Fleet Assignment, *Transportation Science*, 1996, vol. 30, no. 3, pp. 237–248, DOI: 10.1287/trsc.30.3.237.
8. Yan S., Yang D.-H. A decision support framework for handling schedule perturbation, *Transportation Research B*, 1996, vol. 30, iss. 6, pp. 405–419, DOI: 10.1016/0191-2615(96)00013-6.
9. Yan S., Tu Y. P. Multifleet Routing and Multistop Flight Scheduling for Schedule Perturbation, *European Journal of Operational Research*, 1997, vol. 103, iss. 1, pp. 155–169, DOI: 10.1016/S0377-2217(96)00260-3.
10. Lou S., Yu G. On the Airline Schedule Perturbation Problem Caused by the Ground Delay Program, *Transportation Science*, 1997, vol. 31, no. 4, pp. 298–311, DOI: 10.1287/trsc.31.4.298.
11. Arguello M. F., Bard J. F., Yu G. A GRASP for Aircraft Routing in Response to Grounding and Delays, *Journal of Combinatorial Optimization*, 1997, vol. 5, pp. 211–228, DOI: 10.1023/A:1009772208981.
12. Bard J. F., Yu G., Arguello M. F. Optimizing Aircraft Routings in Response to Groundings and Delays, *IEEE Transactions*, 2000, vol. 33, pp. 931–947, DOI: 10.1023/A:1010987008497.
13. Thengvall B. G., Bard J. F., Yu G. Balancing User Preferences for Aircraft Schedule Recovery During Irregular Operations, *IEEE Transactions*, 2000, vol. 32, pp. 181–193, DOI: 10.1023/A:1007618928820.
14. Rosenberger J. M., Johnson E. L., Nemhauser G. L. Re-routing Aircraft for Airline Recovery, *Transportation Science*, 2003, vol. 37, no. 4, pp. 408–421, DOI: 10.1287/trsc.37.4.408.23271.
15. Andersson T., Varbrand P. The Flight Perturbation Problem, *Transportation Planning and Technology*, 2004, vol. 27, no. 2, pp. 91–118, DOI: 10.1080/0308106042000218195.
16. Eggenberg N., Salani M., Bierlaire M. Constraint-specific recovery network for solving airline recovery problems, *Computers and Operations Research*, 2010, vol. 37, iss. 6, pp. 1014–1026, DOI: 10.1016/j.cor.2009.08.006.
17. Jafari N., Zegordi S. H. Simultaneous recovery model for aircraft and passengers, *Journal of the Franklin Institute*, 2010, vol. 348, iss. 7, pp. 20–22, DOI: 10.1016/j.jfranklin.2010.03.012.
18. Bisailon S., Cordeau J.-F., Laporte G., Pasin F. A large neighbourhood search heuristic for the aircraft and passenger recovery problem, *4OR*, 2011, vol. 9, no. 2, pp. 139–157, DOI: 10.1007/s10288-010-0145-5.
19. Sinclair K., Cordeau J.-F., Laporte G. Improvements to a large neighborhood search heuristic for an integrated aircraft and passenger recovery problem, *European Journal of Operational Research*, 2014, vol. 233, no. 1, pp. 234–245, DOI: 10.1016/j.ejor.2013.08.034.
20. Sinclair K., Cordeau J.-F., Laporte G. A column generation post-optimization heuristic for the integrated aircraft and passenger recovery problem, *Computers & Operations Research*, 2016, vol. 65, pp. 42–52, DOI: 10.1016/j.cor.2015.06.014.
21. Hu Y., Xu B., Gao M., Chi H. The study on the aircraft recovery with consideration of passenger transiting, *The Fifth International Conference on Management Science and Engineering Management*, Macau, China, World Academic Press, World Academic Union 2011, pp. 3–7.
22. Yang T., Hu Y. Considering Passenger Preferences in Integrated Postdisruption Recoveries of Aircraft and Passengers, *Mathematical Problems in Engineering*, 2019, vol. 9, pp. 1–19, DOI: 10.1155/2019/9523610.

В. В. Мартынов, д-р техн. наук, проф., e-mail: vvmartynov@bk.ru,
О. В. Ширяев, канд. техн. наук, доц., e-mail: oleg_shiryaev@bk.ru,
Уфимский государственный авиационный технический университет

Обеспечение комплексной безопасности в рамках построения цифровой архитектуры предприятий*

Сегодня известны модели построения архитектуры предприятия (АП) вообще и энергетики в частности. Однако данные модели обеспечивают только системный подход к формированию АП, но не содержат метрик, а соответственно, не позволяют сравнивать различные варианты архитектур и тем более строить оптимальную АП. В наших работах предложена технология проектирования оптимальной АП. В статье представлено развитие данной технологии и интеграция подхода к решению вопросов комплексной безопасности в АП. Данная технология может быть приложена к любому использующему информационные и цифровые технологии предприятию.

Ключевые слова: архитектура предприятия, методология проектирования архитектуры предприятия, модель Захмана, комплексная безопасность, архитектура информационной безопасности предприятия

Введение

В условиях конкурентной борьбы предприятия часто принимают решения по оптимизации своей архитектуры в целях их эффективного функционирования на рынке. Изменения могут затрагивать разные компоненты архитектуры предприятия (АП) (миссию, стратегию, бизнес-процессы, организационную структуру и др.), но ключевое значение в происходящих изменениях играют существующие бизнес-процессы предприятия. Наличие оптимальной архитектуры позволяет предприятию адаптироваться к внешним воздействиям и использовать инновационные подходы, что обеспечивает гибкость, стабильность и эффективность его работы. Сейчас практически нет предприятий, которые для поддержки своих бизнес-процессов не используют информационные и цифровые технологии.

1. Анализ существующих подходов к построению АП

В настоящее время существует множество зарубежных и отечественных подходов к построению АП. Стоит отметить, что большая часть предприятий формирует свою архитектуру стихийно, в лучшем случае с использованием существующих архитектурных моделей, которые, несмотря на применение системного подхода при построении АП, не предоставляют возможности количественно оценить варианты ее построения на базе метрик описания ее компонентов [1]. Соответственно, вести речь о минимизации затрат на принятую архитектуру, обеспечивающую необходимую функциональность, не приходится. Таким образом, формирование и внедрение оптимальной АП является актуальной проблемой. Решение данной проблемы позволит активно внедрять и использовать передовые информационные технологии на основе накопленного потенциала предприятия. Каждый компонент АП должен соответствовать набору функциональных требований к нему, которые определяются бизнес-процессами и бизнес-пользователями.

*Исследование выполнено при финансовой поддержке Российского фонда фундаментальных исследований, проект № 20-08-00796.

Следовательно, разработка целевой АП и плана перехода к ней для предприятий является актуальной задачей исследования. Для решения вышеопределенной задачи требуется систематизировать структуру АП, которая должна соответствовать направлению развития передовых областей цифровых информационных технологий с учетом анализа существующей АП.

Оптимальное архитектурное решение позволяет выделить границу между теми компонентами АП, которые не должны быть изменены, и теми, оптимизация и изменение которых диктуется требованиями бизнеса. Это предъявляет высокие требования к качеству архитектуры. Качество означает, что ИТ-архитектура действительно помогает в достижении основных бизнес-целей организации. АП должна не только помогать достигать бизнес-цели предприятия, но и поддерживать их. Подобное решение требует больших затрат для предприятия, поэтому вопросы минимизации затрат на ее построение также являются актуальными. Следовательно, при построении и поддержании архитектуры в актуальном состоянии выбор должен быть связан с бизнес-целями предприятия и обеспечивать их оптимальное достижение.

Сравнительный анализ существующих методологий и моделей построения АП показал, что большинство из них опираются на личный опыт системных аналитиков и разработчиков (включая предпочтения в выборе методологий и средств моделирования), носят как правило декларативный характер и представляют собой успешный опыт реализации проектов. При этом стоит отметить, что в существующих методологиях отсутствует математическая формализация процесса построения АП [2]. Поэтому разрабатываемые решения нельзя назвать даже рациональными. При построении оптимальной АП необходимо сформировать ряд четких количественных критериев, а также метрик и показателей, которые позволят оценить варианты построения АП и найти среди них оптимальный.

В работах зарубежных и отечественных ученых Дж. А. Захмана [3], Дж. Ф. Соу [4], С. Спивака [5, 6], И. Ильина [7], Д. В. Кудрявцева [8] и др. рассмотрены различные методологии проектирования концепции АП. В научных статьях наиболее часто рассматривают-

ся следующие методологии построения АП: Захмана, TOGAF, FEAF, Gartner, Майкрософт, MDA, GRAI-GIM, SAM, E2AF и другие. В статье [9] рассмотрены существующие методики описания и построения АП.

Анализ работ и представленных в них моделей АП показал, что недостаточно внимания уделено системному решению вопросов комплексной безопасности как части оптимальной АП, хотя очевидно, что архитектура информационной безопасности предприятия (АИБП) является важной частью АП и обеспечивает санкционированное функционирование последней. Основной целью создания АИБП является обеспечение соответствия бизнес-стратегии предприятия необходимому уровню его ИТ-безопасности. При этом стоит отметить, что АИБП должна быть гибкой и адаптивной к воздействиям как со стороны внутренних (например, изменения в организационной стратегии), так и со стороны внешних (изменения в технологиях, требованиях клиентов и др.) факторов.

В работе [10] рассматриваются принципы архитектурного моделирования системы информационной безопасности. Работа [11] посвящена выяснению взаимосвязи между АП и АИБП путем рассмотрения АИБП в различных системах АП. Кроме этого, в работе [11] дается классификация и оценка различных стратегий развития АИБП, а также рассмотрены вопросы безопасности в различных методологиях построения АП. В работе [12] автор предложил подход к обеспечению информационной безопасности путем добавления в структуру методологии Захмана "информационной гарантии" во все уровни проектирования предприятия, что поможет ИТ-специалистам решать проблемы безопасности. Однако такой подход сложно реализовать при интеграции "информационной гарантии" в другие архитектуры, так как безопасность не может быть отделена от данных, процессов, людей и т. д. В работе [13] предложен подход к обеспечению информационной безопасности в методологии FEAF, который получил название "Federal Enterprise Architecture Security and Privacy Profile (SPP)". SPP рассматривается как эталонная модель безопасности для FEAF и поддерживает различные этапы архитектуры безопасности (определение и анализ требований безопасности, выбор решений безопасности), а также

позволяет интегрировать федеральные стандарты информационной безопасности с АП. Методологии DoDAF и TOGAF рассматривают безопасность как основу других продуктов АП, таких как матрица обмена информацией и диаграмма безопасности данных, которые являются компонентами архитектуры данных, позволяющими показать, к каким данным может получить доступ каждый актор [11].

В работе [14] предложена интегрированная концептуальная модель, в соответствии с которой управление АП может поддерживать управление рисками и управление информационной безопасностью.

В статье [15] отмечается отсутствие в существующих методологиях проектирования АП (TOGAF, Захмана, MoDAF, DoDAF, SABSA) целостных методов обеспечения безопасности в масштабах всего предприятия, что, по мнению авторов, может привести к тому, что предприятия будут подвержены значительным рискам безопасности, которые впоследствии приведут к финансовым потерям. Также в статье [15] определены основные требования к безопасности в системах TOGAF, SABSA и COBIT19 и сделан вывод о том, что необходимо включать точные рекомендации по кибербезопасности, согласовывающие бизнес-стратегии АП.

Анализ подходов к построению АП показал, что существующие исследования в данном направлении в основном лишь дополняют системный подход, реализованный в виде методик, моделей и фреймворков построения АП, и на данный момент нет исследований, содержащих метрики, позволяющие оценить эффективность решения вопросов информационной безопасности предприятия (ИБП) в АП. Таким образом, проблема решения вопросов ИБП как части оптимальной АП является актуальной.

2. Вопросы комплексной безопасности при построении архитектуры предприятия

Как уже отмечалось ранее в статье [9], существующие подходы и методологии проектирования АП имеют недостаточную проработку и слабую формализацию методов планирования перехода от текущего состояния АП к запланированному. Авторы указанной статьи осуществили формализацию процесса построения

АП на основе модели Захмана как содержащей основные инварианты элементов других архитектурных моделей.

Для анализа текущей АП необходимы методы комплексного описания ее слоев и отдельных элементов. В приведенной работе рассмотрено пять основных архитектурных представлений:

- 1) бизнес-процессы;
- 2) бизнес-системы;
- 3) данные;
- 4) приложения
- 5) технологии.

Для каждого архитектурного представления предложены выражения оценки его структурного объема и сложности. При нахождении оптимального архитектурного решения на каждом уровне необходимо определить ограничения ресурсов, которые доступны для построения АП, а также набор метрик, с помощью которого планируется оценивать и сравнивать проектные решения.

