

*Intellectual Technologies  
on Transport  
No 1*



*Интеллектуальные технологии  
на транспорте  
№ 1*

*Санкт-Петербург  
St. Petersburg  
2022*

**Интеллектуальные технологии на транспорте  
(сетевой электронный научный журнал)**

**№ 1, 2022**

ISSN 2413-2527

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

Журнал основан в 2015 году.

---

**Учредитель и издатель**

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

---

**Главный редактор**

Хомоненко А. Д., д.т.н., проф., С.-Петербург, РФ

**Сопредседатели редакционного совета**

Панычев А. Ю., ректор ПГУПС, С.-Петербург, РФ

Чаркин Е. И., зам. ген. директора по ИТ ОАО «РЖД», Москва, РФ

---

**Редакционный совет**

Ададулов С. Е., проф., Москва, РФ

Дудин А. Н., д.т.н., проф., БГУ, Минск, Беларусь

Корниенко А. А., проф., ПГУПС, С.-Петербург, РФ

Ковалец П., проф., Техн. ун-т, Варшава, Польша

Меркурьев Ю. А., проф., РТУ, Рига, Латвия

Нестеров В. М., проф., СПбГУ, С.-Петербург, РФ

Пустарнаков В. Ф., зам. ген. дир. «Газинформсервис»,  
С.-Петербург, РФ

Титова Т. С., проф., проректор ПГУПС,

С.-Петербург, РФ

Федоров А. Р., ген. дир. «Digital Design», С.-Петербург, РФ

Юсупов Р. М., проф., чл.-корр. РАН, С.-Петербург, РФ

---

**Редакционная коллегия**

Бубнов В. П., проф., С.-Петербург, РФ –  
заместитель главного редактора

Александрова Е. Б., проф., С.-Петербург, РФ

Атилла Элчи, проф., ун-т Аксарай, Турция

Басыров А. Г., проф., С.-Петербург, РФ

Безродный Б. Ф., проф., Москва, РФ

Благовещенская Е. А., проф., С.-Петербург, РФ

Булавский П. Е., д.т.н., доц., С.-Петербург, РФ

Василенко М. Н., проф., С.-Петербург, РФ

Глухов А. П., д.т.н., Москва, РФ

Гуда А. Н., проф., Ростов-на-Дону, РФ

Железняк В. К., проф., Новополоцк, Беларусь

Заборовский В. С., проф., С.-Петербург, РФ

Канаев А. К., проф., С.-Петербург, РФ

Котенко А. Г., д.т.н., доц., С.-Петербург, РФ

Куренков П. В., проф., Москва, РФ

Лецкий Э. К., проф., Москва, РФ

Макаренко С. И., д.т.н., доц., С.-Петербург, РФ

Мирзоев Т. А., асс. проф., Джорджия, США

Наседкин О. А., к.т.н., доц., С.-Петербург, РФ

Никитин А. Б., проф., С.-Петербург, РФ

Новиков Е. А., д.т.н., доц., С.-Петербург, РФ

Охтилев М. Ю., проф., С.-Петербург, РФ

Привалов А. А., проф., С.-Петербург, РФ

Соколов Б. В., проф., С.-Петербург, РФ

Таранцев А. А., проф., С.-Петербург, РФ

Утепбергенов И. Т., проф., Алматы, Казахстан

Филипченко С. А., к.т.н., доц., Москва, РФ

Фозилов Ш. Х., проф., Ташкент, Узбекистан

Хабаров В. И., проф., Новосибирск, РФ

Ходаковский В. А., проф., С.-Петербург, РФ

Ху Фу-Ниан, проф., Сяйчжоу, Китай

Чехонин К. А., проф., Хабаровск, РФ

Ялышев Ю. И., проф., Екатеринбург, РФ

---

**Адрес редакции:**

190031, Санкт-Петербург, Московский пр., 9

e-mail: itt-pgups@yandex.ru

Телефон: +7 (812) 457-86-06

---

Сетевое издание «Интеллектуальные технологии на транспорте (сетевой электронный научный журнал),  
Intellectual Technologies on Transport» зарегистрировано Федеральной службой по надзору в сфере связи,  
информационных технологий и массовых коммуникаций.  
Свидетельство Эл № ФС77–61707 от 07 мая 2015 г.

Журнал зарегистрирован в Российском индексе научного цитирования (РИНЦ).

Периодичность выхода – 4 номера в год. Выпуски журнала доступны на сайте <http://itt-pgups.ru>.

Информация предназначена для детей старше 12 лет.

© Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Петербургский государственный университет путей сообщения Императора Александра I», 2022

# Intellectual Technologies on Transport

## Issue 1, 2022

ISSN 2413-2527

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

Founded in 2015.

---

### Founder and Publisher

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

---

### Editor-in-Chief

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

### Co-chairs of the Editorial Council

Panychev A. Y., rector of PSTU, St. Petersburg, Russia  
Charkin E. I., CIO of JSC «Russian Railways», Moscow, Russia

---

### Editorial Council Members

Adadurov S. E., Prof., Moscow, Russia  
Dudin A. N., Prof., BSU, Minsk, Belarus  
Kornienko A. A., Prof., PSTU, St. Petersburg, Russia  
Kovalets P., Prof., Tech. University, Warsaw, Poland  
Merkuryev Y. A., Prof., RTU, Academician  
of the Latvian Academy of Sciences, Riga, Latvia  
Nesterov V. M., Prof., SPbSU, St. Petersburg, Russia

Pustarnakov V. F., Deputy CEO at «Gazinformservice» Ltd.,  
St. Petersburg, Russia  
Titova T. S., Prof., Vice-Rector, PSTU, St. Petersburg, Russia  
Fedorov A. R., CEO at «Digital Design» Ltd.,  
St. Petersburg, Russia  
Yusupov R. M., Prof., Corr. Member of RAS,  
St. Petersburg, Russia

---

### Editorial Board Members

Bubnov V. P., Prof., St. Petersburg, Russia –  
Deputy Editor-in-Chief  
Aleksandrova E. B., Prof., St. Petersburg, Russia  
Atilla Elci, Prof., Aksaray University, Turkey  
Basyrov A. G., Prof., St. Petersburg, Russia  
Bezrodny B. F., Prof., Moscow, Russia  
Blagoveshchenskaya E. A., Prof., St. Petersburg, Russia  
Bulavsky P. E., Dr. Sc., As. Prof., St. Petersburg, Russia  
Vasilenko M. N., Prof., St. Petersburg, Russia  
Glukhov A. P., Dr. Sc., St. Petersburg, Russia  
Guda A. N., Prof., Rostov-on-Don, Russia  
Zheleznyak V. K., Prof., Novopolotsk, Belarus  
Zaborovsky V. S., Prof., St. Petersburg, Russia  
Kanaev A. K., Prof., St. Petersburg, Russia  
Kotenko A. G., Dr. Sc., As. Prof., St. Petersburg, Russia  
Kurenkov P. V., Prof., Moscow, Russia  
Letsky E. K., Prof., Moscow, Russia

Makarenko S. I., Dr. Sc., As. Prof., St. Petersburg, Russia  
Mirzoev T. A., As. Prof., Georgia, USA  
Nasedkin O. A., As. Prof., St. Petersburg, Russia  
Nikitin A. B., Prof., St. Petersburg, Russia  
Novikov E. A., Dr. Sc., As. Prof., St. Petersburg, Russia  
Okhtilev M. Y., Prof., St. Petersburg, Russia  
Privalov A. A., Prof., St. Petersburg, Russia  
Sokolov B. V., Prof., St. Petersburg, Russia  
Tarantsev A. A., Prof., St. Petersburg, Russia  
Utepbergenov I. T., Prof., Almaty, Kazakhstan  
Filipchenko S. A., As. Prof., Moscow, Russia  
Fozilov Sh. Kh., Prof., Tashkent, Uzbekistan  
Khabarov V. I., Prof., Novosibirsk, Russia  
Khodakovskiy V. A., Prof., St. Petersburg, Russia  
Hu Fu-Nian, Prof., Xuzhou, China  
Chekhonin K. A., Prof., Khabarovsk, Russia  
Yalyshev Y. I., Prof., Ekaterinburg, Russia

---

### Editorial address:

190031, St. Petersburg, Moskovsky ave., 9  
e-mail: itt-pgups@yandex.ru  
Phone: +7 812 457 86 06

---

The online journal «Intellectual Technologies on Transport» is registered by the Federal Service for Supervision of Communications, Information Technologies, and Mass Media.

EI No. FS77-61707 Testimony from May 7, 2015.

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

Frequency of release - 4 issues per year. Issues of the magazine are available at <http://itt-pgups.ru>

The content is for children over the age of 12.

## Содержание

<i>Моисеев В. И., Ксенофонтова В. А., Комарова Т. А.</i> Математические модели и численные методы тепло- и массопереноса при естественной конвекции горячих жидких нефтепродуктов в котле вагона-цистерны .....	5
<i>Гордина А. Т., Забродин А. В.</i> Особенности технологий бессерверных вычислений .....	16
<i>Переводы докладов, представленных на международном семинаре «Модели информационных систем на транспорте и методы их решения» на базе кафедр «Информационные и вычислительные системы» и «Высшая математика». Санкт-Петербург, Россия. 09–10 декабря 2021 г.</i>	
<i>Варфоломеев В. А., Лецкий Э. К.</i> Оптимальное планирование использования вагонного парка операторской компании .....	24
<i>Басыров А. Г., Калюжный А. В., Хомоненко А. Д.</i> Алгоритм двуххранцевой упаковки с ограниченным межхранцевым обменом .....	29
<i>Проурзин В. А., Проурзин О. В.</i> Методы анализа больших данных надежности восстанавливаемых систем .....	34
<i>Лохвицкий В. А., Гончаренко В. А., Левчик Э. С.</i> Модель масштабируемого микросервиса на основе системы массового обслуживания с «охлаждением» .....	39
<i>Кустов В. Н., Грохотов А. И., Головкин Е. В.</i> Программная модель маскировки скрытого сообщения в задачах стеганографии .....	45
<i>Русанова Е. В., Рунёв Е. В.</i> Использование моделей принятия решений для оптимизации выбора технологий производства геоэкозащитных материалов .....	58

## Contents

*Moiseev V. I., Ksenofontova V. A., Komarova T. A.*  
Mathematical Models and Numerical Methods of Heat and Mass Transfer During Natural Convection  
of Hot Liquid Petroleum Products in Boiler of Tank Wagon ..... 5

*Gordina A. T., Zabrodin A. V.*  
Characteristics of Serverless Computing Technologies ..... 16

*Translations of reports presented  
at the Models of Information Systems in Transport and Methods for Their Solution Workshop  
on the basis of the departments «Information and Computer Systems» and «Higher Mathematics».  
Saint Petersburg, Russian Federation, December 09–10, 2021.*

*Varfolomeev V. A., Letsky E. K.*  
Optimal Planning of Operator Company Wagons Usage ..... 24

*Basyrov A. G., Kalyuzhnyi A. V., Khomonenko A. D.*  
The Double-Packing Algorithm with Limited Inter-Package Exchange ..... 29

*Prouzrin V. A., Prouzrin O. V.*  
Methods for Big Data Analysis of Reliability of Recoverable Systems ..... 34

*Lokhvitsky V. A., Goncharenko V. A., Levchik E. S.*  
A Scalable Microservice Model Based on a Queuing System with «Cooling» ..... 39

*Kustov V. N., Grokhotov A. I., Golovkov E. V.*  
A Software Model of Masking Hidden Message in Steganography Tasks ..... 45

*Rusanova E. V., Runev E. V.*  
The Use of Decision-Making Models to Optimize the Choice of Technologies  
for the Production of Geoeco-Protective Materials ..... 58

# Математические модели и численные методы тепло- и массопереноса при естественной конвекции горячих жидких нефтепродуктов в котле вагона-цистерны

д.т.н. В. И. Моисеев, В. А. Ксенофонтова  
Петербургский университет путей сообщения  
Императора Александра I  
Санкт-Петербург, Россия  
moiseev\_v\_i@list.ru, koc-vera@yandex.ru

к.т.н. Т. А. Комарова  
Научно-внедренческий центр «Дисперсные системы»  
Санкт-Петербург, Россия  
komarova\_tanusha@mail.ru

**Аннотация.** В работе рассмотрена задача охлаждения застывающих нефтепродуктов, находящихся в стратифицированном состоянии в процессе их перевозки в котле вагона-цистерны при низких температурах воздуха. Предложен новый подход к решению задачи: нефтегруз рассматривается как бинарная система без четкой границы между фракциями. Основная цель — математическое описание тепло- и массопереноса в горячем застывающем продукте. Численное решение задачи проводилось с применением метода сеток.

**Ключевые слова:** вязкие нефтепродукты, железнодорожные перевозки, математическая модель, теплоперенос, массоперенос, численные методы, стратифицированное состояние жидкого нефтепродукта.

## ВВЕДЕНИЕ

Численное решение любого вида тепловых задач включает два этапа — построение конечномерной модели рассматриваемой среды, которая представляет в общем виде некоторую систему алгебраических уравнений, и получение решения этой системы. Исходной для построения численного решения задачи охлаждения жидких нефтепродуктов в котле вагона-цистерны является классическая система двумерных уравнений Навье — Стокса и Фурье — Кирхгофа для тепло- и массопереноса при естественной конвекции в приближении Буссинеска [1].

В гидродинамике модель Д. В. Буссинеска используется при описании потоков, вызываемых термогравитационной конвекцией (ТГК). В ее рамках инерционные силы считаются пренебрежимо малыми, а массовые силы (тяжести), напротив, доминирующими.

## ФИЗИЧЕСКОЕ ОБОСНОВАНИЕ МОДЕЛИ РАССМАТРИВАЕМОГО ОБЪЕКТА И ПРОЦЕССОВ ЕГО ОХЛАЖДЕНИЯ

Жидкие нефтепродукты имеют большой коэффициент объемного теплового расширения  $\beta_T \approx 10^{-3} \text{ } ^\circ\text{C}^{-1}$  (у мазутов почти в пять раз больший, чем у воды и лишь в четыре раза меньший, чем у газов), поэтому удельный вес двух участков жидкости с различной температурой заметно отличается. Это температурное изменение плотности жидкости считается главным фактором в приближении Буссинеска, тогда как температурные изменения ее других характеристик исключаются из рассмотрения. В модели также считается, что изменение знака перепада температур  $\theta = T_1 - T_2$  между двумя точками в объеме жидкости на противоположное

вызывает изменение направления ее течения при конвекции. В задаче охлаждения нефтепродуктов при перевозках и это требование выполняется, так как котел вагона-цистерны является крупногабаритным горизонтально установленным цилиндрически симметричным сосудом.

Все нефтепродукты являются сложными углеводородными соединениями, состав которых определяется характеристиками сырой нефти и режимами крекинг-процесса — основного способа ее переработки [2]. Их основу составляют полициклические соединения, включающие  $n$  атомов углерода: арены  $\text{C}_n\text{H}_{2n+2}$  и нафтены  $\text{C}_n\text{H}_{2n-4}$  (у дизельных топлив  $n = 13 \dots 18$ , у мазутов  $n = 18 \dots 25$ ). В состав мазутов дополнительно входят асфальтены и нефтяные смолы, содержание которых в сырой нефти составляет от 3 до 20 %. Молекулярная масса асфальтенов и смол колеблется от 1 000 до 5 000 а.е.м. (атомных единиц массы). Асфальтены склонны вначале к ассоциации, а затем и слипанию, при котором образуются твердые частицы размерами до десятков микрон. Частицы имеют плотность  $\rho_{\text{ТВ}} \approx 1\,100 \text{ кг/м}^3$  и образуют взвесь в более жидкой фракции с плотностью  $\rho \approx 850 \text{ кг/м}^3$ . С понижением температуры ниже  $15 \text{ } ^\circ\text{C}$  частицы очень медленно оседают из взвеси на нижнюю часть котла вагона-цистерны, образуя на ней высоковязкий застывший слой, делающий невозможным слив нефтепродукта без разогрева для восстановления текучести.

Аналогичная картина наблюдается при охлаждении летних дизельных топлив [3]. При температуре ниже минус  $6 \text{ } ^\circ\text{C}$  в них начинают образовываться и выпадать в осадок твердые микрокристаллы  $n$ -парафинов, содержание которых в дизельном топливе также может достигать до 20 %. Фазовое расслоение летних дизельных топлив является необратимой физико-химической реакцией, нарушающей баланс между различными присадками, вводимыми в топливо на стадии его подготовки. Нарушение баланса резко ухудшает эксплуатационные характеристики топлива, вплоть до потери его пригодности.

Технические жидкости, изменяющие свои физические или эксплуатационные характеристики при охлаждении, называют застывающими, поэтому нефти, мазуты и летние дизельные топлива обобщаются термином «застывающие нефтепродукты» (ЗНП).

Значения коэффициентов теплопроводности, удельной теплоемкости, плотности и температуропроводности основных видов ЗНП показаны в таблице 1.

Таблица 1

Теплофизические коэффициенты основных углеводородных топлив при температуре +20 °С

	Теплопроводность $\lambda$ , Вт/(м·°С)	Теплоемкость $C_p$ , Дж/(кг·°С)	Плотность $\rho$ , кг/м <sup>3</sup>	Температуропроводность $a$ , 10 <sup>-7</sup> м <sup>2</sup> /с
Дизельные топлива	0,146	2 104	800	0,869
Мазут флотский Ф12	0,130	1 880	950	0,728
Мазут котельный М100	0,105	1 860	980	0,576
Крекинг-остатки М200	0,102	1 850	1 010	0,546

Как видно из таблицы, названные величины изменяются с переходом от дизельных топлив облегченного состава к тяжелым остаткам крекинг-процесса, но эти изменения в целом не велики: не более чем в 1,3...1,4 раза. Близость характеристик обусловлена тем, что основу ЗНП составляют одни и те же молекулы аренов и нафтенев.

В рамках физической модели ЗНП можно рассматривать как бинарные системы без четкой границы раздела между фракциями, при этом взвесь из твердых частиц в них рассматривается как примесь с концентрацией  $C$ , изменяющейся с температурой и высотой слоя жидкости. Застывание ЗНП происходит не при фиксированной температуре фазового перехода, а в интервале температур, зависящем от состава входящих в нефтепродукт фракций. Все ЗНП являются несжимаемыми жидкостями и имеют не зависящее от температуры давление, а их плотность меняется с температурой и концентрацией:

$$\rho = \rho(C, T).$$

На рисунке 1, *a* показана картина движения горячей жидкости в котле вагона-цистерны при симметричном охлаждении ее стенок [4]. Слои ЗНП у стенок котла охлаждаются и, с ростом плотности, опускаются вниз, создавая внутри котла восходящие потоки. В средней части котла обычно образуются ядра ( $B$ ) неподвижной жидкости.

Стенка котла омывается горячим продуктом и он быстро охлаждается, при этом наибольшую скорость течения имеет на «вертикальном участке» стенки между прямыми  $C$  и  $C'$ , где вектор скорости потока жидкости совпадает с вектором ускорения свободного падения.

Такой характер циркуляционного движения горячего ЗНП при симметричном охлаждении котла вагона-цистерны идеализирован. В реальности котел охлаждается с одной стороны больше, чем с другой из-за ветра. Но цистерна движется несколько суток и проходит большие расстояния, поэтому изменение направления ветра надо считать равновероятным, а картину, показанную на рисунке 1, *a*, считать усредненной по большому промежутку времени.

Для упрощения математического описания ТГК в горизонтальном цилиндре Ниманн [5] предложил заменить его цилиндрическую поверхность совокупностью вертикальных, горизонтальных и наклонных пластин, показанных на рисунке 1, *б*.

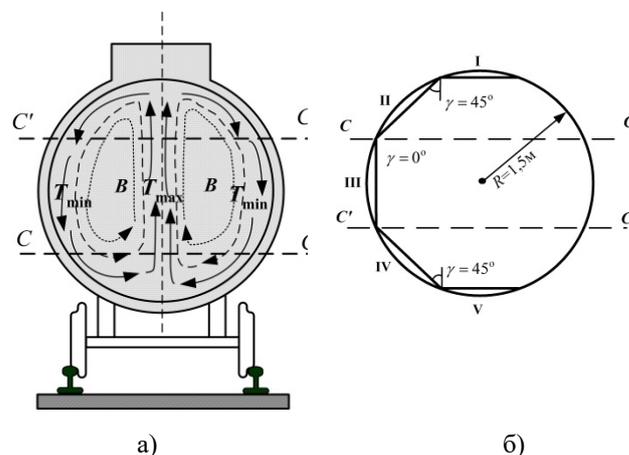


Рис. 1. Модель циркуляционного движения жидкости при естественной конвекции

*a* — схема токов жидкости в котле вагона-цистерны при его симметричном охлаждении; *б* — модель Ниманна для расчета конвекции жидкости внутри горизонтального цилиндра

Обоснованием модели является то, что при малых скоростях движения жидкости, в углах, образованных соседними пластинами, образуются застойные зоны, сглаживающие поле токов жидкости. Достоинством модели Ниманна является простота описания конвекции при продольном обтекании пластин, являющихся элементами этого «расчетного канала», с применением хорошо проработанной теории ламинарного пограничного слоя [6].

Котел вагона-цистерны всегда заполняется с некоторым недоливом для учета возможного увеличения объема перевозимой жидкости, обусловленного ее тепловым расширением при нагревании, вызванном, например, солнечной радиацией. Поэтому в заполненном котле имеется плоская свободная поверхность жидкости, показанная на рисунке 1, *б* участком I, над которой находится слой воздуха.

Участок II моделирует верхнюю часть обечайки котла вагона-цистерны, участок III — ее «вертикальную» часть, участок IV — нижнюю часть котла. Горизонтальная нижняя плоскость V моделирует самую нижнюю часть обечайки котла. Массовой силой, вызывающей ТГК, является вертикальная составляющая силы тяжести, действующая на единичную массу жидкости, равная  $g \cos \gamma$ , где  $\gamma$  — угол между плоскостью пластины и вертикалью. На участках II и IV  $\gamma = 45^\circ$ , на участке III, где  $\gamma = 0^\circ$ , скорость течения максимальная. На горизонтальных верхнем I и нижнем V участках, где  $\gamma = 90^\circ$ , конвекция отсутствует. На участке I — потому, что над ним находится теплоизолирующий слой воздуха, а на участке V — потому, что там накапливается жидкость с наименьшей температурой и, следовательно, с наибольшей плотностью.

Для математического моделирования процесса охлаждения ЗНП в котле вагона-цистерны можно ограничиться рассмотрением двумерных течений, записанных в декартовой системе координат. Такие упрощения оправданы тем, что при диаметре котла 3 м кривизной его стенок можно пренебречь, и тем, что площадь днищ котла составляет всего 12 % от общей площади его поверхности, при этом они удалены друг от друга на 11 м. Очевидно, что в этих условиях

влияние токов жидкости на днищах на общую картину ТГК внутри котла также можно считать незначительным.

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ  
ПРОЦЕССА ОХЛАЖДЕНИЯ НЕФТЕПРОДУКТОВ  
ПРИ ПЕРЕВОЗКАХ В ВАГОНАХ-ЦИСТЕРНАХ

Примем, что у жидкого ЗНП при охлаждении возникают малые изменения плотности  $\rho$ , поверхностного натяжения  $\sigma$  и концентрации примеси  $C$ :

$$\rho = \rho_0 + \rho'; \quad C = C_0 + C'; \quad \sigma = \sigma_0 + \sigma'; \quad T = T_0 + T'$$

при  $p = p_0 = \text{const}$ .

Здесь  $p_0, C_0, \rho_0, \sigma_0$  и  $T_0$  — значения плотности, концентрации, давления, поверхностного натяжения и температуры, при которых конвекции нет и выполняются условия гидростатики, а  $\rho', C', \sigma'$  и  $T'$  — малые отклонения названных параметров от статических значений. Изменения первых двух параметров обуславливают свободную конвекцию горячего ЗНП. Малость отклонений подразумевает выполнение неравенств:

$$\rho' \ll \rho_0; \quad C' \ll C_0; \quad \sigma' \ll \sigma_0; \quad T' \ll T_0.$$

Изменение давления в столбе бинарной жидкости связано с распределением по высоте ее температуры и концентрации примеси:

$$\begin{aligned} \frac{1}{\rho} \frac{\partial p}{\partial y} &= \frac{1}{\rho_0} \frac{\partial p_0}{\partial y} + \frac{1}{\rho_0} \frac{\partial p'}{\partial y} - \frac{1}{\rho_0^2} \frac{\partial p_0}{\partial y} \rho' = \\ &= -g + \beta_T g T' + \beta_C g C' + \frac{1}{\rho_0} \frac{\partial p'}{\partial y}. \end{aligned}$$

Здесь  $\beta_T$  и  $\beta_C$  — термические коэффициенты теплового и концентрационного изменения плотности:

$$\beta_T = -\frac{1}{\rho} \left( \frac{\partial \rho}{\partial T} \right)_{p, C}; \quad \beta_C = -\frac{1}{\rho} \left( \frac{\partial \rho}{\partial C} \right)_{p, T}.$$

Дифференциальные уравнения движения, переноса теплоты, массы и концентрации в горячем ЗНП образуют систему, в которой скорость течения, температура и концентрация примеси зависят от пространственных координат и времени  $t$ . В векторном виде уравнения движения и тепло-массопереноса записываются так:

$$\frac{\partial \vec{V}}{\partial t} + (\vec{V} \times \nabla) \vec{V} = -\frac{1}{\rho} \nabla \rho' + \nu \Delta \vec{V} + \frac{1}{\rho} \vec{F}; \quad (1)$$

$$\nabla \times \vec{V} = 0; \quad (2)$$

$$\left[ \frac{\partial T'}{\partial t} + (\vec{V} \times \nabla) T' \right] = a \Delta T' + \frac{q_v}{C_p \rho}; \quad (3)$$

$$\left[ \frac{\partial C'}{\partial t} + (\vec{V} \times \nabla) C' \right] = D \Delta C'. \quad (4)$$

В равенстве (1) под  $\vec{F}$  понимается сумма всех сил, действующих на рассматриваемый бесконечно малый объем жидкости, который часто называют частицей жидкости. Когда жидкость несжимаема и течение происходит с малой скоростью, то этими силами являются архимедова, трения и тяжести. Также в уравнениях (1)–(4) использованы два

оператора, применяемые в векторном исчислении (Гамильтона и Лапласа):

$$\nabla = \frac{\partial}{\partial x} \vec{i} + \frac{\partial}{\partial y} \vec{j} + \frac{\partial}{\partial z} \vec{k}; \quad \Delta = \nabla \times \nabla = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}.$$

В уравнениях (1), (3) и (4) фигурируют коэффициенты температуропроводности  $a$ , диффузии  $D$ , кинематическая вязкость жидкости  $\nu$  (все три имеют одинаковую размерность — м<sup>2</sup>/с) плотность нефтепродукта  $\rho$ , кг/м<sup>3</sup>, а также объемная плотность тепловыделения  $q_v$ , Вт/м<sup>3</sup>. Коэффициент температуропроводности  $a$  выражается через коэффициенты ее теплопроводности  $\lambda$ , Вт/(м·°C), удельной теплоемкости  $C_p$ , Дж/(кг·°C) и плотности  $\rho$ :

$$a = \frac{\lambda}{C_p \rho}. \quad (5)$$

Коэффициент диффузии  $D$  частиц взвеси зависит от ее концентрации  $C$  и коэффициента активности  $\xi$  — величины, различной для разных систем, определяемой экспериментально:

$$D = D^0 \left( 1 + \frac{d(\ln \xi)}{d(\ln C)} \right),$$

где  $D^0$  — коэффициент диффузии для бесконечно разбавленного раствора (для ЗНП — при температуре, превышающей температуру фазового расслоения). При температуре +20 °C коэффициент диффузии имеет порядок  $D \approx 10^{-1} \dots 10^{-2}$  м<sup>2</sup>/с.

Объемная плотность тепловыделения  $q_v$  вводится, если в задаче учитываются такие эффекты, как разогрев нефтегруза при сливе или теплота фазового перехода при плавлении и затвердевании парафинов, содержащихся в ЗНП.

Вводим в рассмотрение  $u$  и  $v$  — составляющие вектора скорости  $\vec{V}$  движущейся жидкости, причем примем, что составляющая  $u$  направлена вдоль стенки котла по направлению течения жидкости, а составляющая  $v$  — по нормали к стенке, т. е. к оси котла. Векторные уравнения (1) и (2) в скалярной форме образуют систему из трех уравнений:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + f_x; \quad (6)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) + f_y; \quad (7)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \quad (8)$$

В равенствах (6) и (7) записаны составляющие ускорения  $f_x$  и  $f_y$ , которое имеет движущаяся частица жидкости при ТГК. Система дифференциальных уравнений (6)–(8) рассматривается при граничных и начальных условиях. Граничные условия на твердой стенке котла радиусом  $R$ , часто называемые условиями прилипания, выражаются равенством

$$u|_{r=R} = v|_{r=R} = 0,$$

где  $r$  — расстояние от оси котла до данной точки в массе нефтепродукта:

$$r = \sqrt{x^2 + y^2}.$$

Другим условием будет конечность температуры ЗНП на оси котла:

$$T|_{r=0} \neq \infty.$$

Начальными условиями определяются распределения скорости и давления в начальный момент времени  $t = 0$ :

$$u(x, y, 0) = u^0; \quad v(x, y, 0) = v^0; \quad p(x, y, 0) = \text{const}.$$

Уравнения (6)–(8) часто записывают, используя функцию тока  $\Psi$  и вихрь скорости  $\omega$ , причем в безразмерной форме, более удобной для численной реализации. Функция тока  $\Psi$  и вихрь скорости  $\omega$  определяются равенствами:

$$u = \frac{\partial \Psi}{\partial y}; \quad v = -\frac{\partial \Psi}{\partial x}; \quad \omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}. \quad (9)$$

Вихрь скорости частицы жидкости определяется вектором угловой скорости  $\vec{\omega}$  и находится как векторное произведение оператора  $\Delta$  на вектор скорости  $\vec{V}$ :

$$\vec{\omega} = \frac{1}{2} \text{rot } \vec{V} = \frac{1}{2} \nabla \times \vec{V}.$$

Когда круговое движение происходит в плоскости, параллельной площадке на стенке, вихрь имеет только одну проекцию  $z$  и угловая скорость вращения будет:

$$\omega_z = \frac{1}{2} \text{rot}_z \vec{V} = \frac{1}{2} \left( \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right).$$

Касательная к линии тока жидкости  $\Psi = \text{const}$  определяет направление вектора скорости жидкости в данной точке. При медленном течении жидкости линии тока жидкости совпадают с траекторией движения частиц взвеси. Граничные условия на неподвижной твердой стенке определяются соотношениями:

$$\psi = 0; \quad \frac{\partial \psi}{\partial n} = 0.$$

Начальные условия задаются так:

$$\psi(x, y, 0) = \psi^0(x, y); \quad \omega(x, y, 0) = \omega^0(x, y).$$

В декартовых координатах и в безразмерной форме система уравнений (6)–(8) записывается таким образом:

$$\frac{\partial \omega}{\partial t} + u \frac{\partial \omega}{\partial x} + v \frac{\partial \omega}{\partial y} = \frac{1}{\text{Re}} \left( \frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} \right); \quad (10)$$

$$\frac{\partial^2 \Psi}{\partial x^2} + \frac{\partial^2 \Psi}{\partial y^2} = 0.$$

Здесь фигурируют следующие величины: масштабный параметр  $L$ , критерий Рейнольдса  $\text{Re}$  и составляющие  $F_x$  и  $F_y$ . За масштабный параметр взят диаметр котла вагона-цистерны  $L = 2R$ :

$$F = \frac{\partial F_x}{\partial y} - \frac{\partial F_y}{\partial x}; \quad F_x = \frac{f_x L}{V^2}; \quad F_y = \frac{f_y L}{V^2}.$$

Подстановка (9) в (10) приводит равенство к виду:

$$\frac{\partial \omega}{\partial t} + \frac{\partial \Psi}{\partial y} \frac{\partial \omega}{\partial x} - \frac{\partial \Psi}{\partial x} \frac{\partial \omega}{\partial y} = \frac{1}{\text{Re}} \left( \frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} \right) + F. \quad (11)$$

Поверхностное натяжение жидкости  $\sigma$  зависит от ее температуры и концентрации примеси, поэтому на участке I свободной поверхности (рис. 1, б) выполняется

$$\sigma = \sigma_0 (\beta_{\sigma T} T' + \beta_{\sigma C} C').$$

Это требование учитывается в граничном условии на свободной поверхности жидкости:

$$u = 0; \quad \nu \rho \frac{\partial v}{\partial n} = \frac{\partial \sigma}{\partial l} = \sigma_0 \left( \beta_{\sigma T} \frac{\partial T'}{\partial l} + \beta_{\sigma C} \frac{\partial C'}{\partial l} \right),$$

где  $\nu$  и  $\rho$  — кинематическая вязкость и плотность жидкости соответственно,  $n$  — нормаль, а  $l$  — касательная к поверхности жидкости.

Среди граничных условий к уравнениям (2) и (3) можно использовать закон Фурье, определяющий закон теплообмена с окружающей средой на наружной поверхности котла вагона-цистерны:

$$q = \alpha_{\text{конв}} (T_{\text{ст}} - T_{\text{ср}}), \quad (12)$$

где  $q$  — плотность теплового потока от поверхности котла вагона-цистерны в окружающее пространство;  $\alpha_{\text{конв}}$  — коэффициент внешней конвективной теплоотдачи;  $T_{\text{ст}}$  и  $T_{\text{ср}}$  — температура стенки котла вагона-цистерны и окружающей среды соответственно. Диффузионный поток в окружающее пространство отсутствует.

Условие (12) оправдано тем, что стальная стенка котла имеет высокую теплопроводность ( $\lambda_{\text{ст}} = 4 \text{ Вт/(м}^\circ\text{C)}$ ) и малую толщину ( $\sigma_{\text{ст}} = 0,01 \text{ м}$ ), поэтому ее термическое сопротивление теплового потоку

$$\delta_{\text{ст}}/\lambda_{\text{ст}} = 2,27 \times 10^{-4} (\text{м}^2 \times ^\circ\text{C})/\text{Вт}$$

является величиной высшего порядка малости.

В моделировании ТГК используются избыточная температура жидкости  $\theta$  относительно окружающей среды  $\theta = T - T_{\text{ср}}$ , а также безразмерные критерии теплового и диффузионного подобия, отмеченные в таблице 2, и параметр  $T_N$ , являющийся безразмерным временем процесса.

Таблица 2

Основные критерии теплового, динамического и диффузионного подобия, применяемые при моделировании свободной конвекции

Критерий подобия физического процесса	Перенос теплоты и импульса (гидродинамика)	Перенос массы вещества (диффузия)
Рейнольдса	$\text{Re} = VL/\nu$	
Грасгофа	$r = \frac{g\beta_m L^3 \Delta T}{\nu^2}$	$r_D = \frac{g\beta_D L^3 \Delta C}{\nu^2}$
Прандтля	$\text{Pr} = \nu/\alpha$	$\text{Pr}_D = \nu/D$
Марангони	$\text{Mn} = \frac{\sigma_0 \beta_{\sigma T} L \Delta T}{\rho \nu \alpha}$	$\text{Mn}_C = \frac{\sigma_0 \beta_{\sigma C} L \Delta C}{\rho \nu D}$
Фурье	$\text{Fo} = at/L^2$	$\text{Fo}_D = Dt/L^2$
Релея	$\text{Ra} = r \times \text{Pr} = g\beta_E \Delta T \frac{L^3}{\nu \alpha}$	$\text{Ra}_D = Gr_D \times \text{Pr}_D = g\beta_C \Delta C \frac{L^3}{\nu D}$
Ричардсона	$\text{Ri} = \frac{gL}{\nu^2} = \frac{r}{\text{Re}^2 \beta_T \Delta T}$	$\text{Ri}_D = \frac{gL}{\nu^2} = \frac{r_D}{\text{Re}^2 \beta_C \Delta C}$

Исходя из принципов физического моделирования гидродинамических и тепловых процессов систему дифференциальных уравнений (1)–(4) сводят к безразмерному виду:

$$\frac{\partial \vec{V}}{\partial t} + (\vec{V} \times \nabla) \vec{V} = -\nabla p + \frac{1}{Re} \Delta \vec{V} + \vec{n} \times \left( \frac{r}{Re^2} \theta + \frac{r_c}{Re^2} C \right); \quad (13)$$

$$\nabla \times \vec{V} = 0; \quad (14)$$

$$\rho c_p \left[ \frac{\partial \theta}{\partial t} + (\vec{V} \times \nabla) \theta \right] = \frac{1}{RePr} \Delta \theta; \quad (15)$$

$$\left[ \frac{\partial C}{\partial t} + (\vec{V} \times \nabla) C \right] = \frac{1}{RePr_D} \Delta C. \quad (16)$$

Безразмерным аналогом граничного условия (11) является равенство

$$\frac{\partial u}{\partial n} = \frac{Mn}{RePr} \frac{\partial \theta}{\partial l} + \frac{Mn_c}{RePr_D} \frac{\partial C}{\partial l}.$$

Система дифференциальных уравнений (13)–(16) описывает два вида естественной конвекции — тепловую и концентрационную. Интенсивность термогравитационной конвекции определяется числами Грасгофа  $Gr$  и Прандтля  $Pr$ , а концентрационной конвекции — их аналогами  $Gr_D$  и  $Pr_D$ . Числа Марангони  $Mn$  и  $Mn_c$  определяют интенсивность поверхностных механизмов конвекции.