Эти пять архитектурных представлений учитывают достаточное разнообразие бизнес-процессов, бизнес-систем, модели данных и т. д. Однако вопросам комплексной безопасности при построении АП в данной работе внимание не уделено. В связи с этим предлагается расширить предложенный список до шести представлений, включив в него "архитектуру информационной безопасности".

Представим АП в виде кортежа:

$$EA = \{R, B, S, D, A, T, C, I, M\},$$

где $R = (r_1, r_2, \dots, r_n)$ — набор системных требований;

$B = (b_1, b_2, \dots, b_p)$ — набор бизнес-процессов;

$S = (s_1, s_2, \dots, s_q)$ — набор бизнес-систем;

$D = (d_1, d_2, \dots, d_g)$ — набор элементов данных;

$A = (a_1, a_2, \dots, a_k)$ — набор приложений;

$T = (t_1, t_2, \dots, t_w)$ — набор технологий;

$C = (c_1, c_2, \dots, c_h)$ — набор ресурсных ограничений и правил проектирования;

$I = (i_1, i_2, \dots, i_e)$ — набор вариантов обеспечения достаточной информационной безопасности предприятия;

$M = (M_B, M_S, M_D, M_A, M_T, M_I)$ — набор метрик для архитектурных представлений.

В статье [16] конкретизированы вышеприведенные формальные основы процесса построения АП на предприятия энергетического комплекса Республики Башкортостан.

Рассмотрим более подробно вопросы ИБП. Важным является обеспечение комплексного подхода при разработке системы безопасности предприятия. Существует множество методик, обеспечивающих выявление самых уязвимых мест в структуре работы с информацией. Комплекс превентивных мер против атак на информацию регламентируется политикой безопасности предприятия.

Комплексная система безопасности включает:

1) классификацию информационных объектов по различным категориям информационной безопасности (требования к доступности, целостности и конфиденциальности обрабатываемых данных);

2) политику ролей (определение для сотрудников предприятия ролей, уровня доступа к информации и ответственности за ее использование);

3) политику информационной безопасности (учет рисков информационных атак, современной ситуации, внешних факторов и стоимости проекта);

4) методы обеспечения безотказности (использование систем самотестирования и внесение избыточности на аппаратном, программном, а также информационном уровнях).

Требования по работе с конфиденциальной информацией зависят от ее типа (табл. 1).

При этом стоит отметить, что каждый последующий класс конфиденциальности наследует требования по работе с конфиденциальной информацией предыдущего класса.

Политика ролей заключается в том, что в зависимости от роли определяется доступ и возможности по работе с конфиденциальной информацией на предприятии. В табл. 2 при-

ведены типовые роли по работе с конфиденциальной информацией.

Разработка политики информационной безопасности осуществляется по нескольким направлениям: определение данных и уровня их защиты; определение источника и степени ущерба; расчет рисков и формирование рекомендаций по их уменьшению.

При проектировании оптимальной АП также необходимо выбрать комплекс методов информационной безопасности. Существуют различные методы обеспечения информационной безопасности: технические (межсетевые экраны, системы аутентификации и шифрования, антивирусные программы, разграничение прав доступа к конфиденциальной информации и др.); административные (запрет на использование сотрудниками внешних накопителей данных, ноутбуков и т. д.); правовые (ответственность за правонарушения в сфере информационной безопасности, аттестация объектов информатизации и др.); физические (охранные системы, камеры наблюдения и т. д.).

Выбор вариантов обеспечения достаточной информационной безопасности предприятия осуществляется следующим образом:

1) составляется перечень возможных атак (спам, копирование жесткого диска и т. д.);

2) для каждой угрозы определяется показатель риска;

3) устанавливается максимально допустимое значение риска;

4) определяется перечень угроз с показателем риска, превышающим максимально допустимое значение;

5) формируются рекомендации с указанием первоочередных целей разработки политики безопасности.

Таблица 1

Требования по работе с конфиденциальной информацией

Внутренняя информация (1 класс)	Конфиденциальная информация (2 класс)	Секретная информация (3 класс)
- Осведомление сотрудников о закрытости данной информации; - общее ознакомление сотрудников с основными возможными методами атак на информацию; - ограничение физического доступа к информации; - полный набор документации по правилам выполнения операций с данной информацией	- Расчет рисков атак на информацию; - поддержание списка лиц, имеющих доступ к данной информации; - по возможности выдача подобной информации под расписку; - автоматическая система проверки целостности системы и ее средств безопасности; - надежные схемы физической транспортировки; - обязательное шифрование при передаче по линиям связи; - схема бесперебойного питания ЭВМ	- Детальный план спасения либо надежного уничтожения информации в аварийных ситуациях (пожар, наводнение, взрыв); - защита ЭВМ либо носителей информации от повреждения водой и высокой температурой; - криптографическая проверка целостности информации

Типовые роли по работе с конфиденциальной информацией

Роль	Функции
Специалист по информационной безопасности	- Разработка и поддержание политики безопасности предприятия; - расчет и перерасчет рисков; - поиск информации об обнаруженных уязвимостях в программном обеспечении предприятия и в стандартных алгоритмах
Владелец информации	- Работа с конфиденциальной информацией, к которой он имеет доступ; - оценка класса обрабатываемой информации; - определение нестандартных методов атак на обрабатываемую информацию
Поставщик аппаратного и программного обеспечения	Обеспечение достаточного уровня информационной безопасности в поставляемых им продуктах
Разработчик системы и/или программного обеспечения	- Обеспечение достаточного уровня безопасности разрабатываемой системы или программного обеспечения; - взаимодействие со специалистами по информационной безопасности при разработке системы или программного обеспечения
Линейный менеджер или менеджер отдела	Инструктаж и мониторинг выполнения персоналом требований службы безопасности
Операторы	Работа с конфиденциальной информацией
Аудиторы	Проверка организации и функционирования системы безопасности предприятия

Показатель риска высчитывается по следующей формуле:

$$Risk_i = u_i v_i,$$

где u_i — ущерб от i -й угрозы; v_i — вероятность наступления i -й угрозы.

По результатам анализа рисков предприятия формируется набор вариантов обеспечения достаточной информационной безопасности предприятия (I).

В зависимости от выбранного набора вариантов обеспечения ИБП формируется набор компонентов ($I_1, I_2, \dots, I_x$) для соответствующего архитектурного представления.

По аналогии с работой [9] для оценки структурного объема и сложности архитектурного представления "Архитектура информационной безопасности" предлагается следующая метрика:

$$Q_I = \ln(I_1 I_2 \dots I_x),$$

$$M_I = 1 - e^{Q_I / R_I}.$$

В данном случае переменные $I_1, \dots, I_x$ являются равнозначными; Q_I — параметр для метрики M_I ; R_I — эталонный параметр, необходимый для "привязки" полученного метрического значения, а метрика M_I — это совокупное распределение для экспоненциальной случайной величины.

Также стоит отметить, что затраты Z на построение оптимальной АП зависят от затрат на выбранные наборы бизнес-процессов Z_B , бизнес-систем Z_S , элементов данных Z_D , приложений Z_A , технологий Z_T и вариантов обеспечения достаточной ИБП Z_I :

$$Z = Z_B + Z_S + Z_D + Z_A + Z_T + Z_I.$$

При этом затраты на построение оптимальной АП, достаточные для обеспечения необходимой ее функциональности, должны стремиться к минимуму:

$$Z \rightarrow \min.$$

Исследования показали, что затраты на построение АП необходимо минимизировать. При этом сама АП должна соответствовать потребностям предприятия при условии обеспечения достаточной информационной безопасности.

Заключение

Для поддержания ИБП на соответствующем уровне необходимо применять комбинацию различных методов информационной безопасности. Стоит отметить, что набор методов информационной безопасности должен

формироваться в зависимости от специфики предприятия и требований к защите информации с учетом последующей оптимизации для построения эффективной архитектуры.

В работе представлено расширение технологии проектирования оптимальной АП в части включения в нее подхода к решению вопросов комплексной безопасности.

Список литературы

1. **Martynov V. V., Salimova A. I.** The approach to construction information enterprise architecture // Proceedings of the Workshop on CSIT'12, Russia, Ufa-Hamburg-Norwegian Fjords, USATU. 2012. Vol. 2. P. 45–49.
2. **Мартынов В. В., Шавалеева Д. Н., Дидык Т. Г.** Переход к цифровому обществу и далее к Индустрии 5.0 на базе существующих и перспективных технологий // Инновационные, информационные и коммуникационные технологии: сборник трудов XV Международной научно-практической конференции. 2019. С. 86–91.
3. **Zachman J. A.** Framework for Information System Architecture // IBM System Journal. 1987. Vol. 26, N. 3. P. 276–292.
4. **Sowa J. F., Zachman J. A.** Extending and Formalizing the Framework for Information System Architecture // IBM Systems Journal. 1992. Vol. 31, N. 3. P. 590–616.
5. **Spewak S. H., Steven C. H.** Enterprise Architecture Planning: Developing a Blueprint for Data, Application and Technology. NY: A WileyQED Publication, 1992, 392 p.
6. **Spewak S. H., Tiemann M.** Updating the enterprise architecture planning model // Journal of Enterprise Architecture. 2006. N. 12. P. 11–19.
7. **Ильин И. В., Левина А. И.** Интеграция проектного подхода в модель бизнес-архитектуры предприятия // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Экономические науки. 2013. № 6-2(185). С. 74–82.
8. **Кудрявцев Д. В., Арзуманян М. Ю., Григорьев Л. Ю.** Технологии бизнес-инжиниринга: Учеб. пособ. Санкт-Петербург, Санкт-Петербургский политехнический университет Петра Великого, 2014. 427 с.
9. **Мартынов В. В., Салимова А. И., Шавалеева Д. Н.** Формальные основы процесса построения архитектуры предприятия на базе метрик описания ее компонентов // Вестник Рязанского государственного радиотехнического университета. 2019. № 70. С. 52–64. DOI 10.21667/1995-4565-2019-70-52-64.
10. **Ильина О. П., Сотавов А. К.** Архитектурное моделирование системы информационной безопасности // ТТПС. 2019. № 2 (48). URL: <https://cyberleninka.ru/article/n/arhitekturnoe-modelirovanie-sistemy-informatsionnoy-bezopasnosti> (дата обращения: 28.07.2021).
11. **Jalaliniya S.** Enterprise Architecture & Security Architecture Development. URL: https://www.researchgate.net/publication/257934615_Enterprise_Architecture_Security_Architecture_Development (accessed: 27.07.2021).
12. **Heaney J., Hybertson D., Reedy A., Chapin S., Bollinger T., Williams D., Kirwan M.** Information Assurance for Enterprise Engineering. URL: <https://hillside.net/plop/plop2002/final/PLoP-2002-Heaney-7-22.pdf> (accessed: 27.07.2021).
13. **NIST and OMB**, 2010, Federal Enterprise Architecture Security and Privacy Profile, Version 3.0. URL: <https://s3.amazonaws.com/sitesusa/wp-content/uploads/sites/1151/2016/10/FEA-Security-Privacy-Profile-v3-09-30-2010.pdf> (accessed: 27.07.2021).
14. **Diefenbach T., Lucke C., Lechner U.** Towards an Integration of Information Security Management, Risk Management and Enterprise Architecture Management — A Literature Review // 2019 IEEE International Conference on Cloud Computing Technology and Science (CloudCom). 2019. P. 326–333. DOI: 10.1109/CloudCom.2019.00057.
15. **Al-Turkistani H. F., Aldobaian S., Latif R.** Enterprise Architecture Frameworks Assessment: Capabilities, Cyber Security and Resiliency Review // 2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA). 2021. P. 79–84. DOI: 10.1109/CAIDA51941.2021.9425343.
16. **Martynov V., Filosoza E., Zaytseva A.** Methodology for modeling optimal enterprise architecture in the application to the strategy of socio-economic development of energy enterprises of the Republic of Bashkortostan // Materials of International scientific and practical Conference on Electrotechnical Complexes and Systems. Ufa: USATU, 2020. P. 372–376.

V. V. Martynov, Professor, e-mail: vvmartynov@bk.ru,
O. V. Shiryayev, Assistant Professor, e-mail: oleg_shiryayev@bk.ru,
Ufa State Aviation Technical University, Russian Federation

Ensuring Integrated Security as Part of Building Digital Architecture for Energy Companies

Today, there are models for building an enterprise architecture (EA) in general and power engineering in particular. However, these models provide only a systematic approach to the formation of EA, but do not contain metrics, therefore, they do not allow comparing different versions of architectures, and even more so to build an optimal EA. In our works, we have proposed a technology for designing an optimal EA. The article is the development of this technology and the integration of an approach to solving complex safety issues in an accident. This technology can be applied to any enterprise using information and digital technologies.