Наиболее простыми частными случаями являются режимы переноса теплоты и массы молекулярными процессами теплопроводности и диффузии в неподвижной жидкости, которые реализуются при  $Gr = 0$ ,  $Gr_c = 0$ ,  $Mn = 0$ ,  $Mn_c = 0$ .

При этом уравнения (15) и (16) описывают перенос теплоты и массы движущейся жидкостью, в которой тепло- и массообмен не влияют на движение.

Решение системы уравнений (13)–(16), определяющее распределение температур, скорости циркуляционного движения жидкости и концентрации примеси определяется критериальной зависимостью вида

$$\Phi = \Phi \left( \frac{x}{L}, \frac{y}{L}, Fo, r, r_D, Pr, Pr_D, Mn, Mn_c \right). \quad (17)$$

Нахождение функции  $\Phi$  (17) является основной целью схематизации процесса тепло- и массопереноса. Равенство безразмерных критериев подобия, отмеченных в таблице 2, для модельных и натурных условий обеспечивает подобие тепловых и гидродинамических процессов и является необходимым требованием для своего выполнения как в физическом, так и математическом моделировании:

$$r = idem, \quad r_c = idem, \quad Mn = idem, \\ Mn_c = idem, \quad Pr = idem, \quad Pr_D = idem.$$

Уравнение конвективной теплопроводности Фурье — Кирхгофа (3) в общем случае также можно решить только численными методами. Однако имеются весьма удачные попытки упрощения математического описания теплопереноса в жидких средах. Еще в XVIII веке И. Ньютон установил, что перенос теплоты в неподвижной среде, имеющей внутри себя некоторое распределение температур, выражается равенством

$$q = -\lambda \Delta T,$$

где  $q$  — плотность теплового потока;

$\lambda$  — коэффициент теплопроводности среды.

В одномерном случае это уравнение существенно упрощается:

$$q = -\lambda \left( \frac{T_2 - T_1}{\delta} \right), \quad (18)$$

где  $\delta$  — расстояние между двумя точками среды, имеющими температуры  $T_1$  и  $T_2$ , причем  $T_2 > T_1$ .

Для описания теплообмена в прямоугольном канале, имеющем нагретую и холодную стенки и заполненном жидкостью, академик М. А. Михеев использовал формулу (18), в которую вместо  $\lambda$  ввел «эквивалентный коэффициент теплопроводности»  $\lambda_{эkv}$ . Этим коэффициентом он предложил учитывать конвективный теплоперенос между стенками как таковой, без рассмотрения характера движения жидкости. Отношение удельных тепловых потоков в циркулирующей и неподвижной жидкости он назвал коэффициентом конвекции:

$$\frac{q_{цирк.ж}}{q_{неподв.ж}} = \frac{\lambda_{эkv}}{\delta} (T_{ст.1} - T_{ст.2}) = \frac{\lambda_{эkv}}{\lambda_{ж}} = \varepsilon_k.$$

Данную модель принято называть моделью элементарной теплопроводности Михеева.

Если за определяющий размер  $L$  взять расстояние между стенками канала, а за температуру жидкости принять полусумму температур стенок канала

$$\bar{T} = \frac{T_{ст.1} + T_{ст.2}}{2},$$

то коэффициент конвекции однозначно выражается через критерий Релея (табл. 2). Эта однозначность сохраняется независимо от вида жидкостей (воздух, вода, органические жидкости) и формы канала. Она описана эмпирическими соотношениями:

$$\begin{cases} \varepsilon_k^* = 0,105 Ra^{0,3} & \text{при } 10^3 \leq Ra \leq 10^6 \\ \varepsilon_k^* = 0,400 Ra^{0,2} & \text{при } 10^6 \leq Ra \leq 10^{10}. \end{cases} \quad (19)$$

При значениях  $Ra \leq 10^3$   $\varepsilon_k = 1$ , то есть перенос тепла осуществляется молекулярной теплопроводностью, а при  $Ra > 10^{10}$  эквивалентная теплопроводность циркулирующей жидкости в 50...60 раз превышает ее молекулярную теплопроводность ( $\varepsilon_k > 50$ ).

Обобщая данные экспериментов, Ниманн [5] построил формулу для нахождения коэффициента конвекции жидкости в канале, составленном из плоских стенок (рис. 1, б):

$$\varepsilon_k = 1 + \frac{ARa^n}{B + Ra},$$

где  $A$ ,  $B$  и  $n$  — постоянные числа, зависящие от конфигурации циркулирующего слоя и направления теплового потока, их обобщенные значения даны в таблице 3.

Упрощенно процесс конвективного тепло- и массопереноса можно описать так. На участке III, например, вектор скорости имеет составляющую  $V \times \cos 45^\circ$ , и для единичной массы жидкости появляется составляющая импульса  $mV \times \cos 45^\circ$ , направленная перпендикулярно стенке котла к оси котла.

Перенос импульса возбуждает турбулентные движения в более удаленных от стенки котла слоях жидкости, что и обеспечивает конвективный теплоперенос. Коэффициент

эквивалентной теплопроводности изменяется с расстоянием от оси котла  $\lambda_{\text{экр}} = f(r)$ .

Таблица 3

Численные значения постоянных  $A, B$  и  $n$

$N$ п/п	Конфигурация слоя движущейся жидкости	Направление теплового потока	$A$	$B \times 10^{-4}$	$n$
1	Коаксиальный цилиндрический слой	радиальное	0,1190	1,45	1,27
2	Плоскопараллельный слой горизонтальный	снизу вверх	0,0700	0,32	1,33
3	Плоскопараллельный слой вертикальный	горизонтальное	0,0236	1,01	1,39
4	Плоскопараллельный слой, установленный под углом $45^\circ$ к вертикальной оси	снизу вверх	0,0430	0,41	1,36
5		сверху вниз	0,0250	1,30	1,36

Одной из основных целей настоящей работы являлось математическое описание тепло- и массопереноса в горячем ЗНП, находящегося в стратифицированном состоянии на стадии его перевозки в железнодорожных вагонах-цистернах при низких температурах воздуха.

Из (19) следует, что стратифицированному состоянию ЗНП отвечает  $\epsilon_k = 1$  и будет наблюдаться при  $Ra \leq 10^3$ .

При математическом описании течения стратифицированных жидкостей важную роль играет частота плавучести Брента — Вэйсяля, имеющая размерность  $c^{-1}$ :

$$N = \sqrt{-\frac{g}{\rho} \frac{\partial \rho}{\partial z}}. \quad (20)$$

Параметр  $N$  является мерой стабильности жидкости при вертикальных перемещениях, вызванных естественной конвекцией, то есть мерой устойчивости ее стратифицированного состояния. Он характеризует частоту малых вертикальных колебаний выделенной частицы жидкости в статически стабильном окружении ее той же жидкостью.

Окружающая среда мысленно разделяется на большое число слоев, и при этом считается, что элементарный объем, отклоненный по вертикали от своего исходного положения, приобретает вертикальное ускорение, обусловленное возвращающей силой, которую образуют разнонаправленные силы — тяжести и архимедова. Если эта сила возвращает элементарный объем к исходному положению, расслоение считается устойчивым и частота колебаний определяется (20), это будет при  $\partial \rho / \partial z < 0$ .

Если же  $\partial \rho / \partial z > 0$  частота колебаний  $N$  будет чисто мнимым числом, расслоение становится неустойчивым и в жидкости возникнет конвекция. В модели Буссинеска частота плавучести бинарной системы определяется равенством

$$N = \sqrt{g\beta_T \frac{\partial T}{\partial z} + g\beta_C \frac{\partial C}{\partial z}}. \quad (21)$$

Параметр  $N$  определяют безразмерным временем  $T_N$ , выражаемым числами Грасгофа, отмеченными в таблице 2:

$$T_N = \frac{v}{NL^2} = \frac{1}{\sqrt{r} + r_C}. \quad (22)$$

Одним из основных в задаче моделирования ТГК является отмеченное в таблице 2 число Ричардсона  $Ri$ .

Им определяется отношение потенциальной энергии элементарного объема движущейся жидкости к его кинетической энергии. Если  $Ri > 1$ , то архимедова сила, действующая на выделенный элементарный объем, оказывается доминирующей в том смысле, что ТГК не может эффективно перемешать расслоившуюся жидкую среду. При  $Ri < 1$  конвекция возникает самопроизвольно.

#### ЧИСЛЕННАЯ РЕАЛИЗАЦИЯ МАТЕМАТИЧЕСКОЙ МОДЕЛИ

Для численного решения данной системы уравнений одним из наиболее распространенных является метод сеток и особенно две из его разновидностей, называемые методом конечных разностей (МКР) и методом конечных элементов (МКЭ) [7, 8].

Суть метода сеток состоит в следующем. В области изменения независимых переменных вводится дискретная совокупность узловых точек, называемая сеткой. Вместо функций непрерывного аргумента рассматриваются «сеточные» функции, значения которых задаются в узловых точках сетки. Обозначим сеточную функцию через  $\varphi$ . По характеру решаемой задачи она может быть функцией одного, двух или трех аргументов (например, двух пространственных координат и времени), аргументы отмечаются стандартным образом.

По своему смыслу функция  $\varphi$  может выражать составляющие вектора скорости  $u$  и  $v$ , функции тока и вихря  $\Psi_x, \Psi_z, \omega$ , концентрацию примеси  $C$  или избыточную температуру  $\theta$ . Производные, входящие в дифференциальные уравнения, заменяются приближенными конечно-разностными аппроксимациями. Сами же дифференциальные уравнения заменяются приближенными сеточными алгебраическими уравнениями, образующими большую по объему, но в принципе легко решаемую систему.

Одномерные конечные разности строятся так. Для дискретной независимой переменной  $x$  (под которой может пониматься как пространственная координата, так и время) строится множество  $(L+1)$  дискретных равноотстоящих точек  $x_l (l = 0, 1, 2, \dots, L)$ , образующих собственно сетку. При этом

$$0 \leq x \leq L; \quad x_0 = 0, \quad x_l = L; \quad x_{l+1} - x_l = \Delta x. \quad (23)$$

Аппроксимация первых производных производится тремя типами разностей, называемых «разностью вперед», «разностью назад» и «центральной разностью». Эти разности соответственно своим названиям задаются в виде:

$$\begin{aligned} \left. \frac{d\varphi}{dx} \right|_l &\approx \frac{\varphi_{l+1} - \varphi_l}{\Delta x}; \\ \left. \frac{d\varphi}{dx} \right|_l &\approx \frac{\varphi_l - \varphi_{l-1}}{\Delta x}; \\ \left. \frac{d\varphi}{dx} \right|_l &\approx \frac{\varphi_{l+1} - 2\varphi_l + \varphi_{l-1}}{2\Delta x}. \end{aligned} \quad (24)$$





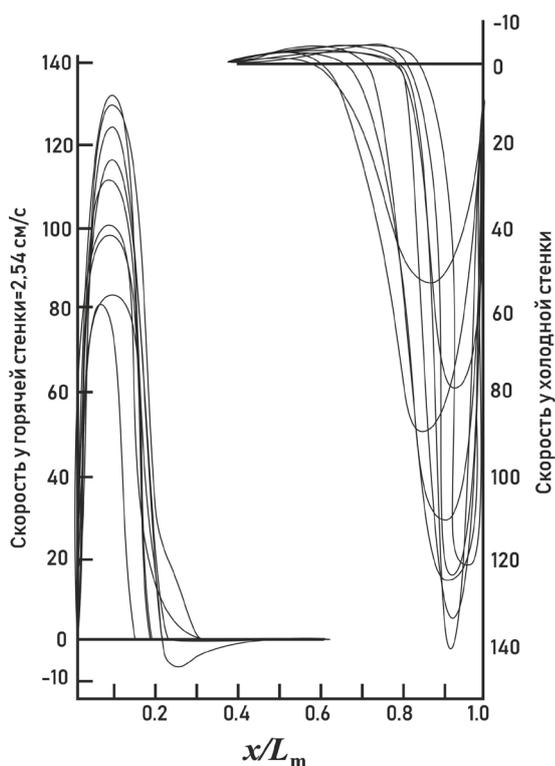


Рис. 3. Распределение скоростей движения жидкостей при естественной конвекции в прямоугольном канале с обогреваемой и охлаждаемой параллельными вертикальными стенками

Вдоль оси абсцисс отложено отношение координаты точки наблюдения к ширине канала. Эксперименты показывают наличие ядра неподвижной жидкости в центральной части канала, подобное ядрам (B), изображенным на рисунке 1, а. Распределение скоростей течения иллюстрируют кривые, усредняющие результаты наблюдений. Исходя из кривых можно предположить, что значение скорости, входящее в число Ричардсона для различных нефтепродуктов, находится в интервале  $V = 0,3 \dots 0,5$  м/с.

Приняв, что характерный размер области  $L = 2R = 3$  м, из таблицы 2 находим, что число Ричардсона находится в интервале  $Ri = 120 \dots 300$ , т. е. много больше единицы.

Выше отмечалось, что котел вагона цистерны заполняется не полностью, и в нем всегда имеется плоская поверхность жидкости, показанная на рисунке 1, б участком I, а часть котла над ней заполнена воздухом. Движение цистерны сопровождается боковыми и продольными колебаниями, вызывающими в свою очередь колебания заполняющей котел жидкости. Эти колебания вызывают не естественную, а вынужденную конвекцию жидкости и характерная скорость ее движений составляет  $V \sim 0,2 \dots 0,3$  м/с. Число Ричардсона в этом случае остается большим  $Ri > 300$ , следовательно, стратифицированное состояние останется устойчивым и в случае колебаний жидкости.

Было выполнено численное моделирование процесса охлаждения различных ЗНП, находящихся в стратифицированном состоянии, при их перевозках в железнодорожных вагонах-цистернах. При расчетах использовались оригинальные пакеты программ, написанные на языке FORTRAN в стандарте 77 (оболочка FPS-1.0) [11, 12].

На рисунке 4 показаны результаты расчета температурного поля в мазуте М100 внутри котла вагона-цистерны модели 15-1566.

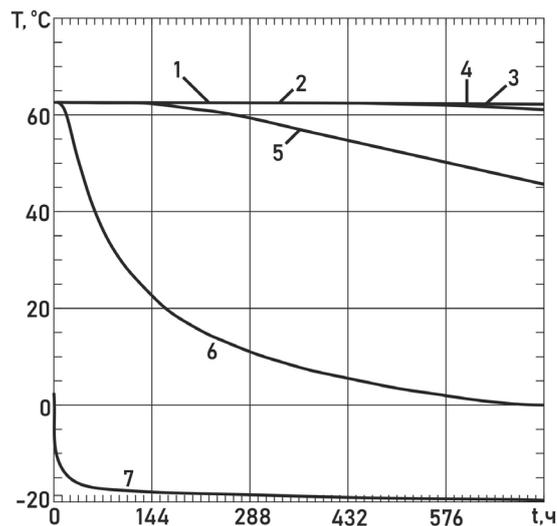


Рис. 4. Изменение температуры в массе мазута М100, перевозимого в стратифицированном состоянии в железнодорожном вагоне-цистерне при температуре воздуха  $T_{cp} = -20$  °С  
1 — ось котла; 2 —  $r = 0,1$  м; 3 —  $r = 0,4$  м; 4 —  $r = 0,8$  м; 5 —  $r = 1,2$  м; 6 —  $r = 1,45$  м; 7 —  $r = 1,5$  м

Условия расчета были следующими: радиус котла  $R = 3$  м, масса нефтепродукта — 67 т, начальная температура (при наливе)  $T_0 = +70$  °С, средняя температура воздуха  $T_{cp} = -20$  °С, коэффициент внешней теплоотдачи  $\alpha_{конт} = 50$  Вт/м<sup>2</sup>·°С (отвечает скорости воздуха 72 км/ч, получаемой при средней скорости движения поезда 45 км/ч в условиях встречного ветра, имеющего скорость 12 м/с).

По оси абсцисс отложено время охлаждения (ч), по оси ординат — температура нефтепродукта (°С), параметром  $r$  является расстояние от оси котла (м). Температура слива мазута (зависит от содержания в нем парафиновой фракции)  $T_{слив} \sim +50$  °С.

Из рисунка 4 следует, что за время перевозки в течение 432 ч (18 суток — среднее время доставки нефтегрузов на территории России с учетом азиатской части страны) охладится с переходом в высоковязкое состояние не вся масса мазута в котле вагона-цистерны, а лишь сравнительно тонкий слой, непосредственно прилегающий к стенке котла (обозначено кривыми 6 и 7). Толщина слоя возрастает с течением времени охлаждения, но не превосходит  $\delta = 0,05$  м (кривая 5). При этом, если оценочно несколько увеличить длину котла до  $L = 12$  м (для учета массы нефтепродукта застывшего на днищах котла), его масса будет меньше 4 тонн:

$$m = \pi r L (R^2 - r^2) \approx 2\pi r L R^2 \delta = 3\,770 \text{ (кг)}.$$

Остальная масса (кривые 1, 2, 3, 4 и 5 на рисунке 4) сохранит температуру, превышающую  $+50$  °С, то есть достаточную для выгрузки самотеком.

#### РЕКОМЕНДАЦИИ ПО РЕЗУЛЬТАТАМ РАСЧЕТА

Для обеспечения полной выгрузки доставленного нефтегруза нужно нагреть и расплавить не все содержимое

67-тонного котла вагона цистерны, а лишь малую его часть, не превосходящую 4 тонн. Этим обеспечивается экономия в затратах времени и тепловой энергии на организацию слива, а также сокращается время оборота подвижного состава, что является очень важным фактором.

Для перевода ЗНП в стратифицированное состояние нужно, чтобы его температура в верхней части вагона-цистерны была бы на 6–10 °С выше, чем в нижней его части.