Keywords: enterprise architecture, enterprise architecture design methodology, Zachman model, integrated security, enterprise information security architecture

DOI: 10.17587/it.27.651-657

References

1. Martynov V. V., Salimova A. I. The approach to construction information enterprise architecture, *Proceedings of the Workshop on CSIT'12*, Russia, Ufa-Hamburg-Norwegian Fjords, USATU, 2012, vol. 2, pp. 45–49.
2. Martynov V. V., Shavaleeva D. N., Didyk T. G. Transition to a digital society and further to Industry 5.0 based on existing and promising technologies, *Collection of works of the XV International Scientific and Practical Conference Innovative, information and communication technologies*, 2019, pp. 86–91 (in Russian).
3. Zachman J. A. Framework for Information System Architecture, *IBM System Journal*, 1987, vol. 26, no. 3, pp. 276–292.
4. Sowa J. F., Zachman J. A. Extending and Formalizing the Framework for Information System Architecture, *IBM Systems Journal*, 1992, vol. 31, no. 3, pp. 590–616.
5. Spewak S. H., Steven C. H. Enterprise Architecture Planning: Developing a Blueprint for Data, Application and Technology, NY, A WileyQED Publication, 1992, 392 p.
6. Spewak S. H., Tiemann M. Updating the enterprise architecture planning model, *Journal of Enterprise Architecture*, 2006, no. 12, pp. 11–19.
7. Ilin I. V., Levina A. I. The integration of the project management approach into the business architecture model of the company, *St. Petersburg State Polytechnical University Journal. Economics*, 2013, no. 6–2 (185), pp. 74–82 (in Russian).
8. Kudriavtsev D. V., Arzumaniy M. Iu., Grigor'ev L. Iu. Business engineering technologies: Tutorial, *St. Petersburg, Peter the Great St. Petersburg Polytechnic University*, 2014, 427 p. (in Russian).
9. Martynov V. V., Salimova A. I., Shavaleeva D. N. Formal foundations of the process of building an enterprise architecture based on metrics for describing its components, *Bulletin of the Ryazan State Radio Engineering University*, 2019, no. 70, pp. 52–64, DOI: 10.21667/1995-4565-2019-70-52-64.
10. Ilin O. P., Sotavov A. K. Architectural modeling of the information security system, *TTPS*, 2019, no. 2 (48), available at: <https://cyberleninka.ru/article/n/arhitekturnoe-modelirovanie-sistemy-informatsionnoy-bezopasnosti> (accessed: 28.07.2021).
11. Jalaliniya S. Enterprise Architecture & Security Architecture Development. available at: https://www.researchgate.net/publication/257934615_Enterprise_Architecture_security_Architecture_Development (accessed: 27.07.2021).
12. Heaney J., Hybertson D., Reedy A., Chapin S., Bollinger T., Williams D., Kirwan M. Information Assurance for Enterprise Engineering, available at: <https://hillside.net/plop/plop2002/final/PLoP-2002-Heaney-7-22.pdf> (accessed: 27.07.2021).
13. NIST and OMB, 2010, Federal Enterprise Architecture Security and Privacy Profile, Version 3.0, available at: <https://s3.amazonaws.com/sitesusa/wp-content/uploads/sites/1151/2016/10/FEA-Security-Privacy-Profile-v3-09-30-2010.pdf> (accessed: 27.07.2021).
14. Diefenbach T., Lucke C., Lechner U. Towards an Integration of Information Security Management, Risk Management and Enterprise Architecture Management — A Literature Review, *2019 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, 2019, pp. 326–333, DOI: 10.1109/CloudCom.2019.00057.
15. Al-Turkistani H. F., Aldobaian S., Latif R. Enterprise Architecture Frameworks Assessment: Capabilities, Cyber Security and Resiliency Review, *2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA)*, 2021, pp. 79–84, DOI: 10.1109/CAIDA51941.2021.9425343.
16. Martynov V., Filsova E., Zaytseva A. Methodology for modeling optimal enterprise architecture in the application to the strategy of socio-economic development of energy enterprises of the Republic of Bashkortostan, *Materials of International scientific and practical Conference on Electrotechnical Complexes and Systems*, Ufa, USATU, 2020, pp. 372–376.

31 мая – 21 июня 2022 г. в Санкт-Петербурге
на базе ОАО "Концерн «ЦНИИ «Электронприбор»
состоится

XXVIII Санкт-Петербургская Международная конференция по интегрированным навигационным системам

Тематика конференции

- Инерциальные датчики, системы навигации и ориентации
- Интегрированные системы навигации и управления движением
- Глобальные навигационные спутниковые системы
- Средства гравиметрической поддержки навигации

В рамках каждого направления рассматриваются:

- схемы построения и конструктивные особенности
- методы и алгоритмы
- особенности разработки и применения для различных подвижных объектов и условий движения (аэрокосмические, морские, наземные, подземные)
- испытания и метрология.

Контактная информация:

Тел.: +7 (812) 499 82 10 +7 (812) 499 81 57
Факс: +7 (812) 232 33 76 E-mail: icins@eprib.ru

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ЭКОНОМИКЕ, УПРАВЛЕНИИ И ПРОИЗВОДСТВЕ

INFORMATION TECHNOLOGIES IN ECONOMY, MANAGEMENT AND PRODUCTION

УДК 519.876.5

DOI: 10.17587/it.27.658-667

А. В. Медведев, д-р техн. наук, проф., e-mail: mav2745@mail.ru,
Д. И. Ярещенко, ст. преподаватель, e-mail: yareshenkodi@yandex.ru,
Сибирский федеральный университет, г. Красноярск

Идентификация и управление многомерными процессами с запаздыванием в условиях неполной информации*

Рассматриваются задачи идентификации и управления многомерными дискретно-непрерывными процессами с запаздыванием в условиях неполной информации об объекте, т. е. в условиях, когда вид параметрических уравнений по различным каналам объекта отсутствует из-за недостатка априорной информации. При этом многомерные процессы имеют стохастические зависимости компонент вектора выходных переменных. В таких условиях математическое описание подобных процессов приводит к системе неявных уравнений. Предлагаются непараметрические алгоритмы идентификации и управления для многомерных систем.

Главной задачей моделирования таких процессов является определение прогнозных значений выходных переменных по известным входным, причем для неявных уравнений известно лишь то, что та или иная выходная переменная может зависеть от других входных и выходных переменных, определяющих состояние многомерной системы.

В данном исследовании возникает нетривиальная ситуация решения системы неявных уравнений в условиях, когда неизвестны зависимости между компонентами выходных переменных. Применение параметрической теории идентификации в этом случае не приведет к успеху. Одним из возможных направлений является использование теории непараметрических систем. Основным содержанием работы является решение задачи идентификации при наличии зависимостей выходных переменных и далее решение задачи управления таким процессом. Здесь следует обратить внимание на то, что при определении задающих воздействий для многомерной системы сначала необходимо решить систему задающих воздействий, так как выбирать произвольно задающие воздействия из области определения выходных переменных не представляется возможным.

Приводятся вычислительные эксперименты, направленные на исследование эффективности предложенных алгоритмов идентификации и управления.

Ключевые слова: многомерные объекты, непараметрическая неопределенность, идентификация, управление, непараметрические оценки, дискретно-непрерывный процесс, непараметрические алгоритмы

Введение

Настоящая статья не совсем обычная. Как известно, теория управления была изложена достаточно представительно в книгах [1–5]. Эту часть теории управления назовем параметрической теорией управления, которой она и является. Нас же будет интересовать непараметрическая теория управления, которая в той или иной степени изложена в цитируемой литературе. Основное отличие теории

управления будет заключаться в отсутствии либо наличии априорной информации о параметрической структуре исследуемого объекта. В связи с этим при решении различных задач адаптации, кибернетики уместно широкое использование информационных технологий. Это также касается некоторых задач, которые изложены в настоящем журнале "Информационные технологии" в работах [6, 7]. В. М. Глушков называл информационные технологии безбумажными. В частности, это вопрос широко обсуждался в работе [8]. Объекты, реальность, природа, безусловно, остаются главными. В сущности, именно они являются началом исходных постановок задач. Здесь

*Работа выполнена в рамках государственного задания Министерства науки и высшего образования Российской Федерации (код научной темы FSRZ-2020-0011).

уместно вспомнить высказывание А. Галлера: "... только природа никогда не отказывается нас учить... это неисчерпаемый родник, из которого черпали истину в первые века и будут черпать потомки, причем он неиссякаем. Только природа всегда нова, только она правдива, ее никогда достаточно не изучишь, но никогда не изучаешь напрасно". Важная роль, на которую следует обратить внимание, принадлежит априорной информации. В случае, если априорной информации достаточно, чтобы хорошо параметризовать исследуемый процесс, нужно использовать параметрическую теорию. В случае отсутствия такой информации естественно воспользоваться теорией непараметрических систем. Но надо учесть то обстоятельство, что "язык" этих разделов теории управления может существенно отличаться. Так, в непараметрической теории управления отсутствуют, например, алгебраические критерии Рауса, Гурвица, метод D-разбиения, метод Гольдфарба, неминимально фазовые объекты и многое другое. Но, естественно, появляются другие определения, присущие непараметрической теории управления. И следует помнить, что основой и той и другой теории является природа, действительность, реальность. Таким образом, использовать ту или другую теорию управления необходимо в условиях, в которых мы находимся. Здесь уместно вспомнить известную фразу И. Канта: "Это может показаться парадоксальным, но вся наука подчинена идее аппроксимации".

Выпишем в явном виде основные аппроксимации, соответствующие параметрической и непараметрической теориям управления:

$$y = f(x, \alpha),$$

где α — вектор параметров, например какое-нибудь разложение в ряд типа $y = \sum_{i=1}^n \alpha_i \varphi_i(x)$ или $y = f(x, x_s, y_s)$, где x_s, y_s — результаты наблюдений или обучающая выборка, причем s никогда не становится бесконечностью. В противном случае мы сталкиваемся с проблемой нестационарности.

Таким образом, в настоящее время идентификация и управление многомерными дискретно-непрерывными процессами с запаздыванием продолжают оставаться актуальными задачами [9]. К дискретно-непрерывным процессам относят процессы, протекающие во времени непрерывно, но контроль переменных осуществляется в дискретные моменты времени. Актуальность задач объясняется тем, что такие процессы типичны для реаль-

ных производств. В качестве примера можно привести различные технологические, технические, производственные процессы, например, в стройиндустрии — процесс производства цемента [10], в нефтепереработке — процесс гидроочистки дизельного топлива от сернистых соединений [11]. При этом следует учитывать, что по разным каналам многомерного объекта процессы чаще всего могут быть и динамическими, оставаясь непрерывными, но контроль выходных переменных осуществляется через дискретные интервалы времени. Примером может быть процесс измельчения клинкера (продукта обжига), который подлежит измельчению, а измельчение приводит к получению цемента. Основными параметрами, определяющими качество цемента, являются активность (прочность при сжатии), тонкость помола, удельная поверхность, распыл конуса и другие. Измерение этих показателей на производстве осуществляется через несколько часов, в то время как постоянная времени по различным каналам многомерного процесса составляет от 3 до 5 мин. Обычно на практике время переходного процесса принимается равным 4...5 постоянных времени, следовательно, переходный процесс длится в течение 20...25 мин. Это обстоятельство приводит к тому, что те или иные каналы многомерной системы вынужденно должны рассматриваться как статические с запаздыванием, а также это приводит к зависимости выходных переменных процесса, в частности, активность цемента зависит от тонкости помола и других параметров. В этом случае процесс, происходящий в многомерном объекте, будет описываться в виде некоторых неявных функций. Таким образом, многомерный объект в общем виде будет описываться в виде системы неявных функций. Особенностью настоящей задачи является то, что вид этой системы функций априори оказывается неизвестным. Поэтому необходимо решить эту систему неявных функций относительно компонент выходных переменных $x(t)$ объекта при известных входных $u(t)$. Такие многомерные процессы были названы Т-процессами, а их модели — Т-моделями [9, 10].

Проведенные вычислительные эксперименты по идентификации и управлению многомерными Т-объектами показали достаточно высокую их эффективность. В статье приведены некоторые фрагменты численных исследований по моделированию и управлению многомерным объектом.

1. Многомерные безынерционные процессы

Многомерный Т-процесс — это дискретно-непрерывный процесс, имеющий стохастическую зависимость выходных переменных. Обозначим вектор выходных переменных $x(t) = (x_1(t), x_2(t), \dots, x_j(t), \dots, x_n(t))$, $j = \overline{1, n}$, а вектор входных — $u(t) = (u_1(t), u_2(t), \dots, u_k(t), \dots, u_m(t))$, $k = \overline{1, m}$. Тогда математическое описание объекта можно представить в виде некоторой системы неявных функций следующего вида:

$$F_j(u(t), x(t + \tau), \xi(t)) = 0, \quad j = \overline{1, n}, \quad (1)$$

которая имеет единственное решение в области протекания технологического процесса, $F_j(\cdot)$ — не известны, τ — известное запаздывание по различным каналам многомерной системы. В дальнейшем, из соображений простоты, τ будет опущено, а при вычислительных экспериментах в матрице наблюдений происходит сдвиг выходных переменных на значение запаздывания, поскольку оно известно.

В этом случае задача идентификации сводится к задаче решения системы нелинейных уравнений (1) относительно компонент вектора выходных переменных при известных значениях входных. В общем виде исследуемая многомерная система, реализующая многомерный процесс с зависимыми выходными переменными, может быть представлена в виде, показанном на рис. 1.

На рис. 1 приняты следующие обозначения: $u_1(t), u_2(t), \dots, u_k(t), \dots, u_m(t)$, $k = \overline{1, m}$, — компоненты вектора $u(t)$ входных переменных; $x_1(t), x_2(t), \dots, x_j(t), \dots, x_n(t)$, $j = \overline{1, n}$, — компоненты вектора $x(t)$ выходных переменных; $\xi(t)$ — случайные помехи, действующие на объект; вертикальные стрелки свидетельствуют о стохастической зависимости выходных переменных; стрелки внутри объекта показывают зависимости входных переменных от выходных; t — непрерывное время. Измерение всех переменных процесса осуществляется с помехами.

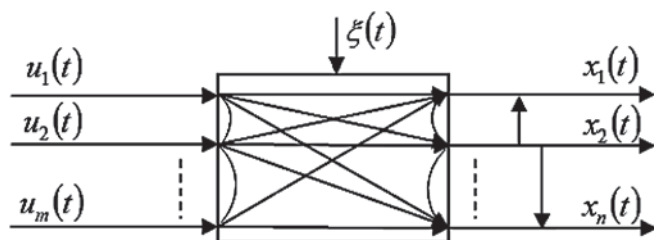


Рис. 1. Многомерный объект (процесс)

По разным каналам исследуемого многомерного объекта зависимость j -й компоненты вектора x может быть представлена в виде некоторой зависимости от тех или иных компонент вектора u : $x^{<j>} = f_j(u^{<j>})$, $j = \overline{1, n}$.

Каждый j -й канал многомерной системы может зависеть от нескольких компонент вектора u , а может и от всех, в зависимости от априорной информации. Функция $f_j(u^{<j>})$, $j = \overline{1, n}$, зависит от многих переменных $u^{<j>}$, $j = \overline{1, n}$, которые известны из априорной информации. Совокупность переменных $u^{<j>}$, $j = \overline{1, n}$, называется составным (или ситуационным) вектором. Составной вектор — это вектор, составленный из компонент векторов входных и выходных переменных. Составные векторы выписываются исследователем на основании имеющейся априорной информации. В этом случае система уравнений (1) примет вид:

$$F_j(u^{<j>}(t), x^{<j>}(t), \xi(t)) = 0, \quad j = \overline{1, n}, \quad (2)$$

где $u^{<j>}(t)$, $x^{<j>}(t)$ — составные векторы. Заметим, что вид уравнений (2) продолжает оставаться неизвестным и не может интерпретироваться как модель исследуемого процесса. Задача состоит в моделировании подобных процессов, т. е. Т-процессов [12, 13].

2. Непараметрический алгоритм идентификации

Описание процесса, показанного на рис. 1, может быть принято в виде (2). При этом особенностью моделирования подобного процесса в условиях непараметрической неопределенности является тот факт, что вид функций (2) неизвестен, так как неизвестны зависимости выходных переменных. Система моделей исследуемого процесса может быть представлена в следующем виде:

$$\hat{F}_j(u^{<j>}(t), x^{<j>}(t), x_s, u_s) = 0, \quad j = \overline{1, n}, \quad (3)$$

где x_s, u_s — временные векторы (набор данных, поступивший к s -му моменту времени), но и в этом случае функции $F_j(\cdot)$ продолжают оставаться неизвестными. В теории идентификации подобные задачи не только не рассматриваются, но и не ставятся. Обычно идут по пути выбора параметрической структуры (2), но, к сожалению, преодоление этого этапа затруднено из-за недостатка априорной информации.

Рассмотрим задачу построения Т-моделей в условиях непараметрической неопределенности, т. е. в условиях, когда система (3) не известна с точностью до параметров. Таким образом, задача идентификации сводится к тому, что при заданном значении вектора входных переменных необходимо решить систему (3) относительно вектора выходных переменных. Сделаем некоторые специальные замечания, которые и будут раскрывать отличие идентификации Т-процессов от общепринятой схемы идентификации. Оно состоит в том, что алгоритм идентификации не может быть представлен так, как обычно принято в теории идентификации: $x(t) = F(u(t), \alpha)$, где α — вектор параметров модели объекта. В случае, если α как-то оценен, то зависимость $x(t) = F(u(t), \alpha)$ является моделью исследуемого объекта. При наличии описания (2) этот путь оказывается неприемлемым. Но возможно выстроить цепочку алгоритмов, которые в своей взаимосвязи и будут представлять модель Т-процесса. Таким образом, обычная в теории идентификации формула $x(t) = F(u(t), \alpha)$ подменяется взаимосвязанной цепочкой некоторой последовательности алгоритмических соотношений, которые рассматриваются ниже.

Общая схема решения такой системы сводится к следующей последовательности действий. Сначала вычисляются невязки по формуле:

$$\varepsilon_{ij} = \varepsilon_j(u^{<j>}, x^{<j>}(i), x_s, u_s), \quad j = \overline{1, n}, \quad (4)$$

где $\varepsilon_j(\cdot)$ вычисляются с использованием непараметрической оценки функции регрессии Надарая—Ватсона [14]:

$$\begin{aligned} \varepsilon_j(i) &= \varphi_{\varepsilon_j}(u^{<j>}, x_j(i)) = \\ &= x_j(i) - \frac{\sum_{i=1}^s x_j[i] \prod_{k=1}^{<m>} \Phi\left(\frac{u'_k - u_k[i]}{c_{su_k}}\right)}{\sum_{i=1}^s \prod_{k=1}^{<m>} \Phi\left(\frac{u'_k - u_k[i]}{c_{su_k}}\right)}, \quad j = \overline{1, n}, \end{aligned} \quad (5)$$

где $<m>$ — размерность составного вектора u_k , $<m> \leq m$, в дальнейшем это обозначение используется и для других переменных.

Колоколообразные функции $\Phi(\cdot)$ и параметр размытости c_{su_k} удовлетворяют следующим условиям [14—16]:

$$\Phi(\cdot) < \infty; \quad \int_{\Omega(u)} \Phi(c_s^{-1}(u_k - u_{ki})) du = 1;$$

$$\lim_{s \rightarrow \infty} c_s \rightarrow 0;$$

$$\lim_{s \rightarrow \infty} c_s^{-1} \Phi(c_s^{-1}(u_k - u_{ki})) = \delta(u_k - u_{ki});$$

$$\lim_{s \rightarrow \infty} s c_s^m \rightarrow \infty.$$

Следующий шаг состоит в оценивании условного математического ожидания:

$$x_j = M\{x_j | u^{<j>}, \varepsilon = 0\}, \quad j = \overline{1, n}. \quad (6)$$

В конечном итоге прогноз для каждой компоненты вектора выходных переменных будет выглядеть следующим образом:

$$x_j = \frac{\sum_{i=1}^s x_j[i] \prod_{k_1=1}^{<m>} \Phi\left(\frac{u_{k_1} - u_{k_1}[i]}{c_{su}}\right) \prod_{k_2=1}^{<n>} \Phi\left(\frac{\varepsilon_{k_2}[i]}{c_{se}}\right)}{\sum_{i=1}^s \prod_{k_1=1}^{<m>} \Phi\left(\frac{u_{k_1} - u_{k_1}[i]}{c_{su}}\right) \prod_{k_2=1}^{<n>} \Phi\left(\frac{\varepsilon_{k_2}[i]}{c_{se}}\right)}, \quad (7)$$

$$j = \overline{1, n},$$

где колоколообразные функции $\Phi(\cdot)$ примем в виде треугольного ядра:

$$\begin{aligned} \Phi\left(\frac{u_{k_1} - u_{k_1}[i]}{c_{su}}\right) &= \\ &= \begin{cases} 1 - \frac{|u_{k_1} - u_{k_1}[i]|}{c_{su}}, & \frac{|u_{k_1} - u_{k_1}[i]|}{c_{su}} < 1, \\ 0, & \frac{|u_{k_1} - u_{k_1}[i]|}{c_{su}} \geq 1 \end{cases} \end{aligned} \quad (8)$$

и

$$\Phi\left(\frac{\varepsilon_{k_2}[i]}{c_{se}}\right) = \begin{cases} 1 - \frac{|0 - \varepsilon_{k_2}[i]|}{c_{se}}, & \frac{|0 - \varepsilon_{k_2}[i]|}{c_{se}} < 1, \\ 0, & \frac{|0 - \varepsilon_{k_2}[i]|}{c_{se}} \geq 1. \end{cases} \quad (9)$$

Но могут быть и другие виды ядерной функции.

В результате данной процедуры получаются значения выходных переменных x при входных воздействиях на объект u , которые в дальнейшем могут быть использованы в различных системах управления, в том числе организационных, включая образовательный процесс [17].

Точность моделирования оценивается по следующей формуле:

$$\delta_j = \left(\sum_{i=1}^s |x_{ij} - x_{sj}(u_i)| \right) / \left(\sum_{i=1}^s |x_{ij} - \bar{x}_j| \right), \quad (10)$$

$$j = \overline{1, n},$$

где x_{ij} — наблюдения на объекте; $x_{sj}(u_i)$ — прогноз выхода объекта; $\bar{x}_j$ — среднее значение по каждой компоненте вектора x .

3. Управление дискретно-непрерывным процессом

Рассмотрим задачу управления многомерным Т-объектом в условиях непараметрической неопределенности. Схема непараметрической системы управления безынерционным объектом показана на рис. 2 [18], где введены следующие обозначения: $u(t) = (u_1(t), u_2(t), \dots, u_m(t))$ — вектор входных управляемых переменных процесса; $\mu(t) = (\mu_1(t), \mu_2(t), \dots, \mu_p(t))$ — вектор входных неуправляемых, но контролируемых переменных процесса; $x(t) = (x_1(t), x_2(t), \dots, x_n(t))$ — вектор выходных переменных процесса; $x^* = (x_1^*, \dots, x_n^*) \in \Omega(x^*) \subset R^n$ — задающее воздействие, $\xi(t)$ — случайные помехи, действующие на объект.

Переменная $\mu(t)$ является входной контролируемой, но неуправляемой переменной, например это могут быть технологические параметры, для измерения которых используется физико-химические методы. Например, клин-

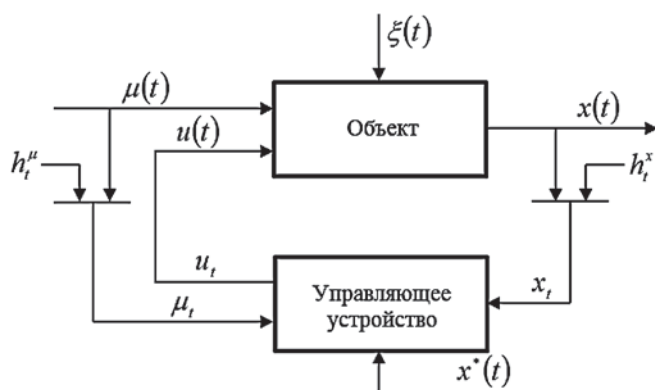


Рис. 2. Схема непараметрической системы управления безынерционным объектом

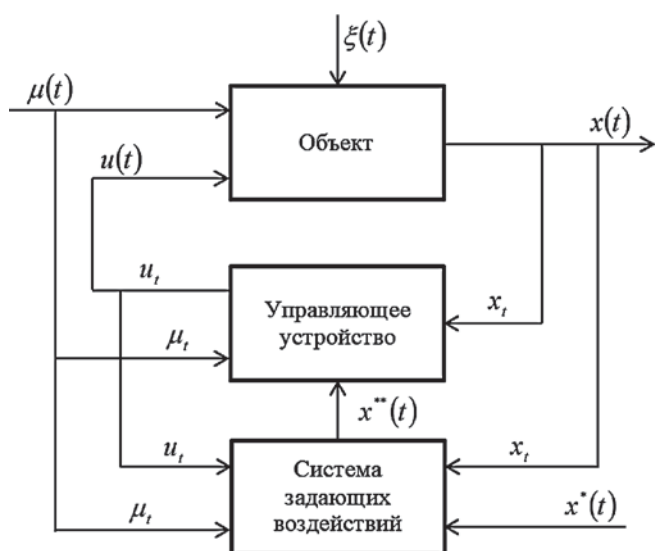


Рис. 3. Схема непараметрической системы управления

кер может быть недообожженным или переобожженным. Такая переменная существенно влияет на объект и на значения выходных переменных $x(t)$. Задача управления таким объектом усложняется, так как необходимо поддерживать на выходе объекта заданное значение $x^*(t)$ при известном значении $\mu(t)$.