С наименьшими затратами ресурсов этого можно добиться на стадии налива горячего нефтепродукта в цистерну по следующим вариантам:

- замедлить скорость охлаждения ЗНП в верхней половине котла установкой на нем временной (съемной) теплоизолирующей оболочки;
- дополнительным разогревом верхней половины котла инфракрасным излучением при принудительном охлаждении его нижней половины, например пропусканием под паробогревательный кожух технической воды;
- увеличением начальной температуры ЗНП при наливе в котел цистерны соответствующим мероприятием, отмеченным в предыдущем пункте.

Данные варианты проработаны авторами, имеются патенты на изобретения [13–16].

#### ЛИТЕРАТУРА

1. Математическое моделирование конвективного теплообмена на основе уравнений Навье — Стокса / В. И. Полежаев, А. В. Бунэ, Н. А. Вerezуб, [и др.]; отв. ред. В. С. Авдеевский. — Москва: Наука, 1987. — 272 с.

2. Петров, А. А. Углеводороды нефти / А. А. Петров; отв. ред. Н. С. Наметкин. — Москва: Наука, 1984. — 264 с.

3. Гуреев, А. А. Топливо для дизелей. Свойства и применение: Учебное пособие / А. А. Гуреев, В. С. Азев, Г. М. Камфер. — Москва: Химия, 1993. — 336 с. — (Для высшей школы).

4. Острах, С. Естественная конвекция внутри горизонтального цилиндра / С. Острах, Э. Р. Мэнлоу // Тепло- и массообмен: Материалы III Всесоюзного совещания по тепло- и массообмену (Минск, СССР, 14–18 мая 1968 г.): в 10 тт. Т. 1: Тепло- и массообмен при взаимодействии тел с потоками жидкостей и газов. — Москва: Энергия, 1968. — С. 640–661.

5. Гребер, Г. Основы учения о теплообмене = Die Grundgesetze der Wärmeübertragung. Dritte völlig neubearbeitete auflage / Г. Гребер, С. Эрк, У. Григуль; пер. с нем. под общ. ред. А. А. Гухмана. — Москва: Издательство иностранной литературы, 1958. — 566 с.

6. Повх, И. Л. Техническая гидромеханика. — 2-е изд., перераб. и доп. — Ленинград [Санкт-Петербург]: Машиностроение. Ленингр. отд-ние, 1976. — 504 с.

7. Гершуни, Г. З. Численное исследование конвективного движения в замкнутой полости / Г. З. Гершуни, Е. М. Жуховицкий, Е. Л. Тарунин // Известия Академии наук СССР. Механика жидкости и газа. 1966. № 5. С. 56–62.

8. Пасконов, В. М. Численное моделирование процессов тепло- и массообмена / В. М. Пасконов, В. И. Полежаев, Л. А. Чудов. — Москва: Наука. Главная редакция физико-математической литературы, 1984. — 288 с.

9. Основы метода конечных элементов: Учебное пособие / сост. Г. М. Макарьянц, А. Б. Прокофьев. — Самара: Изд-во Самарского гос. аэрокосмического ун-та им. С. П. Королёва, 2013. — 80 с.

10. Emery, A. F. Free Convection Through Vertical Plane Layers of Non-Newtonian Power Law Fluids / A. F. Emery, H. W. Chi, J. D. Dale // Journal of Heat Transfer. 1971. Vol. 93, Is. 2. Pp. 163–171. DOI: 10.1115/1.3449778.

11. Свидетельство о государственной регистрации программы для ЭВМ № 2012616057 Российская Федерация. Программа для расчета распределения температур в темных нефтепродуктах при их выгрузке из ж.-д. цистерны в зимних условиях: опублик. 03.07.2012 / В. И. Моисеев, Д. В. Елисеев.

12. Свидетельство о государственной регистрации программы для ЭВМ № 2012618249 Российская Федерация. Программа расчета температурных полей в цистерне с вязкими нефтепродуктами: опублик. 12.09.2012 / Д. В. Елисеев, В. И. Моисеев.

13. Железнодорожные цистерны: конструкции, техническое обслуживание и ремонт: Учебное пособие для работников железнодорожного транспорта / И. Г. Морчиладзе, А. П. Никодимов, М. М. Соколов, А. В. Третьяков. — Москва: ООО «ИБС-Холдинг», 2006. — 516 с.

14. Патент № 2682130 Российская Федерация, МПК В65D 88/00 (2006.01). Способ перевозки вязких нефтепродуктов и железнодорожная цистерна для его реализации: № 2018116284: заявл. 28.04.2018: опублик. 14.03.2019 / Моисеев В. И., Комарова Т. А.; заявитель ФГБОУ ВО ПГУПС. — 17 с.

15. Патент № 2666018 Российская Федерация, МПК В61D 5/00 (2006.01), МПК В61D 88/74 (2006.01). Цистерна для перевозки вязких нефтепродуктов: № 2017124560: заявл. 10.07.2017: опублик. 05.09.2018 / Моисеев В. И., Комарова Т. А.; заявитель ФГБОУ ВО ПГУПС. — 12 с.

16. Патент № 2639095 Российская Федерация, МПК В65D 88/00 (2006.01). Способ перевозки вязких нефтепродуктов и железнодорожная цистерна для его реализации: № 2016146672: заявл. 28.11.2016: опублик. 19.12.2017 / Моисеев В. И., Комарова Т. А.; заявитель ФГБОУ ВО ПГУПС. — 12 с.

# Mathematical Models and Numerical Methods of Heat and Mass Transfer During Natural Convection of Hot Liquid Petroleum Products in Boiler of Tank Wagon

Grand PhD V. I. Moiseev, V. A. Ksenofontova

Emperor Alexander I St. Petersburg State Transport University  
Saint Petersburg, Russia  
moiseev\_v\_i@list.ru, koc-vera@yandex.ru

PhD T. A. Komarova

Research and Implementation Center «Dispersed Systems»  
Saint Petersburg, Russia  
komarova\_tanusha@mail.ru

**Abstract.** The work considered the task of cooling stationary petroleum products which are stratified during their carriage in the tank-wagon boiler at low temperatures. A new approach to solving the problem is proposed: the oil cargo is considered as a binary system without a clear boundary between the fractions. The main goal is a mathematical description of heat and mass transfer in a hot-melting product. The numerical solution of the task was carried out using the method of nets.

**Keywords:** viscous petroleum products, rail transport, mathematical model, heat transfer, mass transfer, numerical methods, stratified state of liquid petroleum product.

## REFERENCES

1. Polezhaev V. I., Bune A. V., Verezub N. A., et al. Mathematical modeling of convective heat and mass transfer based on the Navier — Stokes equations [Matematicheskoe modelirovanie konvektivnogo teplomassoobmena na osnove uravneniy Nav'ye — Stoksa]. Moscow, Nauka Publishers, 1987, 272 p.
2. Petrov A. A. Petroleum hydrocarbons [Uglevodorody nefti]. Moscow, Nauka Publishers, 1984, 264 p.
3. Gureev A. A., Azev V. S., Kamfer G. M. Diesel fuel. Properties and applications: Study guide [Toplivo dlya dizeley. Svoystva i primeneniye: Uchebnoe posobie]. Moscow, Chemistry Publishers, 1993, 336 p.
4. Ostrach S., Manlow E. R. Natural Convection Inside a Horizontal Cylinder [Estestvennaya konvektsiya vnutri gorizontal'nogo tsilindra], *Heat and Mass Transfer: Proceedings of the III All-Union Conference on Heat and Mass Transfer. Vol. 1: Heat and Mass Transfer in the Interaction of Bodies with Flows of Liquids and Gases [Teplo- i massoperenos: Materialy III Vsesoyuznogo soveshchaniya po teplo- i massoobmenu. T. 1: Teplo- i massoperenos pri vzaimodeystvii tel s potokami zhidkostey i gazov]*, Minsk, USSR, May 14–18, 1968. Moscow, Energiya Publishers, 1968, Pp. 640–661.
5. Gröber H., Erk S., Grigull U. Die Grundgesetze der Wärmeübertragung. Dritte völlig neubearbeitete auflage [Osnovy ucheniya o teploobmene]. Moscow, Foreign Literature Publishing House, 1958, 566 p.
6. Povkh I. L. Technical fluid mechanics [Tekhnicheskaya gidromekhanika]. Leningrad [St. Petersburg], Mechanical Engineering Publishing House, 1976, 504 p.
7. Gershuni G. Z., Zhukhovitsky E. M., Tarunin E. L. Numerical Study of Convective Motion in a Closed Cavity [Chislennoe issledovanie konvektivnogo dvizheniya v zamknutoy polosti], *Proceedings of the Academy of Sciences of the USSR. Fluid and Gas Mechanics [Izvestiya Akademii nauk SSSR. Mekhanika zhidkosti i gaza]*, 1966, No. 5, Pp. 56–62.
8. Paskonov V. M., Polezhaev V. I., Chudov L. A. Numerical modeling of heat and mass transfer processes [Chislennoe modelirovanie protsessov teplo- i massoobmena]. Moscow, Nauka Publishers, 1984, 288 p.
9. Makar'yants G. M., Prokof'yev A. B. (eds) Finite element method fundamentals: Study guide [Osnovy metoda konechnykh elementov: Uchebnoe posobie]. Samara, S. P. Korolev Samara State Aerospace University, 2013, 80 p.
10. Emery A. F., Chi H. W., Dale J. D. Free Convection Through Vertical Plane Layers of Non-Newtonian Power Law Fluids, *Journal of Heat Transfer*, 1971, Vol. 93, Is. 2, Pp. 163–171. DOI: 10.1115/1.3449778.
11. Moiseev V. I., Eliseev D. V. A Program for Calculating the Temperature Distribution in Dark Oil Products During Their Unloading from the Railway Tanks in Winter [Programma dlya rascheta raspredeleniya temperatur v temnykh nefteproduktakh pri ikh vygruzke iz zheleznodorozhnoy tsisterny v zimnikh usloviyakh]. Certificate of State registration of a computer program RU No. 2012616057, published at July 03, 2012.
12. Eliseev D. V., Moiseev V. I. A Program for Calculating Temperature Fields in a Tank with Viscous Oil Products [Programma rascheta temperaturnykh poley v tsisterne s vyazkimi nefteproduktami]. Certificate of State registration of a computer program RU No. 2012618249, published at September 12, 2012.
13. Morchiladze I. G., Nikodimov A. P., Sokolov M. M., Tret'yakov A. V. Railroad tanks: Design, maintenance and repair: A training manual for railroad workers [Zheleznodorozhnye tsisterny: konstruksii, tekhnicheskoe obsluzhivanie i remont: Uchebnoe posobie dlya rabotnikov zheleznodorozhnogo transporta]. Moscow, IBS-Holding LLC, 2006, 516 p.
14. Moiseev V. I., Komarova T. A. Method of Transporting of Viscous Refined Oil Products and Railway Tank for Its Implementation [Sposob perevozki vyazkikh nefteproduktov i zheleznodorozhnaya tsisterna dlya ego realizatsii], patent RU No. 2682130, published at March 14, 2019, 17 p.
15. Moiseev V. I., Komarova T. A. Tank for Transportation of Viscous Oil Products [Tsisterna dlya perevozki vyazkikh nefteproduktov], patent RU No. 2666018, published at September 05, 2018, 12 p.
16. Moiseev V. I., Komarova T. A. Method of Transportation of Viscous Oil Products and Railway Tank for Its Implementation [Sposob perevozki vyazkikh nefteproduktov i zheleznodorozhnaya tsisterna dlya ego realizatsii], patent RU No. 2639095, published at December 19, 2017, 12 p.

# Особенности технологий бессерверных вычислений

А. Т. Гордина, к.и.н. А. В. Забродин

Петербургский государственный университет путей сообщения Императора Александра I

Санкт-Петербург, Россия

anngordtm@gmail.com, teach-case@yandex.ru

**Аннотация.** Статья посвящена одному из перспективных направлений в области облачных вычислений, известному под названием Serverless Computing (бессерверные вычисления). Рассматриваются технологии, применяемые для Serverless, дается сравнение облачных моделей и поставщиков услуг, описание основных компонентов архитектурного шаблона FaaS (функций, ресурсов и событий). Показаны преимущества и недостатки технологии Serverless-вычислений, а также раскрываются проблемы миграции приложений с локальных серверов в облако, и как при этом из процесса разработки программного продукта исключаются задачи по управлению ресурсами и их оптимизации.

**Ключевые слова:** облачные вычисления, бессерверные вычисления, модель FaaS, функции, автомасштабирование, архитектура приложения.

## ВВЕДЕНИЕ

Для функционирования и организации работы любой информационной системы необходима инфраструктура, которая позволит объединить, организовать и трансформировать информацию в соответствии с поставленными задачами. Инфраструктура является комплексом взаимосвязанных информационных сервисов и систем, таких как специализированное программное обеспечение, серверное, сетевое и компьютерное оборудование, СУБД, каналы связи, персонал и многое другое, необходимых для функционирования и развития инструментов информационного взаимодействия в компании. Создание эффективной инфраструктуры — это довольно сложный процесс, требующий значительных ресурсов (как финансовых, так и временных), а также высокого уровня компетенций. Одним из направлений снижения капитальных затрат на инфраструктуру, а также повышения гибкости информационных задач и процессов является использование облачных технологий (хостинга). Период пандемии и массовый переход на удаленную работу способствовали росту востребованности облачных платформ.

Несомненно, миграция в облако также приносит проблемы в планировании производительности и управлении операциями серверной части [1]. Так провайдеры облачных услуг стали развивать решения, в которых облако автоматически и динамически занимается распределением вычислительных ресурсов. Одним из таких решений является стратегия бессерверных вычислений. В парадигме Serverless весь процесс разработки программного обеспечения превращается в совокупность рабочих процессов с администрированием извне, что позволяет не акцентировать внимание на инфраструктуре вокруг, а сфокусироваться на самом процессе разработки.

Так, Serverless решения становятся следующим этапом развития архитектуры вычислительных систем в эпоху облаков.

## МОДЕЛЬ ОБСЛУЖИВАНИЯ «КАК УСЛУГА»

Сущность облачного хостинга заключается в том, что физические серверы провайдера соединяются в единый пул, мощности которого могут гибко распределяться между клиентами в зависимости от их потребностей.

Облачный рынок в настоящее время продолжает развиваться, расширяются модели облачных вычислений [2]. При планировании использования облаков нужно понимать, какие облачные сервисы существуют, чтобы остановить свой выбор на правильном и наиболее подходящем решении в пользу той или иной модели работы, что позволит найти необходимый баланс между гибкостью и простотой.

Неотъемлемыми характеристиками при выборе инфраструктурной платформы являются:

- увеличение скорости и качества разработки;
- автоматизация обслуживания и эксплуатации;
- безопасность и стабильность.

Концепция «как услуга» (англ. *as-a-Service, aaS*) подразумевает использование услуги с применением облачных технологий по необходимости. Каждая модель IT-инфраструктуры, которая поставляется в этом формате, уникальна по-своему. Рассмотрим самые востребованные aaS-сервисы, представленные на рынке облачных вычислений.

Модель обслуживания «инфраструктура как услуга» (англ. *Infrastructure-as-a-Service, IaaS*) стала первым шагом в модернизации IT-инфраструктуры при переходе от физических к виртуальным серверам. В 2012 году поставщики «инфраструктуры как услуги» занимали ведущие позиции в развитии облачных вычислений [3].

IaaS-провайдеры располагают услугами разного масштаба. Есть возможность арендовать единственный виртуальный сервер или пул виртуальных серверов с возможностью их объединения в виртуальную сеть.

Парадигма IaaS предоставляет полные административные права внутри арендованных виртуальных серверов. Все, что касается настройки операционных систем арендованных серверов, необходимо выполнять самостоятельно. Изначально IaaS-провайдер лишь гарантирует, что ваш сервер будет доступен по сети в соответствии с соглашением об уровне услуг (англ. *Service Level Agreement, SLA*) [4].

Преимуществом в выборе IaaS является тот факт, что такая модель подходит для случаев, когда значение имеет масштабируемость. IaaS также может быть полезна для приложений с постоянной нагрузкой или необходимостью наращивать вычислительные ресурсы.

Основной недостаток IaaS: по-прежнему остается необходимость и ответственность за управление безопасностью. Кроме того, IaaS-провайдеры предоставляют только серверы и API, а настройкой всего остального приходится заниматься самостоятельно.

Создание качественного программного обеспечения зависит от информационно-технологических платформ: среды разработки и тестирования, операционных систем, средств для развертывания, инструментов управления базами данных. Модель «платформа как услуга» (англ. *Platform-as-a-Service, PaaS*) позволяет размещать готовые программы на хостинге с поддержкой всего необходимого. Комплект сервисов PaaS передает ответственность по управлению всей информационно-технологической инфраструктурой провайдеру. Облачный провайдер берет на себя установку всех зависимостей операционной системы, необходимых для поставляемой версии среды выполнения. Остается только сосредоточиться на создании приложения, которое работает в выбранной среде выполнения.

Zimki (2006) и Heroku (2007) были первыми компаниями, которые использовали модель разработки «платформа как услуга» [5]. В то время было ясно, что такой подход значительно ускоряет время разработки и сокращает расходы на обслуживание. Это позволяет разработчикам настраивать и управлять приложениями, значительно сокращая время от начала разработки до выпуска продукта и последующей его поддержки.

Недостаток PaaS-модели заключается в том, что управление сфокусировано лишь на том, что построено на самой платформе. Провайдер берет ответственность за всю физическую инфраструктуру, а также администрирование на уровне операционных систем. Соответственно, PaaS предоставляет меньше управления над вычислительной инфраструктурой в сравнении с IaaS.

На данный момент практика показывает, что PaaS как отдельная модель в мире облачных услуг утратила свою актуальность. На смену независимым PaaS-платформам пришли PaaS-подобные сервисы [6].

Разработка программного продукта требует оптимального распределения ресурсов, при котором обеспечивается их наиболее эффективное использование, что позволяет внедрять инновации, снижать риски и сокращать время выхода продукта на рынок.

Приложение может использовать сотни серверов, и выбор Serverless для проектирования архитектуры приложения позволит избавиться от проблем, с которыми сталкиваются при управлении серверами. Serverless вовсе не означает отказ от сервера, данная парадигма позволяет разработчикам не беспокоиться о его настройке и администрировании.

В 2014 году Amazon анонсировал AWS Lambda, что позволило Serverless выйти на новый уровень [7]. Данный бессерверный, управляемый событиями вычислительный сервис представил иную тенденцию к минимизации кода и фокусировании на нем. Так появился архитектурный шаблон «функция как услуга» (англ. *Function-as-a-Service, FaaS*), который позволяет сократить необходимую инфраструктуру и инвестиции.