Приведем систему управления многомерным объектом, в которой рассмотрим систему взаимосвязанных задающих воздействий (рис. 3).

На рис. 3 приняты следующие обозначения: $x^*(t)$ — исходные значения задающих воздействий; $x^{**}(t)$ — задающие воздействия, которые необходимо найти из системы уравнений:

$$F_j(u^{<j>}(t), \mu^{<j>}(t), x^{*<j>}(t)) = 0, j = \overline{1, n}, \quad (11)$$

где функции $F_j(\cdot)$ продолжают оставаться не известными, а $u^{<j>}, \mu^{<j>}, x^{*<j>}$ — составные векторы.

Следует учитывать, что многомерная система имеет стохастические зависимости выходных переменных, поэтому выбирать произвольно задающие воздействия x_j^{**} из областей $\Omega(x_j)$ не представляется возможным. Поэтому для выбора задающих воздействий x_j^{**} необходимо сначала определить систему задающих воздействий, как показано на рис. 3, и далее выбирать значения из общего пересечения выходных переменных $x_j^{**} \in A(x_j) \subset \Omega(x_j)$. Кратко изложим всю процедуру управления, начиная с конкретного момента времени t .

Пусть имеется обучающая выборка $(u^i, \mu^i, x^i, i = \overline{1, s})$. В конкретный момент времени t на вход поступают неуправляемые, но контролируемые переменные μ_t , при этом управляющие воздействия u_t и выходные переменные x_t неизвестны. Далее из всей обучающей первоначальной выборки $(u^i, \mu^i, x^i, i = \overline{1, s})$ выбираются те строки, в которых значения μ_i наиболее близки к вновь поступившим значениям μ_t , и из этих строк формируется новая выборка. Задающие воздействия x_j^{**} находятся из новой сформированной выборки $x_j^{**} \in A(x_j)$, а именно из решения системы (11). Решение системы (11) сводится к последовательности действий, изложенной ниже.

Для задающего воздействия x_1^{**} берем произвольные значения из области $\Omega(x_1)$ (в качестве первого задающего воздействия можно выбрать и другие выходные компоненты многомерной системы, наиболее важные, по мнению исследователя или технолога, в настоящий момент времени). Вторую переменную

x_2^{**} определяем с учетом выбранной компоненты x_1^{**} из следующего выражения:

$$x_2^{**i} = \frac{\sum_{i=1}^{s_1} x_2^i \Phi\left(\frac{x_1^{**} - x_1^i}{c_{x_1}}\right) \prod_{k=1}^{<m>} \Phi\left(\frac{u_k - u_k^i}{c_{u_k}}\right) \prod_{v=1}^{<p>} \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}{\sum_{i=1}^{s_1} \Phi\left(\frac{x_1^{**} - x_1^i}{c_{x_1}}\right) \prod_{k=1}^{<m>} \Phi\left(\frac{u_k - u_k^i}{c_{u_k}}\right) \prod_{v=1}^{<p>} \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}, \quad (12)$$

где $s_1 \subseteq \Omega(x_1)$, т. е. суммирование проводится не по всей первоначальной выборке, а только по тем значениям, которые были наиболее близки к вновь поступившим значениям μ_v ; скобки "< >" в выражении обозначают составной вектор, т. е. используются только те компоненты входных и выходных переменных, которые влияют на второй выход, если такой информации нет, то используются все входные и выходные переменные.

В общем виде процедура вычислений принимает следующий вид:

$$x_j^{**s} = \frac{\sum_{i=1}^{s_{j-1}} x_j^i \prod_{j=1}^{i-1} \left(\frac{x_j^{**} - x_j^i}{c_{x_j}}\right) \prod_{k=1}^{<m>} \Phi\left(\frac{u_k - u_k^i}{c_{u_k}}\right) \prod_{v=1}^{<p>} \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}{\sum_{i=1}^{s_{j-1}} \prod_{j=1}^{i-1} \left(\frac{x_j^{**} - x_j^i}{c_{x_j}}\right) \prod_{k=1}^{<m>} \Phi\left(\frac{u_k - u_k^i}{c_{u_k}}\right) \prod_{v=1}^{<p>} \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}. \quad (13)$$

Управление дискретно-непрерывным процессом рассматривается в условиях непараметрической неопределенности, т. е. в условиях, когда модель процесса с точностью до вектора параметров не определена из-за недостатка априорной информации. В этом случае известные приемы не применимы [1–5] и следует использовать другие подходы для решения задачи [19, 20]. Поэтому для многомерной системы естественно использовать нижеследующую последовательность действий для нахождения управляющего воздействия. Входную переменную $u_1(t)$ берем произвольно из области $\Omega(u_1)$. Входная переменная $u_2(t)$ может быть определена следующим образом:

$$u_2 = \frac{\sum_{i=1}^s u_2^i \Phi\left(\frac{u_1 - u_1^i}{c_{u_1}}\right) \prod_{j=1}^n \Phi\left(\frac{x_j^{**} - x_j^i}{c_{x_j}}\right) \prod_{v=1}^p \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}{\sum_{i=1}^s \Phi\left(\frac{u_1 - u_1^i}{c_{u_1}}\right) \prod_{j=1}^n \Phi\left(\frac{x_j^{**} - x_j^i}{c_{x_j}}\right) \prod_{v=1}^p \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}, \quad (14)$$

где $(u^i, \mu^i, x^i, i = \overline{1, s})$ — обучающая выборка; μ_v — поступившие входные неуправляемые, но контролируемые переменные.

Вычислительная процедура для $u_3(t)$ будет выглядеть следующим образом:

$$u_3 = \frac{\sum_{i=1}^s u_3^i \Phi\left(\frac{u_1 - u_1^i}{c_{u_1}}\right) \Phi\left(\frac{u_2 - u_2^i}{c_{u_2}}\right) \times \prod_{j=1}^n \Phi\left(\frac{x_j^{**} - x_j^i}{c_{x_j}}\right) \prod_{v=1}^p \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}{\sum_{i=1}^s \Phi\left(\frac{u_1 - u_1^i}{c_{u_1}}\right) \Phi\left(\frac{u_2 - u_2^i}{c_{u_2}}\right) \times \prod_{j=1}^n \Phi\left(\frac{x_j^{**} - x_j^i}{c_{x_j}}\right) \prod_{v=1}^p \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}. \quad (15)$$

И так далее для каждой компоненты входа $u(t)$ объекта. В общем виде для многомерной системы выражение для вычисления управляющих воздействий будет выглядеть следующим образом:

$$u_s^k = \frac{\sum_{i=1}^s u_s^i \prod_{k=1}^{k-1} \Phi\left(\frac{u_k - u_k^i}{c_{u_k}}\right) \times \prod_{j=1}^n \Phi\left(\frac{x_j^{**} - x_j^i}{c_{x_j}}\right) \prod_{v=1}^p \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}{\sum_{i=1}^s \prod_{k=1}^{k-1} \Phi\left(\frac{u_k - u_k^i}{c_{u_k}}\right) \times \prod_{j=1}^n \Phi\left(\frac{x_j^{**} - x_j^i}{c_{x_j}}\right) \prod_{v=1}^p \Phi\left(\frac{\mu_v - \mu_v^i}{c_{\mu_v}}\right)}, \quad (16)$$

$$k = \overline{1, m}.$$

В реальных задачах часто число компонент вектора входных переменных u больше числа компонент вектора выходных переменных x . В этом случае в число компонент вектора μ включают компоненты вектора u , с тем, чтобы размерность вектора u и x сделать одинаковой.

Настраиваемыми параметрами будут параметры размытости c_{u_k} , c_{x_j} и c_{μ_v} , для них будем использовать следующие формулы: $c_{u_k} = \alpha |u_k - u_k^i|$, $c_{x_j} = \beta |x_j^{**} - x_j^i|$ и $c_{\mu_v} = \gamma |\mu_v - \mu_v^i|$, где α , β и γ — некоторые параметры, $\alpha > 1$, $\beta > 1$, $\gamma > 1$. Следует заметить, что выбор c_{u_k} , c_{x_j} и c_{μ_v} осуществляется на каждом такте управления. При этом, если сначала определен c_{u_k} , то определение c_{x_j} и c_{μ_v} осуществляется с учетом этого факта. Однако может быть и

наоборот, например, сначала определяется c_{x_j} или c_{μ_j} , а потом остальные.

Приведенные в статье задачи идентификации и управления не рассматривались ранее и могут быть изучены в рамках непараметрической теории управления.

4. Вычислительные эксперименты

Для проведения вычислительного эксперимента в качестве примера был взят объект с четырьмя входными переменными $u(t) = (u_1(t), u_2(t), u_3(t), u_4(t))$ и тремя выходными переменными $x(t) = (x_1(t), x_2(t), x_3(t))$. Для данного объекта была сформирована выборка входных

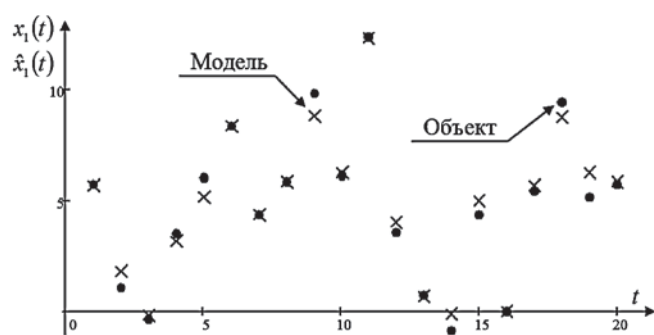


Рис. 4. Прогноз выходной переменной $x_1(t)$ объекта, измеренный с равномерной помехой 10 %, $\delta_1 = 0,06$

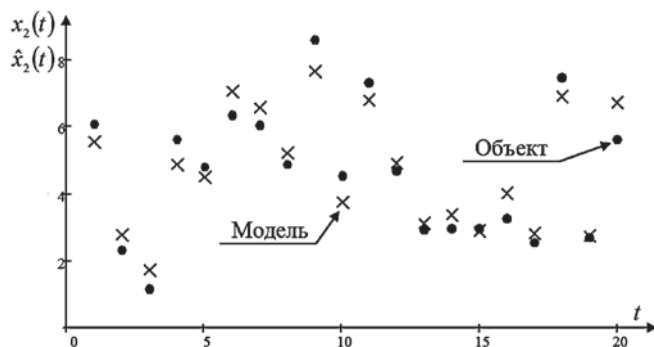


Рис. 5. Прогноз выходной переменной $x_2(t)$ объекта, измеренный с равномерной помехой 10 %, $\delta_2 = 0,08$

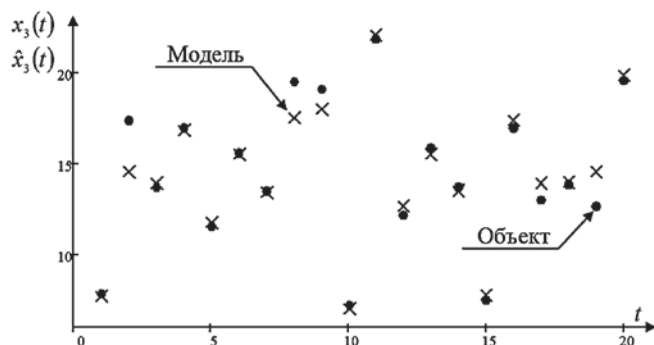


Рис. 6. Прогноз выходной переменной $x_3(t)$ объекта, измеренный с равномерной помехой 10 %, $\delta_3 = 0,07$

и выходных переменных и найдены прогнозные значения выходных переменных при известных входных с использованием формул (5) и (7). Для вычисления принимали значения выборки $s = 1000$, параметров размытости $c_{su} = 0,4$; $c_{se} = 0,2$, помехи, действующей на компоненты вектора выходных переменных, $\alpha = 0,1$. Описание объекта с точностью до параметров было принято только для проведения компьютерного исследования, и оставалось неизвестным для изложенной выше теории.

На рис. 4–6 по оси абсцисс представлены элементы каждой 50 точки из имеющейся выборки, по оси ординат — значения выхода объекта и модели. На рисунках представлены 20 точек из имеющейся выборки из-за простоты представления результатов, т. е. каждая пятидесятая точка выборки объемом $s = 1000$. Погрешности определения переменных обусловлены наличием помех. Как видно из представленных рисунков, прогноз значений выходных переменных многомерного объекта по известным входным переменным с учетом 10 % помехи достаточно точен. Обратим еще раз внимание на то, что исследователю неизвестен вид системы уравнений, описывающей управляемый объект. В качестве информации о последнем используются измерения входных и выходных переменных $(u^i, \mu^i, x^i, i = \overline{1, s})$.

Далее приводятся результаты вычислительных экспериментов для данного объекта при использовании вычислительной процедуры (16). В данном вычислительном эксперименте число компонент вектора u больше числа компонент вектора x . Если же размерность вектора u превышает размерность вектора x , т. е. $m > n$, то сделаем замену: $u_4(t) \rightarrow \mu_1(t)$, а $u_5(t) \rightarrow \mu_2(t)$, чтобы размерность вектора u и x сделать одинаковой. Так как входные переменные принимали случайные значения в интервале $u(t) \in [0, 3]$, то $\mu_1(t)$ и $\mu_2(t)$ также принимают случайные значения в заданном интервале.

С использованием формулы (13) находим задающие воздействия для данного объекта. Далее значения задающего воздействия $x_1^{**}(t)$ представим в виде ступенчатой функции (рис. 7).

Как видно из рис. 7, при управлении объектом выход объекта $x_1(t)$ близок к задающему воздействию $x_1^{**}(t)$.

Значения задающего воздействия $x_2^{**}(t)$ представим в виде плавно изменяющейся функции (рис. 8).

Из рис. 8 можно увидеть, что выход объекта $x_2(t)$ также близок к задающему воздействию $x_2^{**}(t)$.

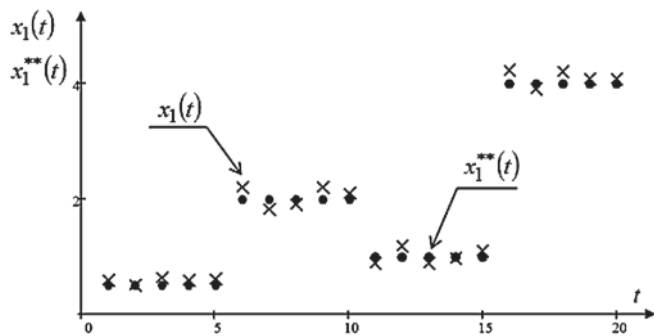


Рис. 7. Зависимость выхода объекта $x_1(t)$ от задающего воздействия $x_1^{**}(t)$

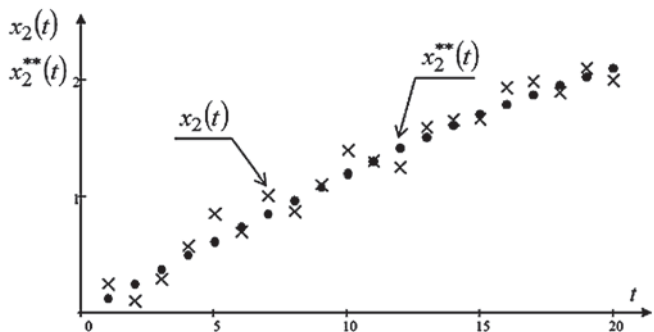


Рис. 8. Зависимость выхода объекта $x_2(t)$ от задающего воздействия $x_2^{**}(t)$

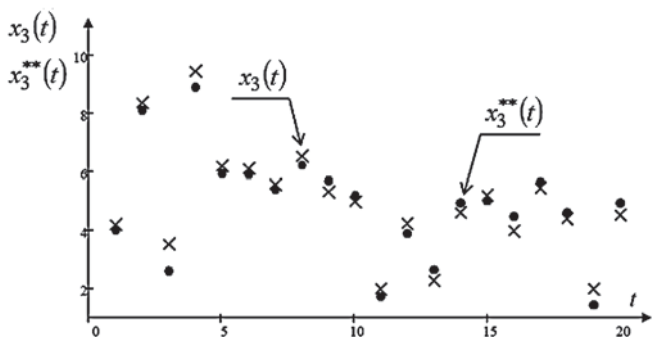


Рис. 9. Зависимость выхода объекта $x_3(t)$ от задающего воздействия $x_3^{**}(t)$

Значения задающего воздействия $x_3^{**}(t)$ представим в виде, показанном на рис. 9.

Из рис. 9 можно увидеть, что выход объекта $x_3(t)$ также близок к задающему воздействию $x_3^{**}(t)$.

Таким образом, непараметрический алгоритм управления многомерным объектом со стохастически зависимыми выходными переменными показывает достаточно удовлетворительные результаты.

Заключение

В настоящей работе рассматривается задача идентификации и управления многомерными

безынерционными объектами в условиях неполной информации об объекте исследования. Следует отметить, что здесь излагается теория, которая названа непараметрической. Особенностью изложенной выше теории является то, что и при идентификации и при управлении используются последовательности непараметрических формул. Суть состоит в следующем: для задачи идентификации прогноз выстраивается как условное математическое ожидание выходных переменных x при известных входных u , а для задачи управления — как условное математическое ожидание входных переменных u при известных задающих воздействиях x^{**} .

Таким образом, рассмотренные некоторые фрагменты численных исследований показали достаточно хорошие результаты предлагаемых последовательностей алгоритмов идентификации и управления многомерными объектами. При статистическом моделировании алгоритмов идентификации и управления рассматривалось и исследовалось влияние на объект различных случайных факторов при различных объемах обучающей выборки, различных способах и приемах оценивания коэффициентов размытости. Также сравнительно произвольно изменялись и размерности многомерных объектов, положенные в основу статистического моделирования.

Список литературы

1. **Методы** классической и современной теории автоматического управления. Т. 1: Математические модели, динамические характеристики и анализ систем управления / Под. ред. К. А. Пупкова, Н. Д. Егупова. М.: Изд-во МГТУ им. Н. Э. Баумана, 2004. 748 с.
2. **Методы** классической и современной теории автоматического управления. Т. 2: Статистическая динамика и идентификация систем автоматического управления / Под. ред. К. А. Пупкова, Н. Д. Егупова. М.: Изд-во МГТУ им. Н. Э. Баумана, 2004. 640 с.
3. **Методы** классической и современной теории автоматического управления. Т. 3: Синтез регуляторов систем автоматического управления / Под. ред. К. А. Пупкова, Н. Д. Егупова. М.: Изд-во МГТУ им. Н. Э. Баумана, 2004. 748 с.
4. **Методы** классической и современной теории автоматического управления. Т. 4: Теория оптимизации систем автоматического управления / Под. ред. К. А. Пупкова, Н. Д. Егупова. М.: Изд-во МГТУ им. Н. Э. Баумана, 2004. 748 с.
5. **Методы** классической и современной теории автоматического управления. Т. 5: Методы современной теории автоматического управления / Под. ред. К. А. Пупкова, Н. Д. Егупова. М.: Изд-во МГТУ им. Н. Э. Баумана, 2004. 742 с.

6. Гвоздкова Ю. Д., Гвоздкова И. А., Курочкин А. В., Черняев А. В. Информационная система оценки экологической безопасности авиационных материалов и технологий методом анализа иерархий // Информационные технологии. 2019. № 3 (25). С. 185—192.
7. Чертовской В. Д. Методология математического описания и моделирования адаптивной автоматизированной системы управления производством // Информационные технологии. 2018. № 2 (24). С. 81—86.
8. Медведев А. В. Информатизация управления / Учеб. пособ. Красноярск: САА, 1995. 80 с.
9. Медведев А. В. Основы теории адаптивных систем. Красноярск: Сиб. гос. аэрокосмич. ун-т., 2015. 526 с.
10. Медведев А. В. Основы теории непараметрических систем. Красноярск: Сиб. гос. аэрокосмич. ун-т., 2018. 727 с.
11. Агафонов Е. Д., Медведев А. В., Орловская Н. Ф., Синюта В. Р., Ярешенко Д. И. Прогнозная модель процесса каталитической гидродепарафинизации в условиях недостатка априорных сведений // Известия Тульского государственного университета. Технические науки. 2018. № 9. С. 456—468.
12. Медведев А. В. К теории непараметрических систем // Вестник СибГАУ. 2010. Вып. 5 (31). С. 4—9.
13. Медведев А. В., Ярешенко Д. И. Непараметрическое моделирование Т-процессов в условиях неполной информации // Информационные технологии. 2019. № 10 (25). С. 579—584.
14. Надарая Э. А. Непараметрическое оценивание плотности вероятностей и кривой регрессии. Тбилиси: Изд-во Тбилисского ун-та, 1983. 194 с.
15. Катковник В. Я. Непараметрическая идентификация и сглаживание данных: метод локальной аппроксимации. М.: Главная редакция физико-математической литературы, 1985. 336 с.
16. Кошкин Г. М., Пивен И. Г. Непараметрическая идентификация стохастических объектов. Хабаровск: Дальневост. отд-ние Рос. акад. наук, 2009. 336 с.
17. Ярешенко Д. И. Некоторые замечания об оценке знаний студентов университетов // Открытое образование. 2017. Вып. 4. С. 66—72.
18. Медведев А. В. О теории непараметрических систем управления // Вестник Томского государственного университета. Управление, вычисл. техника и информатика. 2013. № 1 (22). С. 6—19.
19. Raskina A. V., Medvedev A. V. On the nonparametric identification and dual adaptive control of dynamic processes // Journal of Siberian Federal University. Mathematics & Physics. 2017. Vol. 10, N. 1. P. 96—107.
20. Medvedev A. V., Chzhan E. A. About nonparametric modeling of multidimensional noninertial systems with delay // Вестник ЮУрГУ. Математическое моделирование. 2017. Vol. 10, N. 2. P. 124—136.

A. V. Medvedev, Dr. Sc., Professor, e-mail: mav2745@mail.ru,
D. I. Yareshchenko, Senior Lecturer, e-mail: yareshenkodi@yandex.ru,
Siberian Federal University, Krasnoyarsk, Russian Federation

Identification and Control of Multidimensional Delayed Processes in Conditions of Incomplete Information

Problems of identification and control of multidimensional discrete-continuous processes with delay in conditions of incomplete information about the object are considered. In such conditions, the form of parametric equations for various channels of the object is absent due to the lack of a priori information. Moreover, multidimensional processes have stochastic dependences of the components of the vector of output variables. Under such conditions, the mathematical description of such processes leads to a system of implicit equations. Nonparametric identification and control algorithms for multidimensional systems are proposed.

The main task of modeling such processes is to determine the predicted values of the output variables from the known input. Moreover, for implicit equations, it is only known that one or another output variable can depend on other input and output variables that determine the state of a multidimensional system.

In this study, a nontrivial situation arises when solving a system of implicit equations under conditions when the dependences between the components of the output variables are unknown. The application of the parametric theory of identification in this case will not lead to success. One of the possible directions is the use of the theory of nonparametric systems. The main content of the work is the solution of the identification problem in the presence of dependencies of the output variables and then the solution of the control problem for such a process. Here you should pay attention to the fact that when determining the reference actions for a multidimensional system, it is first necessary to solve the system of reference actions, since it is not possible to choose arbitrarily setting influences from the range of definition of output variables.

Computational experiments aimed at investigating the effectiveness of the proposed identification and control algorithms are presented.