Модель FaaS вводит новый уровень абстракции, в котором исчезает физическая инфраструктура и архитектура программного обеспечения, остаются только «функции»,

работающие по необходимости. FaaS предоставляет наибольшую гибкость при регулировании производительности: в период простоя функция не потребляет ресурсы, а при необходимости платформа может быстро выделить необходимую мощность, которая справится с почти любой нагрузкой.

#### РАЗВИТИЕ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ

Модель облачных вычислений основывается на традиционном IT-стеке ПО, который состоит из пяти уровней: аппаратное обеспечение, виртуализация, операционная система, среда выполнения, приложение.

Уровень аппаратного обеспечения включает все вычислительные, сетевые и ресурсы хранения. Уровень выше — уровень виртуализации, где происходит виртуализация всех необходимых аппаратных ресурсов с помощью гипервизора для создания виртуальных машин для горизонтального масштабирования. Следующий уровень — операционная система. Здесь располагается программное обеспечение, которое находится на самом деле на удаленном компьютере и позволяет запускать программы. Вдобавок к этому запускается среда выполнения. Она является неотъемлемым связующим звеном между операционной системой и приложением [8]. И также будет средой, где работает приложение. Следующий уровень — уровень приложения — включает в себя всю бизнес-логику. Команды разработчиков проводят на данном уровне большую часть своего времени, потому что именно здесь происходит проектирование, создание и развертывание приложения.

IT-стек в стандартном виде сложен в обслуживании для компании, отказ на одном из уровней может повлиять на любой другой. Но с приходом облачных моделей подход к управлению изменился (рис. 1).

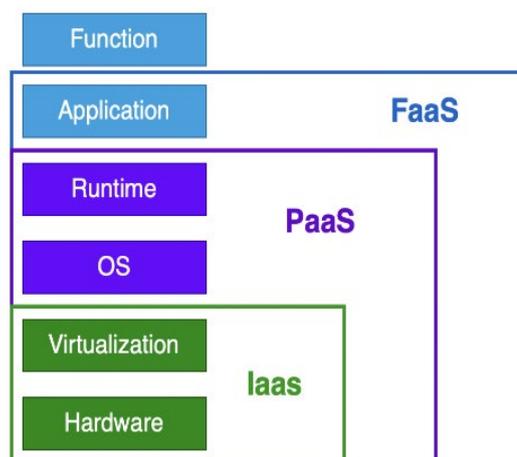


Рис. 1. Развитие облачных моделей

IaaS на самом деле абстрагирует нижние уровни инфраструктуры. Они запускаются от провайдера облачных услуг, что позволяет быстрее подготовить и запустить приложение.

PaaS поднимает уровень абстракции еще дальше. Используя данную модель, разработчики беспокоятся только о своем приложении: это единственное, что нуждается в поддержке. Пропадает необходимость думать о среде выполнения и операционной системе.

FaaS позволяет абстрагировать приложение и позволяет сосредоточиться на функциях. Они представляют собой отдельные компоненты, составляющие это приложение. Такой подход помогает сфокусироваться на разработке без необходимости эксплуатации или управления серверами.

#### ВЫБОР SERVERLESS FAAS

Одна из особенностей Serverless-архитектуры — отсутствие необходимости общения с серверами напрямую. Поэтому помимо термина «бессерверный» прибегают к его синониму — «бесхостовый» (hostless). Положение, которое исключает централизованную архитектуру, исключает расходы на обслуживание парка серверов. Hostless к тому же означает, что архитектура становится эластичной [9]. Это освобождает от ручного управления ресурсами, поэтому многие проблемы, связанные с распределением ресурсов, исчезают.

Идея, которую представляет модель FaaS, на самом деле проста. Традиционное программирование основано на функциях. Они обеспечивают предсказуемый результат и не меняют ничего в самой программе, что выходит за пределы функции. Сами функции являются «stateless», то есть не сохраняют состояние. Stateless также предполагает, что количество ошибок существенно уменьшается.

Модель FaaS предоставляет клиентам выполнение кода в облаке, а все операции по серверной части берут на себя провайдеры. Клиенты отвечают лишь за свои данные и функции, которые выполняются.

Одной из причин, по которой предпочтение отдают использованию Serverless FaaS, является возможность миновать управление процессами на уровне операционной системы и администрирования серверов. Некоторые сервисы PaaS также предоставляют эту возможность. В действительности концепции PaaS и FaaS похожи, но большинство приложений PaaS не ориентированы на то, чтобы запускать приложения в ответ на события, а платформы FaaS делают именно это.

Ключевое отличие FaaS от PaaS — масштабируемость приложения, основным назначением которого является задача смягчить проблемы, возникающие при сбое или перегрузке приложения. При использовании PaaS, как правило, необходимо думать о том, как масштабировать, все еще присутствует. Так, например, в одной из наиболее используемых облачных PaaS-платформ — Heroku — представлены «dynos» — изолированные виртуальные Linux-контейнеры, которые предназначены для выполнения кода по запросу заданной от пользователя команды. То есть один dyno — один запущенный экземпляр приложения. И сколько dynos необходимо запустить? С FaaS же этот процесс становится прозрачным. Даже если использовать автоматическое масштабирование в PaaS, приложение FaaS все еще эффективнее, когда дело доходит до затрат.

Еще одной абстракцией, которая активно используется в наше время, являются контейнеры приложений. Эта технология обеспечивает приложению среду, отдельную от операционной системы, позволяющую работать независимо от платформы. Системы управления контейнеризованными рабочими нагрузками и сервисами, такие как Mesos и Kubernetes, позволяют также абстрагировать приложения от развертывания на уровне операционной системы [10, 11]. А облачный сервис управления контейнерами Amazon ECS к тому же позволяет еще и избегать

управления собственными серверами. Тогда стоит ли оставаться на своем выборе на Serverless FaaS? Аргумент, который использовался при сравнении с PaaS, также актуален и для контейнеров: масштабирование все еще прозрачно и автоматически управляемо. А контейнерные платформы все еще требуют управления размером кластеров. Кроме того, FaaS рассматривается как наиболее подходящий выбор для событийно-ориентированной архитектуры, а при выборе контейнеров отдают предпочтение для компонентов, управляемых синхронным запросом с несколькими точками входа.

Сочетание контейнеров и Serverless может быть и полезным. Например, если в случае сложной контейнерной системы есть необходимость выполнить некоторые дополнительные задачи, которые вызваны событиями. Эти задачи будет выгоднее и удобнее перенести от контейнерных настроек к бессерверным функциям [12]. Таким образом, из этого следует интегрирование и передача информации о состоянии между контейнерной и серверной архитектурой.

При разработке программного обеспечения с помощью Serverless сначала определяется логика, необходимая для запуска провайдером, то есть проектируется необходимое количество функций. Шлюз-система API Gateway, которая является общей точкой входа в приложение, принимает и обрабатывает запросы от клиентов к функциям. Затем по требованию эти функции выполняются. Любые данные, которые сгенерированы функциями, возвращаются обратно устройству, которое их вызвало, также возможен вариант выгрузки во внешнее хранилище. Монолитный API заменяется функциями (рис. 2).

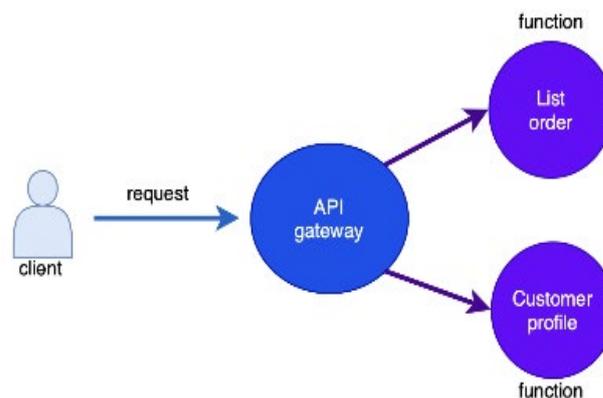


Рис. 2. Схема Serverless-приложения

Serverless-решения хорошо подходят для встраивания в микросервисную архитектуру. В случае небольшой кодовой базы разворачивание отдельного контейнера с операционной системой смысла никакого не имеет, так как это неэкономично с точки зрения использования ресурсов. В таком случае легче использовать функцию. Кроме того, возможно использовать функции как вспомогательные сервисы для тестовой среды. Каждое выполнение функции изолировано и эфемерно, поэтому проектирование приложения с архитектурой Serverless с использованием FaaS требует кроме функции еще и управления данными в хранилищах, механизмов запуска, масштабирования, логирования. В связи с этим построение Serverless не ограничивается

только FaaS. Провайдеры бессерверных вычислений предлагают целые экосистемы состоящие из Serverless-сервисов.

#### ПРОВАЙДЕРЫ SERVERLESS

Как правило, Serverless-вычисления представляют провайдеры облачных услуг. Бессерверные вычисления способствуют росту бизнеса по той причине, что облегчение программирования приложения на Serverless-архитектуре помогает привлекать новых клиентов, а уже существующим клиентам использовать больше возможностей облачных приложений.

Выбирая подходящий сервис, отвечающий необходимым условиям, нужно учитывать следующие ключевые факторы:

- поддержка языков программирования;
- типы триггера функций;
- безопасность;
- ценообразование и факторы биллинга;
- время выполнения и параллелизм;
- метод развертывания.

Serverless развивается, и вместе с ним на рынке появляется все больше возможностей и сервисов, которые предлагают провайдеры.

#### *AWS Lambda*

Компания Amazon была пионером в развитии Serverless-технологий. На текущий момент в рамках Amazon Web Services корпорация предлагает сервис для бессерверных вычислений AWS Lambda.

Суть Lambda заключается в выполнении кода при реагировании на различные события. Иными словами, единственным, что требуется в использовании данного сервиса, является загрузка кода, который должен исполняться в функции, и настройка триггера или события, которое отвечает за выполнение кода. Всю остальную ответственность за администрирование платформы AWS Lambda полностью берет на себя. Таким образом, пропадает необходимость в обслуживании серверов, управлении операционной системой, выделении нужных ресурсов, масштабировании и мониторинге кода. В AWS Lambda от пользователя скрыта внутренняя реализация модели и почти все внутренние процессы, поэтому теряется актуальность безопасности данной платформы.

Список сервисов AWS для бессерверных вычислений широк. Вместе с Lambda активно используется Amazon API Gateway, с помощью которого можно значительно сократить расходы. Изменение данных происходит в корзинах Amazon S3 и базах данных Amazon DynamoDB. С сервисом для оркестрации бессерверных вычислений AWS Step Functions обычно проектируется и выполняется процесс, который может координировать несколько функций AWS Lambda. Также, используя комплект средств разработки AWS SDK, есть возможность запускать код через вызовы API. Кроме того, Lambda охватывает широкий спектр языков программирования.

#### *Google Cloud Functions*

Следующим по распространенности поставщиком услуг FaaS в мире является Google Cloud Functions. Данный сервис также позволяет абстрагироваться от всей базовой инфраструктуры, что позволяет сосредоточиться на коде.

Google Cloud Functions отлично интегрирована с сервисами Google. Сервис обеспечивает быстрый переход от кода к развертыванию, который включает интегрированный мониторинг. Полный обзор и диагностику приложения позволяет получить сервис Google Cloud Trace, предназначенный для оптимизации работы в облаке, и сервис Google Cloud Debugger, который проверяет состояние запущенного приложения в режиме реального времени. Кроме того, Cloud Functions позволяет использовать сторонние облачные сервисы, которые являются строительными блоками для кода.

Ценообразование основывается на количестве вызовов функций. Первые два миллиона вызовов бесплатны, каждый последующий миллион вызовов — 0,40 долларов США. Однако Google взимает дополнительную плату за использование памяти и ЦП.

Присутствует автомасштабирование: Cloud Functions автоматически управляют и масштабируют необходимую инфраструктуру в зависимости от рабочей нагрузки.

#### *Microsoft Azure Cloud Functions*

На текущий момент компания Microsoft является лидером среди поставщиков облачных услуг [13] и представляет платформу для бессерверных вычислений Azure Cloud Functions.

Время на написание кода значительно сокращается при использовании триггеров и привязок, что позволяет приложениям реагировать на события и без особых проблем подключаться к другим службам. Функции Azure предоставляют создание более масштабируемых и стабильных событийно-ориентированных приложений.

Платформа Azure для бессерверных вычислений позволяет создавать надежные приложения со встроенными средствами защиты и мониторинга. Сервис Azure Application Insights предоставляет средства для отслеживания и анализа производительности кода. Контроль политики доступа происходит в безопасном хранилище секретов Azure Key Vault, где можно хранить исходные параметры приложения, при этом оставляя код без изменений.

Основное преимущество Azure Functions — применение любого стека технологий. Использование сервиса не предполагает ограничения на выбор какого-то конкретного языка или сервера. Что дает возможность выбрать оптимальный язык программирования для каждого сценария. В случае использования языка, не имеющего встроенной поддержки, Azure Cloud Functions предлагают особые стратегии для создания функций [14].

Метод ценообразования в Azure такой же, как и в Lambda, и также предоставляются бесплатные пакеты на первые вызовы функций, далее взимается 0,000016 долларов США за гигабайтосекунду. Стоимость при высокой нагрузке у Azure немного ниже, если сравнивать с Lambda. Кроме того, Microsoft также взимает плату за использование памяти.

#### *Yandex Cloud Functions*

Отечественная IT-компания также не отстает от зарубежных и активно развивает свою Serverless-экосистему на платформе Yandex.Cloud. Она берет на себя обслуживание и все необходимые условия для непрерывной работы. Yandex.Cloud предоставляет отказоустойчивое и автомасштабируемое окружение для всей Serverless-экосистемы.

В нее входит сервис Yandex Cloud Functions, который позволяет запускать код приложения в виде функции.

В сервисе присутствуют триггеры для запуска функции; с помощью данной возможности интеграция между Cloud Functions и другими сервисами происходит легко и без использования интеграционного кода. К поддерживаемым языкам программирования на текущий момент относятся: Node.js, Python, Go, PHP, Bash, Java, C# и R [15].

Автомасштабирование позволяет при увеличении нагрузки на количество вызовов создавать дополнительные экземпляры функции. Функции, которые запущены, имеют возможность выполняться параллельно.

В сервисе Cloud Functions действуют удобные тарифы: каждый месяц использования в тариф не входят один миллион вызовов функций и 10 ГБ/ч выполнения функций. Далее за 1 ГБ/ч взимается плата 3,42 руб., стоимость вызова 1 млн функций — 10 руб.

В Serverless-экосистему Yandex.Cloud помимо Yandex Cloud Functions также входят следующие сервисы:

1. Yandex API Gateway. Сервис управления API-шлюзами, который передает запросы от приложений к сервисам Yandex.Cloud и сокращает задержку при обработке запросов к API.

2. Yandex Database (YDB). База данных имеет поддержку Serverless-режима и предоставляет автоматизированное масштабирование выполнения запросов, хранения и резервного копирования. Кроме того, YDB совместима с Amazon DynamoDB.

3. Сервис очереди сообщений Message Queue. При масштабировании системы использование сервиса очередей является неотъемлемой частью обмена сообщениями между компонентами. Очередь сообщений позволяет освобождать ресурсы для новых запросов при выполнении длительных задач.

Таким образом, использование Serverless-сервисов Yandex.Cloud позволяет создавать надежные проекты с простым управлением. В бессерверной инфраструктуре обработка запросов происходит посредством API Gateway, далее они перенаправляются в Cloud Functions. Хранение данных в Yandex Database автоматически масштабируется при росте интенсивности запросов.

Сравнительная характеристика вышеупомянутых провайдеров услуг представлена на рисунке 3.

	AWS Lambda	Google Cloud Functions	Azure Cloud Functions	Yandex Cloud Functions
Стоимость	Первые 1М в месяц бесплатно Далее \$0.2/1М	\$0.4/М	Первые 1М в месяц бесплатно Далее - \$0.16/1М	Первые 1М в месяц бесплатно Далее - ¥10/1М
Встроенная поддержка языка	Java, Go, PowerShell, Node.js, C#, Python, Ruby	Node.js, Python, Go, Java, .NET, Ruby, PHP	Node.js, C #, F #, Python, PHP, Bash, Batch, PowerShell	Node.js, Python, Go, PHP, Bash, Java, C#, R
Интеграция со сторонними сервисами	X	✓	✓	✓
Выполнение функции	1000 выполнений одновременно	1000 выполнений одновременно Для HTTP триггеров не ограничено	Не ограничено Max время выполнения - 5 мин	10 в каждой зоне доступности
Параллелизм	Для функции	Для приложения Для функции	Для функции	Для функции

Рис. 3. Сравнение провайдеров услуг

При выборе провайдера важно помнить и о рисках Serverless, которые связаны с Vendor Lock, привязкой к поставщику. Функции, созданные с помощью AWS, сложно импортируются в другие платформы, например в Google Cloud. Помимо функций обычно используются базы данных, сервисы очередей, логирования и прочие Serverless-сервисы, которые у каждого провайдера могут быть свои. Например, Yandex Cloud Functions предусматривает совместимость с AWS [16].

#### ПРЕИМУЩЕСТВА SERVERLESS

Основные потенциальные преимущества бессерверных вычислений заключаются в следующем:

1. Абстракция от внешней инфраструктуры. Несомненным и главным преимуществом Serverless является абстракция от рабочих процессов, связанных с администрированием серверов и управлением операционной системой.

2. Низкие эксплуатационные расходы. Поддержка меньшего количества компонентов означает меньшее ко-

личество затраченной работы. Serverless-приложения выпускаются быстрее, а значит, время для получения обратной связи от конечного пользователя также уменьшается, такая возможность влечет улучшения в будущем в течение всей разработки программного обеспечения.

3. Максимальная эластичность. Одна из сильных сторон Serverless — масштабирование. Оно происходит без внесенных изменений в уже написанный код, и при росте трафика количество запущенных функций также возрастает.

4. Эффективная стоимость. Оплата происходит только за используемые вычислительные ресурсы, то есть не взимается, когда приложение простаивает.

5. Балансировка нагрузки. В случае высокой посещаемости сервиса возрастает нагрузка на железо. В традиционной архитектуре балансировщик распределяет нагрузку на виртуальные машины. Несомненно, что внедрение Serverless окажет положительный эффект, то есть функции, запросы к которым приходят от балансировщика через API Gateway, будут брать на себя ответственность в обработке части запросов. Таким образом, система оптимизируется, распределяя нагрузку (рис. 4). В таком случае запуск функций обходится выгоднее, чем все время работающие виртуальные машины.

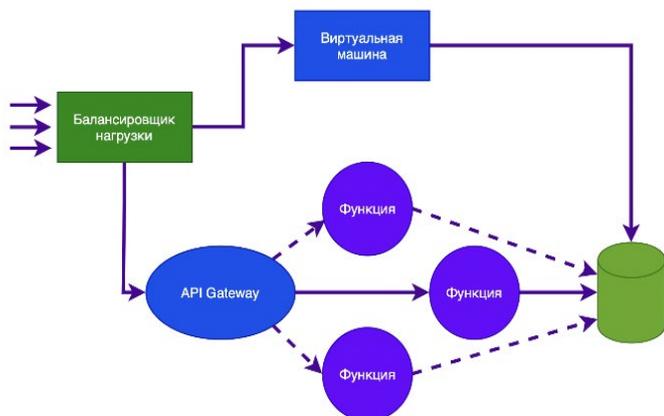


Рис. 4. Архитектура с Serverless-подходом и балансировщиком нагрузки

6. Встроенные интеграции. Большинство представленных на рынке провайдеров Serverless предлагают интеграцию с другими сервисами. Это преимущество благоприятно сказывается на создании программного обеспечения, так как основное внимание уделяется разработке, а не настройке и управлению.

#### НЕДОСТАТКИ SERVERLESS

Как и любая технология, Serverless обладает рядом недостатков. Некоторые из них неподвластны прогрессу и всегда будут актуальны, другие же связаны с текущей реализацией и в скором времени найдут решение, которое позволит технологии быть лучше.

1. Контроль за поставщиками. Прибегая к услугам провайдера, он берет на себя контроль над частями вашей системы. Отсутствие контроля может выражаться в простое, неожиданном изменении стоимости услуг, потере функциональности, принудительном обновлении API и прочим проблемам. Кроме того, определенные способы развертывания, мониторинга, создания Serverless-функций у разных провайдеров разные. И в данном случае не стоит забывать

об обратной совместимости: от любого интерфейса может зависеть другой сервис или функция.

2. Время холодного запуска. Время холодного старта при запуске бессерверных вычислений появляется вследствие того, что при срабатывании триггера сначала происходит загрузка среды выполнения, а лишь потом запуск функции (рис. 5).



Рис. 5. Схема запуска функции при холодном старте

Управлять средой выполнения у разработчика нет возможности, в итоге жизненный цикл среды может быть непредсказуем. Помимо этого, на холодный запуск влияет много зависящих друг от друга факторов: язык программирования, объем написанного кода, встроенные дополнительные ресурсы, такие как базы данных. Разработчик может этим управлять, сократив время старта, но доступ к управлению контейнером есть только у провайдера.

Холодный старт может стать теплым в случае, если количество обращений на функцию растет, или если разработчик запускает функции, следуя времени по таймеру.

3. Serverless не подходит для долгосрочных процессов. При проектировании приложения нужно учитывать, что если в архитектуре присутствует функция, которая используется приложением постоянно, то и потребление ресурсов не будет отличаться от традиционного подхода. Поэтому накладывать ответственность на функцию нужно разумно, тщательно обдумав ее дальнейшее использование.

#### ЗАКЛЮЧЕНИЕ

На основании вышесказанного логично предположить, что облачные вычисления продолжают набирать свою популярность, как и бессерверные технологии, которые предоставляют упрощенную среду для программирования, что намного облегчает использование облака. Несомненно, это способствует привлечению людей, заинтересованных в его использовании.

Неоспоримое преимущество Serverless заключается в фокусировании на разработке кода и избавлении от необходимости управления ресурсами и их оптимизации вручную. Такой подход схож с переходом от языка ассемблера на высокоуровневые языки. При этом следует отметить и повышенную сложность разработки и тестирования приложений по причине того, что бессерверная архитектура требует структурного разделения одного приложения на сервисы и функции, а это в свою очередь значительно увеличивает время проектирования архитектуры [17].

Очевидной ценностью Serverless является сокращение объема инвестиций, необходимых для поддержки работы приложений, что позволяет получить больше пространства для экспериментов и инноваций.

Несмотря на уже достигнутый успех в Serverless вычислениях, они продолжают свою эволюцию, поставщики услуг упорно работают над развитием новых услуг и улучшением уже существующих, устраняя выявленные недостатки. Разработчики уже приступили к изучению вопроса

о дальнейшем процессе развития бессерверных платформ и повышения эффективности рабочей нагрузки [18]. Решаются проблемы в областях абстракции, систем, сетей, безопасности и архитектуры. Нетрудно спрогнозировать, что дальнейший рост и изменение бессерверности будет и идти дальше. Так через десятки лет Serverless станет стандартом среди облачных технологий, заменив серверные вычисления.

ЛИТЕРАТУРА

1. Occupy the Cloud: Distributed Computing for the 99% / E. Jonas, Q. Pu, S. Venkataraman, [et al.] // Proceedings of the Eighth ACM Symposium on Cloud Computing (SoCC'17) (Santa Clara, CA, USA, 24–27 September 2017). — New York (NY): Association for Computing Machinery, 2017. — Pp. 445–451. DOI: 10.1145/3127479.3128601.
2. Переход в облако стал повседневной задачей для бизнеса // C-News. — 2018. — 31 октября. URL: [http://www.cnews.ru/articles/2018-10-31\\_perehod\\_v\\_oblako\\_stal\\_povsednevnoj\\_zadachej\\_dlya\\_srednego\\_biznesa](http://www.cnews.ru/articles/2018-10-31_perehod_v_oblako_stal_povsednevnoj_zadachej_dlya_srednego_biznesa) (дата обращения 07.03.2022).
3. Тренды развития облачных вычислений // TAdviser — Государство. Бизнес. Технологии. — 2022. — 24 января. URL: <http://www.tadviser.ru/a/147459> (дата обращения 07.03.2022).
4. Что такое IaaS-облако, сравнение с PaaS и SaaS на примерах // 1cloud. URL: <http://1cloud.ru/services/private-cloud/iaas-paas-saas> (дата обращения 08.03.2022).
5. Desde IaaS, SaaS, PaaS y FaaS, ¿Cómo Elegir la Plataforma Adecuada? / Traducido por Fu Hui // Programador Clic. URL: <http://programadorclick.com/article/80131207853> (дата обращения 09.03.2022).
6. Стельмах, С. Облачные вычисления: пик популярности пройден? // itWeek. — 2020. — 10 февраля. URL: <http://www.itweek.ru/its/article/detail.php?ID=211336> (дата обращения 09.03.2022).
7. Avram, A. FaaS, PaaS, and the Benefits of the Serverless Architecture // InfoQ. — 2016. — 25 June. URL: <http://www.infoq.com/news/2016/06/faas-serverless-architecture> (дата обращения 09.03.2022).
8. Delimitrou, C. Quasar: Resource-Efficient and QoS-Aware Cluster Management / C. Delimitrou, C. Kozyrakis // Proceedings of the 19th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS XIX) (Salt Lake City, UT, USA, 01–05 March 2014). ACM SIGPLAN Notices. 2014. Vol. 49, Is. 4. Pp. 127–144. DOI: 10.1145/2644865.2541941.
9. Peeking Behind the Curtains of Serverless Platforms / L. Wang, M. Li, Y. Zhang, [et al.] // Proceedings of the 2018 USENIX Annual Technical Conference (USENIX ATC'18) (Boston, MA, USA, 11–13 July, 2018). — Washington (D.C.): USENIX Association, 2018. — Pp. 133–145.
10. Chandra, B. Mesos — Creating Your Own Serverless // Medium. — 2019. — 17 July. URL: <http://bhanuandchandra.medium.com/mesos-creating-your-own-serverless-2aaf65cfb0b7> (дата обращения: 10.03.2022).
11. Brewer, E. A. Kubernetes and the Path to Cloud Native // Proceedings of the Sixth ACM Symposium on Cloud Computing (SoCC'15) (Kohala Coast, HI, USA, 27–29 August 2015). — New York (NY): Association for Computing Machinery, 2015. — P. 167. DOI: 10.1145/2806777.2809955.
12. Serverless vs Containers: Choose One or Use Both? // Cloud Native Wiki. URL: <http://www.aquasec.com/cloud-native-academy/serverless-architecture/serverless-vs-containers-choose-one-or-use-both> (дата обращения 10.03.2022).
13. Cloud Wars Top 10 — The World's Top Cloud Vendors // Acceleration Economy. URL: <http://accelerationeconomy.com/cloud-wars-top-10> (дата обращения 15.03.2022).
14. Supported Languages in Azure Functions // Microsoft Docs. — 2021. — 21 October. URL: <http://docs.microsoft.com/en-gb/azure/azure-functions/supported-languages> (дата обращения 15.03.2022).
15. Среда выполнения. Обзор // Yandex.Cloud. URL: <http://cloud.yandex.ru/docs/functions/concepts/runtime> (дата обращения 12.03.2022).
16. Serverless — бессерверные вычисления // Yandex.Cloud. URL: <http://cloud.yandex.ru/solutions/serverless> (дата обращения 12.03.2022).
17. Hennesy, J. L. A New Golden Age for Computer Architecture / J. L. Hennesy, D. A. Patterson // Communications of the ACM. 2019. Vol. 62, Is. 2. Pp. 48–60. DOI: 10.1145/3282307.
18. Serverless Computing: One Step Forward, Two Steps Back / J. M. Hellerstein, J. Faleiro, J. E. Gonzalez, [et al.] // Proceedings of the 9th Biennial Conference on Innovative Data Systems Research (CIDR 2019) (Asilomar, CA, USA, 13–16 January 2019). 9 p. URL: <http://cidrdb.org/cidr2019/papers/p119-hellerstein-cidr19.pdf> (дата обращения 12.03.2022).

# Characteristics of Serverless Computing Technologies

A. T. Gordina, PhD A. V. Zabrodin

Emperor Alexander I St. Petersburg State Transport University

Saint Petersburg, Russia

anngordtm@gmail.com, teach-case@yandex.ru

**Abstract.** This article is devoted to one of the prospective types of cloud computing known as Serverless computing. The paper considers the technologies used for Serverless, gives comparison of cloud models and service providers, also it describes main components of FaaS architectural pattern (functions, resources and events). The authors consider the advantages and disadvantages of Serverless computing. Besides that, the article discloses problems of application migration from local servers to the cloud and how resource management and optimization are excluded from the software development process.

**Keywords:** cloud computing, Serverless, FaaS model, functions, autoscaling, application architecture.

## REFERENCES

- Jonas E., Pu Q., Venkataraman S., et al. Occupy the Cloud: Distributed Computing for the 99%, *Proceedings of the Eighth ACM Symposium on Cloud Computing (SoCC'17)*, Santa Clara, CA, USA, September 24–27, 2017. New York (NY), Association for Computing Machinery, 2017, Pp. 445–451. DOI: 10.1145/3127479.3128601.
- Migration to Cloud Has Become Routine Task for Business [Perekhod v oblako stal povsednevnoy zadachey dlya biznesa], *C-News*. Published online at October 31, 2018. Available at: [http://www.cnews.ru/articles/2018-10-31\\_perekhod\\_v\\_oblako\\_stal\\_povsednevnoj\\_zadachej\\_dlya\\_srednego\\_biznesa](http://www.cnews.ru/articles/2018-10-31_perekhod_v_oblako_stal_povsednevnoj_zadachej_dlya_srednego_biznesa) (accessed 07 Mar 2022).
- Development Trends of Cloud Computing [Trendy razvitiya oblachnykh vychisleniy], *T-Adviser — Government. Business. Technology [TAdviser — Gosudarstvo. Biznes. Tekhnologii]*. Published online at January 24, 2022. Available at: <http://www.tadviser.ru/a/147459> (accessed 07 Mar 2022).
- What is IaaS Cloud, Comparison with PaaS and SaaS on Examples [Chto takoe IaaS-oblako, sravnenie s PaaS i SaaS na primerakh], *1cloud*. Available at: <http://1cloud.ru/services/private-cloud/iaas-paas-saas> (accessed 08 Mar 2022).
- From IaaS, CaaS, PaaS and FaaS, How to Choose the Right Silver-Form, *Programmer click*. Available at: <http://programmerclick.com/article/80131207853> (in Spanish) (accessed 09 Mar 2022).
- Stelmakh S. Cloud Computing: Boom Has Gone? [Oblachnye vychisleniya: pik populyarnosti proyden?], *itWeek*. Published online at February 10, 2020. Available at: <http://www.itweek.ru/its/article/detail.php?ID=211336> (accessed 09 Mar 2022).
- Avram A. FaaS, PaaS, and the Benefits of the Serverless Architecture, *InfoQ*. Published online at June 25, 2019. Available at: <http://www.infoq.com/news/2016/06/faas-serverless-architecture> (accessed 09 Mar 2022).
- Delimitrou C., Kozyrakis C. Quasar: Resource-Efficient and QoS-Aware Cluster Management, *Proceedings of the 19th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS XIX)*, Salt Lake City, UT, USA, March 01–05, 2014. *ACM SIGPLAN Notices*, 2014, Vol. 49, Is. 4, Pp. 127–144. DOI: 10.1145/2644865.2541941.
- Wang L., Li M., Zhang Y., et al. Peeking Behind the Curtains of Serverless Platforms, *Proceedings of the 2018 USENIX Annual Technical Conference (USENIX ATC'18)*, Boston, MA, USA, July 11–13, 2018. Washington (D.C.), USENIX Association, 2018, Pp. 133–145.
- Chandra B. Mesos — Creating Your Own Serverless, *Medium*. Published online at July 17, 2019. Available at: <http://bhanuandchandra.medium.com/mesos-creating-your-own-serverless-2aaf65cfb0b7> (accessed 10 Mar 2022).
- Brewer E. A. Kubernetes and the Path to Cloud Native, *Proceedings of the Sixth ACM Symposium on Cloud Computing (SoCC'15)*, Kohala Coast, HI, USA, August 27–29, 2015. New York (NY), Association for Computing Machinery, 2015, P. 167. DOI: 10.1145/2806777.2809955.
- Serverless vs Containers: Choose One or Use Both, *Cloud Native Wiki*. Available at: <http://www.aquasec.com/cloud-native-academy/serverless-architecture/serverless-vs-containers-choose-one-or-use-both> (accessed 10 Mar 2022).
- Cloud Wars Top 10 — The World's Top Cloud Vendors, *Acceleration Economy*. Available at: <http://acceleration-economy.com/cloud-wars-top-10> (accessed 15 Mar 2022).
- Supported Languages in Azure Functions, *Microsoft Docs*. Published online at October 21, 2021. Available at: <http://docs.microsoft.com/en-gb/azure/azure-functions/supported-languages> (accessed 15 Mar 2022).
- Runtime. Overview [Sreda vypolneniya. Obzor], *Yandex.Cloud*. Available at: <http://cloud.yandex.ru/docs/functions/concepts/runtime> (accessed 12 Mar 2022).
- Serverless Computing [Serverless — besservernyye vychisleniya], *Yandex.Cloud*. Available at: <http://cloud.yandex.ru/solutions/serverless> (accessed 12 Mar 2022).
- Hennesy J. L., Patterson D. A. A New Golden Age for Computer Architecture, *Communications of the ACM*, 2019, Vol. 62, Is. 2, Pp. 48–60. DOI: 10.1145/3282307.
- Hellerstein J. M., Faleiro J., Gonzales J. E., et al. Serverless Computing: One Step Forward, Two Steps Back, *Proceedings of the 9th Biennial Conference on Innovative Data Systems Research (CIDR 2019)*, Asilomar, CA, USA, January 13–16, 2019. 9 p. URL: <http://cidrdb.org/cidr2019/papers/p119-hellerstein-cidr19.pdf> (accessed 12 Mar 2022).

# Оптимальное планирование использования вагонного парка операторской компании

В. А. Варфоломеев, д.т.н. Э. К. Лецкий

Российский университет транспорта

Москва, Россия

vvictor@mail.ru, letzky@mail.ru

**Аннотация.** Изложена концепция создания центра интеллектуальной поддержки принятия решений при управлении транспортным комплексом. Услуги по планированию и управлению перевозками в различных сегментах транспорта предлагается осуществлять на основе технологий веб-сервисов. При этом пользователями услуг будут являться компании-перевозчики. Обоснованием создания подобного центра являются:

- наличие в каждом сегменте транспорта типовых задач планирования и оперативного управления перевозками;
- вычислительная сложность указанных задач, требующая для их решения использовать высокопроизводительные средства вычислений.

Задачи центра — это поддержка уже созданных сервисов и разработка новых. В статье рассмотрена типовая архитектура сервиса поддержки принятия решений, предусматривающая возможность уточнения постановки задачи в режиме диалога пользователя с чат-ботом. В качестве примера сервиса рассмотрена математическая постановка задачи календарного планирования перемещения вагонов операторской компании при выполнении заданного множества заказов на перевозку грузов.

**Ключевые слова:** поддержка принятия решений, веб-сервис, чат-бот, математическая модель, управление транспортными процессами.

## ВВЕДЕНИЕ

При планировании и оперативном управлении перевозками грузов, осуществляемыми различными организациями, возникает необходимость в решении типовых задач, таких как выбор транспортных средств, определение маршрутов перемещения, календарное планирование использования средств перевозок и т. д. [1, 2]. Эти задачи возникают как в отдельных сегментах транспорта, так и при смешанных перевозках, использующих разные виды транспорта. Близость этих задач не только в их содержательной постановке, но и в том, что для их решения требуется одна и та же информация о состоянии транспортной инфраструктуры, тарифах, нормативах и пр.

Естественно, что при планировании и управлении каждым конкретным перевозочным мероприятием могут иметь место специфические требования и ограничения, которые необходимо учитывать при принятии решений. Важно отметить, что рассматриваемые задачи часто требуют для поиска решения использования высокопроизводительных средств вычислений из-за высокой размерности или ограничения сроков получения

решений. В описанных условиях повышение качества функционирования транспортного комплекса может быть обеспечено созданием системы поддержки принятия решений, основанной на использовании совокупности общедоступных веб-сервисов [3], позволяющих в диалоге с чат-ботом [4, 5] уточнить постановку задачи и получить результаты решения за требуемое время.

## СТРУКТУРА ВЕБ-СЕРВИСА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

Основное назначение системы интеллектуальной поддержки принятия решения [6] — это предоставление пользователям совокупности сервисов, каждый из которых относится к одному из типов задач планирования и управления транспортными процессами. Состав сервисов должен постоянно корректироваться и расширяться в соответствии с потребностями транспортных предприятий и полученными результатами в области постановки решения новых типов задач. Организация этой деятельности, так же как и эксплуатация системы, может быть возложена на центр интеллектуальной поддержки принятия решений.