Keywords: multidimensional objects, nonparametric uncertainty, identification, control, nonparametric estimates, discrete-continuous process, nonparametric algorithms

DOI: 10.17587/it.27.658-667

References

1. **Pupkov K. A., Egupov N. D.** ed. Methods of classical and modern theory of automatic control: in 5 volumes. Vol. 1: Mathematical models, dynamic characteristics and analysis of control systems, Moscow, Publishing house of MGTU im. N. E. Bauman, 2004, 748 p. (in Russian).
2. **Pupkov K. A., Egupov N. D.** ed. Methods of classical and modern theory of automatic control: in 5 volumes. Vol. 2: Statistical dynamics and identification of automatic control systems, Moscow, Publishing house of MGTU im. N. E. Bauman, 2004, 640 p. (in Russian).
3. **Pupkov K. A., Egupov N. D.** ed. Methods of classical and modern theory of automatic control: in 5 volumes. Vol. 3: Synthesis of regulators of automatic control systems, Moscow, Publishing house of MGTU im. N. E. Bauman, 2004, 748 p. (in Russian).
4. **Pupkov K. A., Egupov N. D.** ed. Methods of classical and modern theory of automatic control: in 5 volumes. Vol. 4: Optimization theory of automatic control systems, Moscow, Publishing house of MGTU im. N. E. Bauman, 2004, 748 p. (in Russian).
5. **Pupkov K. A., Egupov N. D.** ed. Methods of classical and modern theory of automatic control: in 5 volumes. Vol. 5: Methods of modern theory of automatic control, Moscow, Publishing house of MGTU im. N. E. Bauman, 2004, 742 p. (in Russian).
6. **Gvozdkova Yu. D., Gvozdkova I. A., Kurochkin A. V., Chernyaev A. V.** Information system for assessing the environmental safety of aviation materials and technologies by the method of hierarchy analysis, *Informatsionnye Tekhnologii*, vol. 25, 2019, no. 3, pp. 185–192 (in Russian).
7. **Chertovskoy V. D.** Methodology for mathematical description and modeling of an adaptive automated production control system, *Informatsionnye Tekhnologii*, vol. 24, 2018, no. 2, pp. 81–86 (in Russian).
8. **Medvedev A. V.** Management informatization, Krasnoyarsk, SAA, 1995. 80 p. (in Russian).
9. **Medvedev A. V.** Fundamentals of the theory of adaptive systems, Krasnoyarsk, Sibirskiy gosudarstvennyy aerokosmicheskiy universitet, 2015, 526 p. (in Russian).
10. **Medvedev A. V.** Fundamentals of the theory of nonparametric systems. Krasnoyarsk, Sibirskiy gosudarstvennyy aerokosmicheskiy universitet, 2018, 727 p. (in Russian).
11. **Agafonov E. D., Medvedev A. V., Orlovskaya N. F., Sinyuta V. R., Yareshchenko D. I.** Predictive model of the catalytic hydrodewaxing process under conditions of a lack of a priori information, *Izvestiya Tul'skogo gosudarstvennogo universiteta. Tekhnicheskie nauki*, 2018, no. 9, pp. 456–468 (in Russian).
12. **Medvedev A. V.** To the theory of nonparametric systems, *Vestnik SibGAU*, vol. 31, 2010, no. 5, pp. 4–9 (in Russian).
13. **Medvedev A. V., Yareshchenko D. I.** Nonparametric modeling of T-processes in conditions of incomplete information, *Informatsionnye Tekhnologii*, 2019, vol. 25, no. 10, pp. 579–584 (in Russian).
14. **Nadaraya E. A.** Nonparametric Estimation of Probability Density and Regression Curve, Tbilisi, Publishing house of Tbilisskiy un-t, 1983, 194 p.
15. **Katkovnik V. Ya.** *Nonparametric identification and data smoothing: local approximation method.* Moscow, Glavnaya redaktsiya fiziko-matematicheskoy literatury, 1985, 336 p. (in Russian).
16. **Koshkin G. M., Piven I. G.** Nonparametric identification of stochastic objects, Khabarovsk, Dal'nevost. otd-nie Ros. akad. nauk, 2009, 336 p. (in Russian).
17. **Yareshchenko D. I.** Some notes on the assessment of knowledge of university students, *Otkrytoe obrazovanie*, 2017, no. 4, pp. 66–72 (in Russian).
18. **Medvedev A. V.** On the theory of nonparametric control systems, *Vestnik Tomskogo gosudarstvennogo universiteta, Upravlenie, Vychisl. Tekhnika i Informatika*, 2013, vol. 22, no. 1, pp. 6–19 (in Russian).
19. **Medvedev A. V., Raskina A. V.** On the nonparametric identification and dual adaptive control of dynamic processes, *Journal of Siberian Federal University. Mathematics & Physics*, 2017, vol. 10, no. 1, pp. 96–107.
20. **Medvedev A. V., Chzhan E. A.** About nonparametric modeling of multidimensional noninertial systems with delay, *Vestnik YuUrGU. Matematicheskoe modelirovanie*, 2017, vol. 10, no. 2, pp. 124–136.

УКАЗАТЕЛЬ СТАТЕЙ, опубликованных в журнале "Информационные технологии" в 2021 г.

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Ульянов М. В., Фомичёв М. И.** Исследование особенностей применения комбинированного алгоритма для решения асимметричной задачи коммивояжёра. № 1
- Абрамешин Д. А., Пожидаев Е. Д., Тумковский С. Р.** Моделирование радиационного заряжения корпусов микроэлектронной аппаратуры космического применения. . . № 2
- Акопов А. С., Бекларян Л. А., Бекларян А. Л., Белюсов Ф. А.** Моделирование движения ансамбля наземных беспилотных транспортных средств с использованием FLAME GPU № 7
- Голосов П. Е., Гостев И. М.** Оптимизация распределения потока задач поиска хеш-решений при априорно заданной сложности решений. № 5
- Гридин В. Н., Анисимов В. И., Васильев С. А.** Методы расчета переходных процессов в больших электронных схемах на основе диакоптики № 11
- Зак Ю. А.** Многостадийные open-shop-problem: расписания выполнения N заданий на K различных по техническим характеристикам машинах при произвольном порядке выполнения заданий № 5
- Колесникова С. И., Караванова С. А.** Оптимизация алгоритма многокритериального выбора с динамически пополняемым большим набором альтернатив № 5
- Курейчик В. В., Родзин С. И.** Вычислительные модели эволюционных и роевых биоэвристик (обзор) № 10
- Курейчик В. В., Родзин С. И.** Вычислительные модели биоэвристик, основанных на физических и когнитивных процессах (обзор). . . № 11
- Маслов О. Н.** Защита стационарного объекта от массированного воздействия мобильных

- объектов в условиях смешанной игры по фон Нейману № 10
- Мезенцев Ю. А., Короткова Ю. Л., Эстрайх И. В.** Эффективный алгоритм решения прикладной задачи оптимизации расписаний параллельно-последовательной системы . . . № 12
- Обухов А. Д., Архипов А. Е., Сидорчук А. О.** Математическое моделирование и визуализация процессов горения в тренажерных комплексах № 5
- Переварюха А. Ю.** Формирование гибридной вычислительной модели для анализа критических сценариев управления в биокибернетике № 7
- Сидоров С. М.** Скрытая марковская модель двухкомпонентной системы с групповым мгновенно пополняемым резервом времени № 2

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Гостева Е. А., Ланин В. В.** Реализация системы интеллектуального поиска информации в промышленных стандартах. № 6
- Израилов К. Е.** Обобщенная классификация уязвимостей интерфейсов транспортной инфраструктуры Умного города № 6
- Кац В. А., Волков А. А.** Метод обучения интеллектуальной системы диагностического мониторинга технического состояния строительных конструкций № 7
- Корней А. О., Крючкова Е. Н.** Категоризация текстов на основе сконденсированного графа № 3
- Куляшова Н. М.** Программное обеспечение и алгоритмы распознавания адресных структур № 5
- Нгуен Т. В., Кравец А. Г.** Оценка и прогнозирование тенденций развития научных исследований на основе библиометрического анализа публикаций № 4
- Поляков А. Н., Позевалкин В. В.** Применение нейронной сети прямого распространения к прогнозированию тепловых характеристик станков № 4

- Салибебян С. М.** Объектно-атрибутивный подход для семантического анализа естественного языка № 5
- Сапаров А. Ю., Бельтюков А. П., Маслов С. Г.** Сравнение текстов программ с открытым кодом при обновлении с использованием расстояния Дамерау–Левенштейна . . . № 9
- Сорокин А. А.** Исследование иерархических систем нечеткого вывода при получении интегральных оценок анализируемых объектов № 11

НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

- Афонин В. Л., Слепцов В. В.** Нейронная сеть на основе специальной потенциальной функции для распознавания многофункциональных сигналов № 9

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

- Григорьев А. Д., Гнеушев А. Н.** Ре-идентификация с предфильтрацией по качеству изображений в задаче слежения за множеством объектов № 8
- Дворников С. В., Пшеничников А. В., Манаенко С. С., Глухих И. Н.** Формирование сигнальных конструкций сложных структур с высоким уровнем неопределенности их параметров № 3
- Ефанов Д. В., Сапожников В. В., Сапожников Вл. В.** Организация схем встроенного контроля на основе метода логического дополнения с предварительным преобразованием рабочих функций в контрольные векторы кодов Бергера № 6
- Жукова Г. Н., Ульянов М. В.** Восстановление символьной периодической последовательности по последовательности с шумом . . № 10
- Игнатьев Н. А., Лолаев М. Я.** Анализ соответствия структур отношений объектов классов на многообразиях их описаний . . . № 1
- Нарцев Д. Ю., Гнеушев А. Н.** Сравнение модифицированных методов обучения Adam в

- задачах оценки параметров регрессионных моделей по изображению. № 9
- Осипович В. С., Медведев О. С., Дудич О. Н., Красильникова В. Л., Яшин К. Д.** Технология преобразования результатов компьютерной томографии в трехмерные модели № 11
- Фахми Ш. С., Рыжов Н. Г., Костикова Е. В.** Новые алгоритмы сжатия без потерь транспортных сюжетов. № 6

ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ

- Грешников И. И.** Внедрение функции управления взглядом в информационно-управляющее поле кабины пилотов № 8
- Гриняк В. М.** Проектирование системы 3D-позиционирования внутри помещений на основе Bluetooth-устройств № 1
- Гриняк В. М., Шуленина А. В.** Кластеризация данных траекторий морских судов для планирования маршрутов через акватории с интенсивным движением. № 11
- Данилов К. В., Мальцева С. В.** Метод автоматической генерации признакового пространства в задаче прогнозирования потребления электроэнергии № 10
- Доронина Е. Б., Скاتков А. В.** Задача анализа эффективности ремонтно-профилактических работ сложной технической аппаратуры. № 1
- Комаров В. А., Сарафанов А. В., Тумковский С. Р.** Опыт цифровой трансформации бизнес-процессов прикладных экспериментальных исследований посредством мультиарендности их ресурсного обеспечения № 1

ИНФОРМАЦИОННО- ТЕЛЕКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ

- Гумбатов Р. Т., Ибрагимов Б. Г., Алиева А. А., Ибрагимов Р. Ф.** Подходы к анализу показателей производительности мультисервисных телекоммуникационных сетей на базе технологии SDN № 8

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ЭКОНОМИКЕ, УПРАВЛЕНИИ И ПРОИЗВОДСТВЕ

- Коковин В. А., Евсиков А. А., Леонов А. П.** Особенности организации и взаимодействия функциональных сетевых компонентов в распределенных управляющих системах . . . № 4
- Медведев А. В., Ярещенко Д. И.** Идентификация и управление многомерными процессами с запаздыванием в условиях неполной информации № 12
- Пашенко Д. С.** Влияние тенденций отрасли информационных технологий на развитие и трансформацию промышленных предприятий № 7

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИОМЕДИЦИНСКИХ СИСТЕМАХ

- Нагорнов Н. Н.** Определение минимальной разрядности коэффициентов вейвлет-фильтров для трехмерной медицинской визуализации № 8
- Терентьев А. Б., Штурц И. В.** Устранение алиасинга в доплеровской эхокардиографии с помощью фильтрации субмаксимальных компонент скоростей № 2
- Федосов В. В., Федосова А. В., Buitrago O.** Стохастическая оценка развития эпидемии в локальных скоплениях населения № 8
- Щербань О. Г., Лазуренко Д. М., Щербань И. В., Кириленко Н. Е.** Синтез адаптивного фильтра для автоматической детекции и локализации по оси времени премоторных ЭЭГ-паттернов мозга человека № 9

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

- Авдошин С. М., Песоцкая Е. Ю., Куруппуге Д. М.** Выбор МООС для российских ИТ-специалистов при планировании карьеры . . . № 2
- Вишнеков А. В., Ерохина Е. А., Иванова Е. М.** Опыт применения цифровых технологий при

переходе базового университетского образования на онлайн-формат обучения № 9

- Колокольцев М. А., Михалёва У. А.** Автоматизированная информационная система "Распределение учебной нагрузки преподавателя" № 3

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

- Бибилло П. Н., Романов В. И.** Минимизация BDD-представлений систем полностью определенных булевых функций с использованием алгебраических разложений кофакторов № 8
- Буй В. Ф., Гаврюшин С. С., Фунг В. Б., Данг Х. М., Прокопов В. С.** Автоматизация и управление процессом проектирования главного привода моечной машины нового типа для овощей и фруктов № 1
- Щёкин А. В.** Алгебра элементов технологического процесса № 6

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

- Абрамов А. Г., Гончар А. А., Евсеев А. В., Шабанов Б. М.** Национальная исследовательская компьютерная сеть нового поколения: текущее состояние и концепция развития . . . № 3
- Бычков И. В., Горский С. А., Феоктистов А. Г., Костромин Р. О.** Поддержка вычислений в распределенных средах на основе непрерывной интеграции № 12
- Левченко Н. Н., Змеев Д. Н.** Динамический контроль целостности вычислительного процесса в параллельной потоковой вычислительной системе № 12
- Ляхов П. А., Ионисян А. С., Валуева М. В., Ларикова А. С.** Высокопроизводительная цифровая фильтрация на модифицированных умножителях с накоплением в системе остаточных классов с модулями специального вида № 4
- Тарасов В. Н.** Системы массового обслуживания с запаздыванием во времени № 6

Тарасов В. Н., Бахарева Н. Ф. Математическая модель задержки в сетях связи на основе СМО с запаздыванием во времени . . . № 12

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

Иванников А. Д. Эмуляторы как средство отладки программного обеспечения для цифровых систем управления объектами. . № 7

Кулагин В. П., Логинов А. А. Анализ программных средств для работы с сетями Петри . № 2

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ

Асратян Р. Э. Сетевая служба параллельной обработки защищенных информационных запросов в мультисерверной среде. . . № 11

Коляда А. А., Кучинский П. В., Протасеня С. Ю. Метод и алгоритм выполнения декодирующей операции в пороговом криптомодуле разделения секрета с использованием минимально избыточной модулярной системы счисления № 2

Мартынов В. В., Ширяев О. В. Обеспечение комплексной безопасности в рамках построения цифровой архитектуры предприятий. № 12

Саксонов Е. А. Способ обезличивания персональных данных № 6

Сухопаров М. Е., Лебедев И. С., Салахутдинова К. И. Метод идентификации состояния

информационной безопасности устройств интернета вещей № 2

Хорев П. Б., Сергеев А. В. Анализ свойств и критерии обнаружения скрытых данных в документах Microsoft Word. № 9

Шиловских П. А. Протокол цифровой подписи на основе криптокодовых конструкций . . № 3

БАЗЫ ДАННЫХ

Аршинский Л. В., Шурховецкий Г. Н. Особенности применения метода рассеечения–разнесения для безопасного хранения данных во внешних хранилищах № 5

Малаховецкий Д. В., Разумовский А. И. Синтаксический анализ символьных массивов методом рекурсивного охвата и структурирования на примере данных VRML-формата № 4

Набибекова Г. Ч. Применение технологии OLAP в среде электронной демографии. . . . № 10

Пенькова Т. Г., Кочетков С. Н. Технологический подход к организации унифицированного обмена данными между гетерогенными системами № 1

Сапунов В. В., Ботман С. А., Камышов Г. В., Шушарина Н. Н. Применение свёртки с периодическим граничным условием для обработки данных от цилиндрических массивов электродов. № 3

Шайтура С. В., Галкин Д. А. Геомаркетинговый анализ больших данных. № 4



24–25 февраля 2022 г. в г. Барнаул пройдет

II Международный научно-практический форум по передовым достижениям в науке и технике (SciTech 2022)

Организатор форума: Ассоциация поддержки научных исследований.

Цель форума — теоретический анализ фундаментальных и прикладных вопросов научно-технологического развития России, прорывных технологий и передовых научных исследований в области органической и неорганической химии, физики, энергетики, материаловедения, компьютерных, инженерных, а также смежных наук.

Материалы форума будут опубликованы в коллективных научных монографиях, издаваемых в рамках книжных серий ведущего мирового издательства «Springer Nature». Каждая глава будет проиндексирована в наукометрических базах "Scopus" (Q3-Q4) и "Web of Science". Специально отобранные статьи будут отправлены на публикацию в высокорейтинговые научные журналы.

ПРОБЛЕМАТИКА И СЕКЦИИ

Форум станет уникальной площадкой для обсуждения актуальных вопросов научно-технологического развития России, прорывных технологий и передовых научных исследований.

СЕКЦИИ:

1. Новая парадигма научно-технологического развития России и мира.
2. Достижения в области интеллектуальных систем и вычислений.
3. "Умные" системы и инновации.
4. Фундаментальные и прикладные исследования в точных науках.
5. Новые материалы и технологии: проблемы, достижения, перспективы.
6. Будущее энергетики: в поисках инновационной и устойчивой модели развития.
7. Сельскохозяйственные и продовольственные системы: технологические вызовы и перспективы развития.
8. Современное STEM-образование для опережающего развития.

КОНТАКТЫ

Петерс Кристина Андреевна

Тел.: +7 (3852) 73-05-82. E-mail: mail@conferences.science

<https://conferences.science/conferences/scitech-2022.html>

Адрес редакции:

107076, Москва, Матросская тишина, 23с2

Телефон редакции журнала **+7 916 392 2167**

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова*.

Корректор *М. Ю. Безменова*.

Сдано в набор 11.10.2021. Подписано в печать 22.11.2021. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT1221. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Авансд солюшнз". Отпечатано в ООО "Авансд солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1. Сайт: www.aov.ru

Рисунок к статье И. В. Бычкова, С. А. Горского, А. Г. Феоктистова, Р. О. Костромина

«ПОДДЕРЖКА ВЫЧИСЛЕНИЙ В РАСПРЕДЕЛЕННЫХ СРЕДАХ НА ОСНОВЕ НЕПРЕРЫВНОЙ ИНТЕГРАЦИИ»

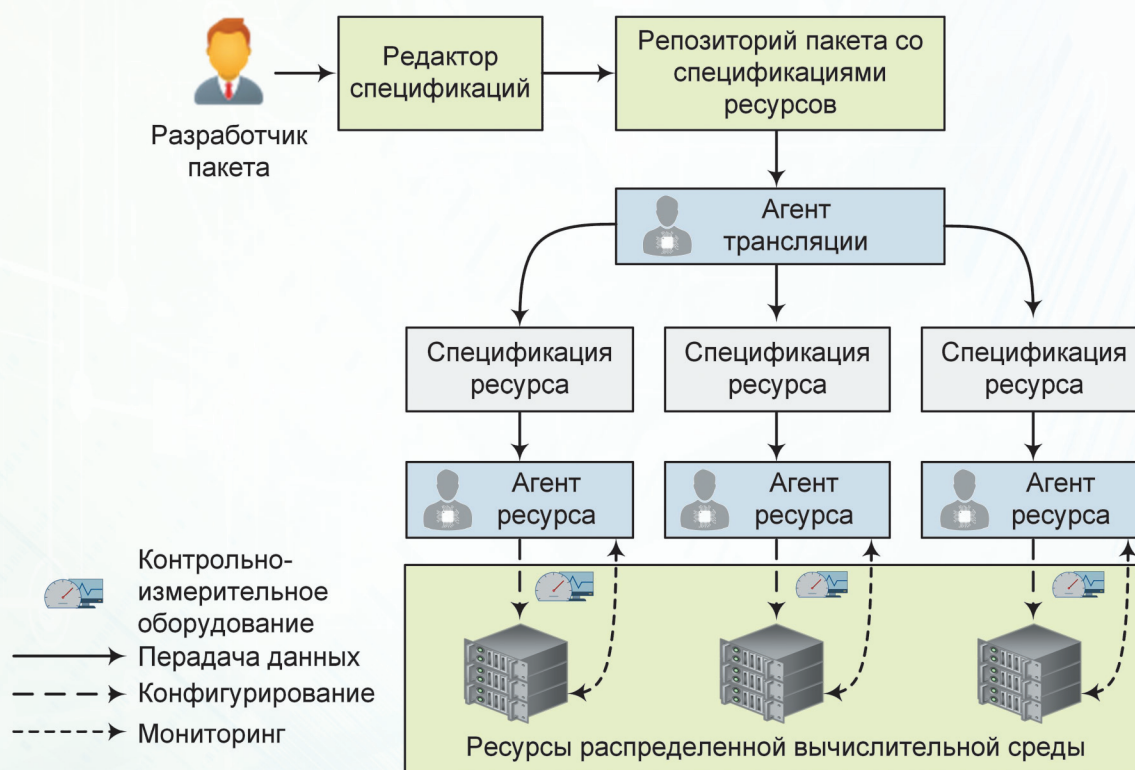


Рис. 3. Схема управления конфигурациями ресурсов

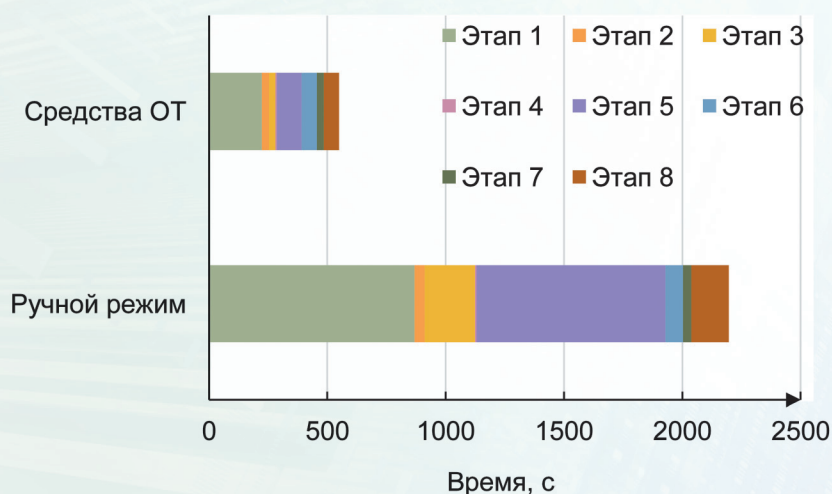
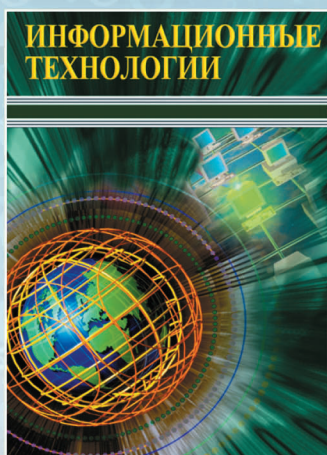


Рис. 4. Время конфигурации ресурсов

Издательство «НОВЫЕ ТЕХНОЛОГИИ» выпускает научно-технические журналы



Ежемесячный теоретический
и прикладной научно-технический журнал

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

В журнале освещаются современное состояние, тенденции и перспективы развития основных направлений в области разработки, производства и применения информационных технологий.

Подписной индекс по Объединенному каталогу
«Пресса России» – 72656



Научно-практический
и учебно-методический журнал

БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ

В журнале освещаются достижения и перспективы в области исследований, обеспечения и совершенствования защиты человека от всех видов опасностей производственной и природной среды, их контроля, мониторинга, предотвращения, ликвидации последствий аварий и катастроф, образования в сфере безопасности жизнедеятельности.

Подписной индекс по
Объединенному каталогу
«Пресса России» – 79963

Междисциплинарный
теоретический и прикладной
научно-технический журнал

НАНО- и МИКРОСИСТЕМНАЯ ТЕХНИКА

В журнале освещаются современное состояние, тенденции и перспективы развития нано- и микросистемной техники, рассматриваются вопросы разработки и внедрения нано микросистем в различные области науки, технологии и производства.



Подписной индекс по
Объединенному каталогу
«Пресса России» – 79493



Ежемесячный теоретический
и прикладной
научно-технический журнал

МЕХАТРОНИКА, АВТОМАТИЗАЦИЯ, УПРАВЛЕНИЕ

В журнале освещаются достижения в области мехатроники, интегрирующей механику, электронику, автоматику и информатику в целях совершенствования технологий производства и создания техники новых поколений. Рассматриваются актуальные проблемы теории и практики автоматического и автоматизированного управления техническими объектами и технологическими процессами в промышленности, энергетике и на транспорте.

Подписной индекс по
Объединенному каталогу
«Пресса России» – 79492

Теоретический
и прикладной
научно-технический журнал

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

В журнале освещаются состояние и тенденции развития основных направлений индустрии программного обеспечения, связанных с проектированием, конструированием, архитектурой, обеспечением качества и сопровождением жизненного цикла программного обеспечения, а также рассматриваются достижения в области создания и эксплуатации прикладных программно-информационных систем во всех областях человеческой деятельности.



Подписной индекс по
Объединенному каталогу
«Пресса России» – 22765

Адрес редакции журналов для авторов и подписчиков:
107076, Москва, Матросская тишина, д. 23, стр. 2, оф. 45. Издательство "НОВЫЕ ТЕХНОЛОГИИ".
Тел.: +7 (499) 270-16-52. E-mail: antonov@novtex.ru