Предлагается использовать веб-сервис, выполняющий следующие функции (рис. 1):

- уточнение математической постановки задачи (выбор критерия оптимальности решения, выбор ограничений) в режиме диалога пользователя с чат-ботом [4];
- сбор из внешних источников и хранение данных, необходимых для решения задачи;
- поиск решения задач на основе методов, привязанных к выбранному варианту постановки задачи;
- предоставление результатов решения в формате, определяемом пользователем.



Рис. 1. Функции веб-сервиса

ПРИМЕР ВЕБ-СЕРВИСА: ЗАДАЧА ПЛАНИРОВАНИЯ  
ИСПОЛЬЗОВАНИЯ ВАГОННОГО ПАРКА  
ТРАНСПОРТНОЙ КОМПАНИИ

Суть задачи состоит в следующем. Компания приняла  $K$  заказов на перевозку грузов на интервале времени длиной в  $D$  суток. Каждый  $i$ -й заказ ( $i = 1, \dots, K$ ) определяется:

- наименованием груза (каждый заказ — один вид грузов);
- множеством типов вагонов  $M_i$ , приемлемых для перевозки груза  $i$ -го заказа ( $M_i$  содержит номера типов вагонов, приемлемых для груза  $i$ -го заказа);
- пункты погрузки и разгрузки ( $\gamma, \lambda$ ), где  $\gamma, \lambda = 1, 2, \dots, N$ , где  $N$  — число станций;
- объем  $i$ -го заказа  $x_i^*$  (число груженых вагонов);
- штраф за недопоставку  $\beta_i$  (за каждый вагон из  $x_i^*$ , погрузка которого не была проведена на интервале планирования);
- срок выполнения заказа  $t_i^{(k)}$ ;
- штраф за доставку груза позднее установленного срока  $\alpha_i$  (за суточную задержку одного вагона);

Компания располагает грузовыми вагонами различного типа. Известно начальное размещение грузовых вагонов по пунктам сети  $X_j^{(\mu)}$ , где  $X_j^{(\mu)}$  — число вагонов  $\mu$ -го типа, находящиеся в начальный момент планового периода в пункте  $j$ .

Необходимо построить оптимальный по критерию затрат календарный план перемещения вагонов компании, то есть найти следующую совокупность величин:

- $x_i^{(\mu d)}$  — число погруженных вагонов  $\mu$ -го типа  $i$ -го заказа в сутки  $d$  планового периода ( $i = 1, \dots, K$ ;  $\mu = 1, \dots, M$ ;  $d = 1, \dots, D$ );
- $y_{\lambda\gamma}^{(\mu d)}$  — число порожних вагонов  $\mu$ -го типа, отправленных из пункта  $\lambda$  в пункт  $\gamma$  в сутки  $d$  планового периода ( $\mu = 1, \dots, M$ ;  $\lambda, \gamma = 1, \dots, N$ ;  $d = 1, \dots, D$ ).

В общем случае затраты на перевозку могут включать следующие составляющие:

- затраты на погрузку и разгрузку, а также на перемещение вагонов от пункта погрузки до пункта разгрузки; эти затраты определены установленными тарифами. Для  $i$ -го заказа затраты на погрузку, перемещение и разгрузку одного вагона обозначены  $c_i$  ( $i = 1, \dots, K$ );
- затраты на перемещение порожних вагонов между пунктами  $\gamma$  и  $\lambda$  железнодорожной сети определены действующими тарифами; затраты на один вагон обозначены  $\Delta c_{\gamma\lambda}$ , где  $\gamma, \lambda = 1, \dots, N$ ;
- штрафы за недопоставку груза на интервале планирования: недопоставка одного груженого вагона  $i$ -го заказа обозначена  $\beta_i$ ;
- штрафы за превышение срока  $t_i^{(k)}$  выполнения  $i$ -го заказа; штраф за опоздание поставки одного вагона  $i$ -го заказа на одни сутки обозначены через  $\alpha_i$ ;
- плата за нахождение вагона  $\mu$ -го типа (простой вагона) на  $j$ -й станции в течение суток обозначается  $\varphi_j^{(\mu)}$ , где  $\mu$  — номер типа вагона.

Математическая постановка задачи построения оптимального календарного плана перемещения грузовых вагонов имеет следующий вид [7]:

$$\min_{\substack{x_i^{(\mu d)}, y_{\lambda\gamma}^{(\mu d)} \\ \mu=1, \dots, M, \\ d=1, \dots, D, \\ i=1, \dots, K \\ \lambda, \gamma=1, \dots, N}} \left( \sum_{i=1}^K x_i c_i + \sum_{\lambda=1}^N \sum_{\gamma=1}^N \Delta c_{\lambda\gamma} \sum_{\mu=1}^M \sum_{d=1}^D y_{\lambda\gamma}^{(\mu d)} + \sum_{i=1}^K \beta_i (x_i^* - x_i) + \sum_{i=1}^K \alpha_i \sum_{\mu \in M(i)} \sum_{d=t_i^{(K)} - \tau + 1}^D x_i^{(\mu d)} (d - t_i^{(K)} + \tau) + \sum_{j=1}^N \sum_{\mu=1}^M \varphi_j^{(\mu)} \sum_{d=1}^D X_j^{(\mu d)} \right) \quad (1)$$

В (1) слагаемые представляют соответственно затраты на доставку груженых вагонов, затраты на переработку порожних вагонов, штрафы за недопоставку, штрафы за превышение сроков доставки, затраты на простой вагонов. Кроме того, в (1):

- $x_i$  — общее число доставленных вагонов  $i$ -го заказа:

$$x_i = \sum_{d=1}^D \sum_{\mu \in M_i} x_i^{(\mu d)} \quad (2)$$

- $X_j^{(\mu d)}$  — число вагонов  $\mu$ -ого типа, находящихся на  $j$ -ой станции в сутки  $d$ :

$$X_j^{(\mu d)} = X_j^{(\mu)} + \sum_{i=1}^K \sum_{d^*=\tau_i+1}^d x_i^{(\mu, d^* - \tau_i)} - \sum_{i \in G_{Kj}} \sum_{d^*=1}^d x_i^{(\mu d^*)} - \sum_{d^*=1}^d \sum_{\lambda=1}^N y_{j\lambda}^{(\mu, d^*)} + \sum_{\gamma=1}^N \sum_{d^*=1+\Delta\tau_i}^d y_{\gamma j}^{(\mu, d^* - \Delta\tau_i)} \quad (3)$$

В (3) первое слагаемое — число вагонов типа  $\mu$  на станции  $j$  в начальный момент планового периода; второе — число вагонов типа  $\mu$ , разгрузка которых имела место на станции  $j$  с начала периода планирования до момента  $d$ ; третье — число вагонов типа  $\mu$ , погрузка которых имела место на станции  $j$  с начала периода планирования до момента  $d$  ( $G_{Kj}$  — множество заказов с погрузкой на станции  $j$ ); четвертое — число порожних вагонов типа  $\mu$ , отправленных со станции  $j$  на другие станции с начала периода планирования до момента  $d$ ; пятое — число порожних вагонов типа  $\mu$ , поступивших на станцию  $j$  с начала периода планирования до момента  $d$ .

При решении задачи должны быть приняты следующие ограничения:

$$X_j^{(\mu d)} \geq 0; \quad (4)$$

$$x_i^{(\mu d)} \geq 0; y_{\lambda\gamma}^{(\mu d)} \geq 0; x_i^* \geq x_i \geq 0 \quad (5)$$

при  $j = 1, \dots, N$ ;  $\mu = 1, \dots, M$ ;  $d = 1, \dots, D$ ;

$i = 1, \dots, K$ ;  $\lambda, \gamma = 1, \dots, N$ .

Задача, определяемая выражениями (1)–(5), относится к задаче линейного программирования. Размерность  $R$  задачи равна

$$R = K \times M \times D + N(N - 1) \times M \times D = MD(K + N(N - 1)).$$

В реальных задачах  $R$  может принимать значения свыше нескольких миллионов.

Приведенная задача имеет множество частых постановок, связанных с формулировкой критерия (отсутствие некоторых составляющих затрат), а также с наличием дополнительных ограничений (например, ограничений числа вагонов, погружаемых или разгружаемых в сутки на каждой станции). В процессе диалога пользователя с чат-ботом соответствующие уточнения могут быть внесены в постановку задачи.

Отметим, что приведенная выше математическая модель задачи календарного планирования использования вагонного парка операторской компании с незначительными изменениями может быть применена и для получения ответов на следующие вопросы:

- какие заказы из заданного множества следует принять к исполнению;
- оценка наименьших затрат на реализацию заданного множества заказов;
- подбор вагонов для реализации принятых заказов из множества вагонов, принадлежащих другим собственникам;
- оценка необходимого объема вагонного парка для реализации с наименьшими затратами прогнозируемого объема предстоящих перевозок.

Интерес представляет и использование модели планирования при оперативном управлении перевозками (рис. 2). В этом случае задачу календарного планирования использования вагонного парка следует решать неоднократно (или периодически) в ходе реализации первоначально найденного плана при отклонении фактического состояния процесса перевозки от планового, появлении возможности принятия дополнительных заказов, изменении временных затрат на реализацию операций перевозочного процесса.

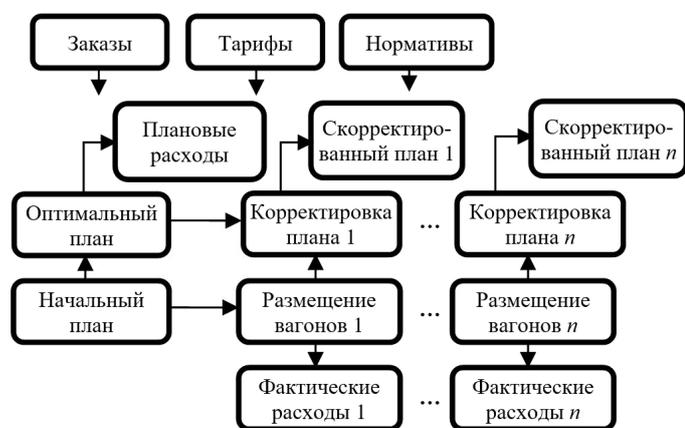


Рис. 2. Схема применения сервиса при оперативном управлении перевозками

Повторные расчеты должны быть обеспечены актуальной информацией о ходе выполнения перевозок, а также уточненной информацией о временных затратах на выполнение предстоящих операций. Очевидно, что повторные расчеты по поиску уточненных планов использования вагонного парка следует осуществлять за ограниченное время. Применение этой процедуры позволит обеспечить наименьшие (или близкие к наименьшим) затраты на выполнение принятых заказов при изменении условий перевозок, не учитываемых первоначальным планом.

#### НАПРАВЛЕНИЯ РАЗВИТИЯ ФУНКЦИОНАЛЬНОСТИ СЕРВИСА

Можно выделить следующие основные направления совершенствования сервиса поддержки принятия решений:

1. Расширение числа типов математических моделей, используемых для поиска решений. Например, для описанной выше задачи планирования использования вагонного парка целесообразно включить в рассмотрение и модели с нелинейными зависимостями штрафов от времени простоя на станции, а также ввести ограничения на пропускную способность станций, осуществляющих операции по погрузке и разгрузке вагонов.
2. Создание аналитической подсистемы [8], предназначенной для выявления причин отказа пользователей от предлагаемых услуг.
3. Реализовать хранение вариантов постановок задач конкретных пользователей для сокращения временных затрат на диалог с чат-ботом при повторном обращении к сервису с задачей, аналогичной предыдущей.

#### ЗАКЛЮЧЕНИЕ

По представленным выше материалам можно сделать следующие выводы:

1. Для повышения качества функционирования транспортного комплекса, характерными особенностями которого является наличие множества перевозочных компаний, типовых задач управления перевозками, общей инфраструктуры и единой информационной среды, эффективным средством может явиться создание совокупности общедоступных сервисов поддержки принятия решений.
2. Необходимым элементом функциональности сервисов поддержки принятия решений является уточнение постановки решаемой задачи в режиме диалога пользователя с чат-ботом.
3. Желаемый эффект сервиса может быть получен лишь при условии адекватности используемой модели и существующего процесса перевозки, что в общем случае приводит к необходимости наличия в сервисе ряда моделей с возможностью выбора в процессе диалога пользователя с чат-ботом.
4. Для поддержки актуальности сервиса необходимо наличие в его составе функций аналитической подсистемы, позволяющей осуществлять мониторинг результатов обращений к сервису.

#### ЛИТЕРАТУРА

1. Зябиров, Х. Ш. Компьютерная поддержка принятия решений в транспортных системах: Учебное пособие для студентов вузов железнодорожного транспорта / Х. Ш. Зябиров, А. В. Кутыркин, Б. А. Лёвин. — Москва: Академкнига, 2010. — 520 с.

2. Информационные технологии на магистральном транспорте: Учебник для студентов вузов железнодорожного транспорта / В. Н. Морозов, Э. К. Лецкий, И. Н. Шапкин, [и др.]. — Москва: Учебно-методический центр по образованию на железнодорожном транспорте, 2018. — 405 с. — (Высшее образование. Учебник для специалистов).

3. Дергачев, А. М. Проблемы эффективного использования сетевых сервисов // Научно-технический вестник Санкт-Петербургского государственного университета информационных технологий, механики и оптики. 2011. № 1 (71). С. 83–87.

4. Рассел, С. Искусственный интеллект: современный подход. Второе издание = Artificial Intelligence: A Modern Approach. Second Edition / С. Рассел, П. Норвиг; перевод с англ. и редакция К. А. Птицына. — Москва: Издательский дом «Вильямс», 2018. — 1408 с.

5. Джанарсанам, С. Разработка чат-ботов и разговорных интерфейсов: чат-боты и голосовые пользовательские интерфейсы на платформах Chatfuel, Dialogflow, Microsoft Bot Framework, Twilio и Alexa Skills = Hands-On Chatbots and Conversational UI Development:

Build chatbots and voice user interfaces with Chatfuel, Dialogflow, Microsoft Bot Framework, Twilio, and Alexa Skills / пер. с англ. М. Райтман. — Москва: ДМК Пресс, 2019. — 340 с.

6. Тиханычев, О. В. Автоматизация поддержки принятия решений: научно-теоретический труд. — Москва: Эдитус, 2015. — 94 с.

7. Лецкий, Э. К. Цифровые сервисы интеллектуальной поддержки принятия решений при управлении грузовыми перевозками на железнодорожном транспорте / Э. К. Лецкий, А. В. Семин // Транспорт: наука, техника, управление. Научный информационный сборник. 2019. № 9. С. 17–20. DOI: 10.36535/0236-1914-2019-09-3.

8. Белов, В. С. Информационно-аналитические системы. Основы проектирования и применения: Учебное пособие. Руководство по изучению курса. Практикум / В. С. Белов; Международный консорциум «Электронный университет», Московский гос. ун-т экономики, статистики и информатики, Евразийский открытый институт. — Москва: Московский гос. ун-т экономики, статистики и информатики, 2004. — 116 с.

# Optimal Planning of Operator Company Wagons Usage

V. A. Varfolomeev, Grand PhD E. K. Letsky,  
Russian University of Transport  
Moscow, Russia  
vvictor@mail.ru, letzky@mail.ru

**Abstract.** The article describes the concept of creating a center for intelligent decision-making support in the management of a transport complex. It is proposed to carry out transportation planning and management services in various transport segments based on web service technologies. In this case, the users of the services will be the carrier companies.

The rationale for creating such a center are:

- the presence in each segment of transport of typical tasks of planning and operational management of transportation;
- the computational complexity of these tasks, which requires the use of high-performance computing tools to solve them.

The center's tasks are to support existing services and develop new ones. The article discusses a typical architecture of a decision support service, which provides for the possibility of specifying a problem statement in a dialogue between a user and a chat bot. As an example of a service, the mathematical formulation of the problem of scheduling the movement of the carriages of an operator company when fulfilling a given set of orders for the transportation of goods is considered.

**Keywords:** decision support, web service, chat bot, mathematical model, transport processes management.

## REFERENCES

1. Zyabirov Kh. Sh., Kutyrkin A. V., Levin B. A. Computer support for decision-making in transport systems: A study guide for students of railway transport universities [Komp'yuternaya podderzhka prinyatiya resheniy v transportnykh sistemakh: Uchebnoe posobie dlya studentov vuzov zheleznodorozhnogo transporta]. Moscow, Akademkniga Publishing House, 2010, 520 p.
2. Morozov V. N., Letsky E. K., Shapkin I. N., et al. Information technologies on mainline transport: A textbook for students of railway transport universities [Informatsionnye tekhnologii na magistral'nom transporte: Uchebnoe posobie dlya studentov vuzov zheleznodorozhnogo transporta]. Moscow, Training and Methodological Center of Railway Transport, 2018, 405 p.
3. Dergachev A. M. Problems of Efficient Use of Network Services [Problemy effektivnogo ispol'zovaniya setevykh servisov], *Scientific and Technical Journal of Information Technologies, Mechanics and Optics* [Nauchno-tekhnicheskii vestnik Sankt-Peterburgskogo gosudarstvennogo universiteta informatsionnykh tekhnologiy, mekhaniki i optiki], 2011, No. 1 (71), Pp. 83–87.

4. Russell S. J., Norvig P. Artificial Intelligence: A Modern Approach. Second Edition [Iskusstvennyy intellekt: sovremennyy podkhod. Vtoroe izdanie]. Moscow, Williams Publishing House, 2018, 1408 p.

5. Janarthanam S. Hands-On Chatbots and Conversational UI Development: Build chatbots and voice user interfaces with Chatfuel, Dialogflow, Microsoft Bot Framework, Twilio, and Alexa Skills [Razrabotka chat-botov i razgovornykh interfeysov: chat-boty i golosovye pol'zovatel'skie interfeysy na platformakh Chatfuel, Dialogflow, Microsoft Bot Framework, Twilio i Alexa Skills]. Moscow, DMK Press, 2019, 340 p.

6. Tikhanychev O. V. Automation of decision support: Scientific and theoretical work [Avtomatizatsiya podderzhki prinyatiya resheniy: nauchno-teoreticheskiy trud]. Moscow, Editus Publishing House, 2015, 94 p.

7. Letsky E. K., Semin A. V. Intelligent Decision Support Services for Freight Transport Management [Tsifrovyye servisy intellektual'noy podderzhki prinyatiya resheniy pri upravlenii gruzovymi perevozkami na zheleznodorozhnom transporte]. *Transport: Science, Equipment, Management. Scientific Information Collection* [Transport: nauka, tekhnika, upravlenie. Nauchnyy informatsionnyy sbornik], 2019, No. 9, Pp. 17–20. DOI: 10.36535/0236-1914-2019-09-3.

8. Belov V. S. Information and analytical systems. Fundamentals of design and application: Training manual. Course Study Guide. Workshop [Informatsionno-analiticheskie sistemy. Osnovy proektirovaniya i primeneniya: Uchebnoe posobie. Rukovodstvo po izucheniyu kursa. Praktikum]. Moscow, Moscow State University of Economics, Statistics and Informatics, 2004, 116 p.

# Алгоритм двухранцевой упаковки с ограниченным межранцевым обменом

д.т.н. А. Г. Басыров, А. В. Калюжный

Военно-космическая академия имени А. Ф. Можайского  
Санкт-Петербург, Россия  
alexanderbas@mail.ru, aleksei.kalyuzhnyi@yandex.ru

д.т.н. А. Д. Хомоненко

Петербургский государственный университет  
путей сообщения Императора Александра I  
Санкт-Петербург, Россия  
khomon@mail.ru

**Аннотация.** Рассматривается NP-полная задача упаковки двух ранцев с возможностью ограниченного обмена между ними упаковываемыми объектами. Приведена вербальная и математическая постановка задачи двухранцевой упаковки объектов с ограниченным межранцевым обменом.

Разработан «жадный» алгоритм, решающий поставленную задачу. Введен показатель качества упаковки объектов по ранцам, характеризующий произведение значения прочности ранцев на долю их объема, заполненного объектами. Приведен пример работы алгоритма и результаты экспериментов, подтверждающие его качество. Схематично представлена упаковка ранцев при нулевом объеме буферов, при объеме буферов, равном 1 и 3. Получена зависимость среднего и нижнего значений показателя качества упаковки, его среднего квадратичного отклонения от размера буфера.

**Ключевые слова:** алгоритм упаковки ранцев, показатель качества, межранцевый обмен, ограниченный буфер, доверительный интервал.

## ВВЕДЕНИЕ

Проблемы планирования информационных процессов возникают во многих прикладных задачах. В системах реального времени требование получения оптимального плана, как правило, отсутствует ввиду высокой временной и ресурсной сложности его синтеза. В статье рассмотрено применение известного приближенного алгоритма, решающего задачу «об упаковке ранца» [1–4], для решения задачи планирования загрузки двух удаленных друг от друга бортовых систем блоками целевой информации с возможностью ограниченного информационного обмена между ними.

Классическая задача об упаковке ранца (рюкзака) заключается в необходимости упаковать некоторый набор объектов разного приоритета и размера в ранец ограниченной вместимости [5, 6]. В статье предложено решение задачи, заключающейся в упаковке двух ранцев двумя соответствующими наборами объектов с возможностью ограниченного обмена объектами между исходными наборами.

Алгоритм решения этой задачи может быть использован в таких прикладных областях, как логистика, планирование, параллельные вычисления, системы связи и др. Задача относится к классу NP-полных [7, 8] и для ее решения в условиях временных ограничений целесообразно использовать приближенные методы или эвристические алгоритмы, которые позволят получить результат, близкий к оптимальному, за приемлемое время. Данной проблеме уделено много внимания [9–14], что подтверждает ее актуальность.

**ПОСТАНОВКА ЗАДАЧИ ПЛАНИРОВАНИЯ УПАКОВКИ ДВУХ РАНЦЕВ С ОГРАНИЧЕНИЕМ НА МЕЖРАНЦЕВЫЙ ОБМЕН**

Дано:

Два удаленных друг от друга ранца вместимостью  $V_1$  и  $V_2$ , прочностью  $\rho_1$  и  $\rho_2$  соответственно.

Множество предназначенных для упаковки в ранцы объектов  $A = \{a_1, a_2, \dots, a_{N_1}, a_{N_1+1}, a_{N_1+2}, \dots, a_{N_1+N_2}\}$ , причем объекты с номерами  $1, \dots, N_1$  расположены вблизи первого ранца, а с номерами  $N_1 + 1, \dots, N_1 + N_2$  — вблизи второго.

Каждый  $i$ -й объект имеет размер  $v_i$  и приоритет  $\pi_i$ ,  $i = 1, \dots, N_1 + N_2$ .

Имеется возможность переместить часть объектов, расположенных вблизи первого ранца (с номерами  $1, \dots, N_1$ ) общим объемом не более  $w_{12}$ , во второй ранец, а также наоборот — переместить часть объектов, расположенных вблизи второго ранца (с номерами  $N_1 + 1, \dots, N_1 + N_2$ ), общим объемом не более  $w_{21}$ , в первый ранец.

Найти:

Значения булевых переменных  $x_i \in \{0,1\}$  и  $y_i \in \{0,1\}$ ,  $i = 1, \dots, N_1 + N_2$ , где  $x_i = 1$  при размещении  $i$ -го объекта в первом ранце,  $y_i = 1$  — во втором ранце,  $x_i = 0$  и  $y_i = 0$ , если  $i$ -й объект не размещается, таких, что

$$\sum_{i=1}^{N_1+N_2} \pi_i \times (\rho_1 \times x_i + \rho_2 \times y_i) \rightarrow \max \quad (1)$$

при ограничениях

$$\begin{aligned} x_i + y_i &\leq 1; \\ \sum_{i=1}^{N_1+N_2} x_i \times v_i &\leq V_1; \\ \sum_{i=1}^{N_1+N_2} y_i \times v_i &\leq V_2; \\ \sum_{i=1}^{N_1} y_i \times v_i &\leq w_{12}; \\ \sum_{i=N_1+1}^{N_1+N_2} x_i \times v_i &\leq w_{21}. \end{aligned}$$

Первое ограничение означает, что один и тот же объект не может быть помещен одновременно в два ранца.

Второе ограничение означает, что суммарный размер всех объектов, размещаемых в первом ранце, не может превышать его объем.

Третье ограничение означает, что суммарный размер всех объектов, размещаемых во втором ранце, не может превышать его объем.

Четвертое ограничение означает, что суммарный размер всех объектов множества  $A$ , перемещаемых для размещения во втором ранце, не может превышать установленный объем  $w_{12}$ .

Пятое ограничение означает, что суммарный размер всех объектов множества  $A$ , перемещаемых для размещения в первом ранце, не может превышать установленный объем  $w_{21}$ .

Ниже представлен относящийся к классу «жадных» разработанный алгоритм решения поставленной задачи, который на каждом шаге пытается упаковать в ранец наиболее приоритетный объект.

АЛГОРИТМ ДВУХРАНЦЕВОЙ УПАКОВКИ ОБЪЕКТОВ С ОГРАНИЧЕННЫМ МЕЖРАНЦЕВЫМ ОБМЕНОМ

В настоящее время существует ряд приближенных алгоритмов, решающих задачу упаковки. К наиболее известным можно отнести:

- первый подходящий — FF (First Fit);
- наилучший подходящий — BF (Best Fit);
- первый подходящий с убыванием — FFD (First Fit Decreasing);
- наилучший подходящий с убыванием — BFD (Best Fit Decreasing).

В статьях [15, 16] представлены доказательства того, что асимптотический коэффициент приближения алгоритмов упаковки FF и BF к оптимальному равен 1,7, а в статье [17] доказывается что данный коэффициент для алгоритмов FFD и BFD не превышает значения 1,22.

Разработанный алгоритм можно отнести к алгоритму FFD, но с добавлением возможности обмена объектами между ранцами через ограниченный по размеру буфер. Описание работы алгоритма приведено в таблице 1.

Таблица 1

Описание работы алгоритма

Шаги	Действия
1.	Начало.
2.	Создать множество $U = \{u_i\}, i = 1, \dots, N_1+N_2$ , где $u_i = 1$ , если $i \leq N_1$ , $u_i = 2$ – в противном случае.
3.	Перенумеровать элементы множества $A$ и соответствующие им элементы множества $U$ в порядке невозрастания значений $\pi_i/v_i, i = 1, \dots, N_1+N_2$
4.	Присвоить номер 1 ранцу с наибольшим значением $\rho_i \times V_i$
5.	$x_i := 0, y_i := 0$ для всех $i = 1, \dots, N_1+N_2$ . Установить в ноль счетчики: $i$ — количества объектов; $c_1$ — занятого объема первого ранца; $c_2$ — занятого объема второго ранца; $c_{12}$ — объема переданных объектов от первого ранца ко второму; $c_{21}$ — объема переданных объектов от второго ранца к первому.
6.	$i := i + 1$
7.	Если $i > N_1+N_2$ , то переход на шаг 12.

8.	Если $u_i = 1$ и $c_1 + v_i \leq V_1$ то $x_i := 1, c_1 := c_1 + v_i$ . Переход на шаг 6.
9.	Если $u_i = 1$ , и $c_1 + v_i > V_1$ , и $c_{12} + v_i \leq w_{12}$ , и $V_2 \geq c_2 + v_i$ , то $y_i := 1, c_2 := c_2 + v_i, c_{12} := c_{12} + v_i$ . Переход на шаг 6.
10.	Если $u_i = 2$ и $c_2 + v_i \leq V_2$ , то $y_i := 1, c_2 := c_2 + v_i$ . Переход на шаг 6.
11.	Если $u_i = 2$ , и $c_2 + v_i > V_2$ , и $c_{21} + v_i \leq w_{21}$ , и $V_1 \geq c_1 + v_i$ , то, $x_i := 1, c_1 := c_1 + v_i, c_{21} := c_{21} + v_i$ . Переход на шаг 6.
12.	Конец.

Если рассмотреть вышеописанную постановку задачи с учетом отсутствия приоритетности объектов, то целевая функция (1) примет вид

$$\sum_{i=1}^{N_1+N_2} v_i(\rho_1 x_i + \rho_2 y_i) \rightarrow \max, \quad (2)$$

а упорядочивать объекты множества  $A$  и соответствующие им элементы множества  $U$  на шаге 2 алгоритма необходимо в порядке неубывания значений  $v_i, i = 1, \dots, N_1+N_2$ .

Показателем качества  $\xi$  распределения объектов без учета их приоритетов по ранцам будем считать нормированную в диапазоне  $[0,1]$  величину

$$\xi = \sum_{i=1}^{N_1+N_2} v_i \left( \frac{\rho_1}{V_1} x_i + \frac{\rho_2}{V_2} y_i \right), \quad (3)$$

характеризующую произведение значения прочности ранцев на долю их объема, заполненного объектами.

Рассмотрим работу алгоритма на примере.

Пусть дано два ранца с объемами  $V_1 = 10$  и  $V_2 = 10$  и прочностями  $\rho_1 = 0,8$  и  $\rho_2 = 0,2$ . Вблизи первого ранца находятся объекты с размерами 3, 4 и 5, вблизи второго ранца — объекты с размерами 1, 2, 4, 5. Положим, что приоритеты объектов одинаковые (рис. 1).

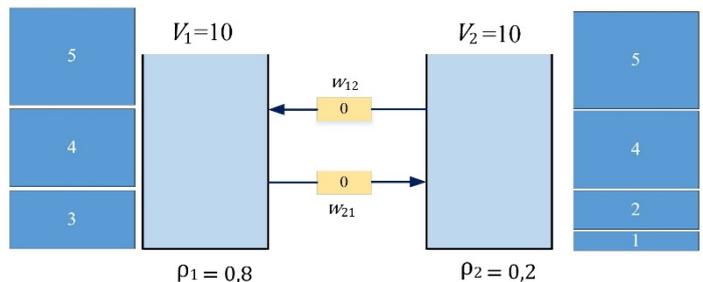


Рис. 1. Исходные данные для примера

При отсутствии буфера между ранцами, рассмотренный алгоритмом объекты упакуются так, как указано на рисунке 2.

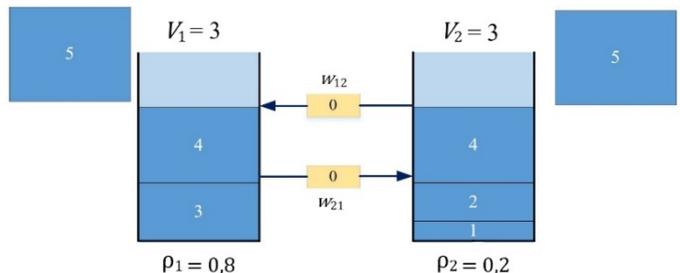


Рис. 2. Упаковка ранцев при нулевом объеме буферов

Сначала в ранцы складываются объекты с наибольшим значением удельного приоритета  $\pi_i/v_i$ , а так как приоритеты одинаковые, то удельный приоритет будет зависеть только от объема объектов — чем меньше объем объекта, тем больше его удельный приоритет. Рассчитаем показатель качества распределения объектов без учета их приоритетов согласно (3).

При отсутствии обмена между ранцами, то есть если  $w_{12}$  и  $w_{21}$  равны 0, то  $\xi_0 = 0,7$ .

Теперь увеличим объем буферов с 0 до 1. Объект размером 1, находящийся вблизи второго ранца, будет передан в первый ранец, так как первый ранец имеет большую прочность (рис. 3). Показатель качества при этом составит  $\xi_1 = 0,76$ .

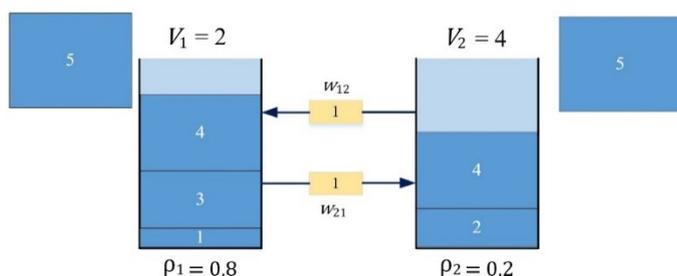


Рис. 3. Упаковка ранцев при объеме буферов, равном 1

Увеличим объем буферов до 3. Теперь объекты с размерами 1 и 2, находящиеся вблизи второго ранца, будут переданы в первый ранец (рис. 4). Показатель качества увеличится до  $\xi_3 = 0,98$ .

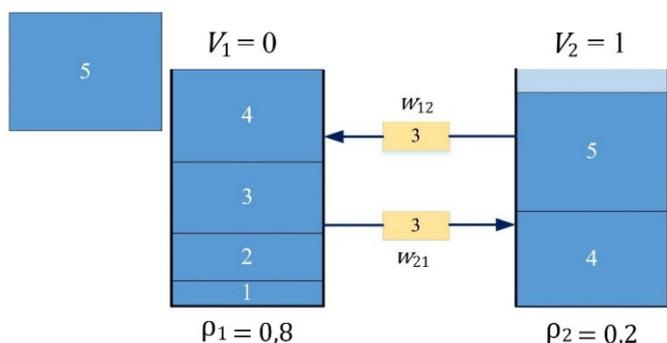


Рис. 4. Упаковка ранцев при объеме буферов, равном 3

Проведя анализ полученных данных, можно сделать вывод, что даже при небольшом увеличении объема буфера между ранцами более «прочный» ранец заполняется быстрее вследствие чего показатель качества существенно растет.

Для проверки качества разработанного алгоритма проведен ряд экспериментов, исходными данными для которых являлись следующие.

Размеры ранцев  $V_1 = 100$  и  $V_2 = 100$ , прочности  $\rho_1 = 0,8$  и  $\rho_2 = 0,2$ , количество объектов рядом с каждым из ранцев составляет 15, объемы объектов равномерно распределены в интервале от 20 до 50 и выбираются случайным образом, приоритет объектов равен 1.

Для каждого  $10^6$  экспериментов менялся размер буферов между ранцами от 0 до 100 с шагом 10. Зависимость среднего и нижнего значений показателя качества упаковки (3), его среднего квадратичного отклонения (СКО) от размера буфера приведена в таблице 2.

Таблица 2

Изменение среднего и нижнего значения показателя качества, СКО при изменении размеров буферов между ранцами

Размеры буферов	Среднее значение показателя качества	Нижнее значение показателя качества	СКО
0	0,818	0,530	0,096
10	0,819	0,533	0,095
20	0,816	0,534	0,093
30	0,812	0,536	0,091
40	0,808	0,538	0,090
50	0,813	0,534	0,089
60	0,819	0,546	0,085
70	0,824	0,558	0,083
80	0,827	0,565	0,083
90	0,825	0,584	0,082
100	0,828	0,622	0,081

Средние и нижние значения показателя качества (3) для различных значений размера буферов обмена приведены в таблице 2. Доверительный интервал для среднего значения показателя качества (3) с уровнем доверия 95 % по каждой строке таблицы не превышает 0,2 % его среднего значения, что свидетельствует о высокой надежности полученной оценки. Нижнее значение показателя качества (3), увеличивается на 15 %, а СКО уменьшается на 16 % с увеличением буферов обмена, что в свою очередь говорит о влиянии размеров буферов на показатель качества распределения объектов по ранцам.

#### ЗАКЛЮЧЕНИЕ

Разработан алгоритм, решающий задачу двухранцевой упаковки с возможностью ограниченного межранцевого обмена. Приведен пример работы алгоритма, а также проведены эксперименты, показывающие, что доверительный интервал не превышает 0,2 % среднего значения показателя качества. Нижнее значение показателя качества увеличивается на 15 %, а СКО уменьшается на 16 % при увеличении буфера обмена. Представленный алгоритм может быть использован для решения широкого круга задач от обмена данными в телекоммуникационных системах до организации доставки товаров транспортными компаниями.

Разработанный алгоритм, в отличие от известных, учитывает «надежность» ранцев и возможность обмена между ними упаковываемыми объектами с учетом размеров буферов обмена.

#### ЛИТЕРАТУРА

1. Гудман, С. Введение в разработку и анализ алгоритмов = Introduction to the design and analysis of algorithms / С. Гудман, С. Хидетниemi; пер. с англ. под ред. В. В. Мартынюка. — Москва: Мир, 1981. — 368 с.
2. Fisher, M. L. Worst-Case Analysis of Heuristic Algorithms // Managment Science. 1980. Vol. 26, No. 1. Pp. 1–17. DOI: 10.1287/mnsc.26.1.1.
3. Coffman, Jr., E. G. Approximation Algorithms for Bin-Packing — An Updated Survey / E. G. Coffman, Jr., M. R. Garey, D. S. Johnson // Algorithm Design for Computer System

Design / G. Ausiello, [et al.] (eds). — Wien: Springer-Verlag, 1984. — Pp. 49–106. — (International Centre for Mechanical Sciences: Courses and Lectures, Vol. 284).

DOI: 10.1007/978-3-7091-4338-4\_3.

4. Johnson, D. S. Fast Algorithms for Bin Packing // *Journal of Computer and System Sciences*. 1974. Vol. 8, Is. 3. Pp. 272–314. DOI: 10.1016/S0022-0000(74)80026-7.

5. Dyckhoff, H. Cutting and Packing in Production and Distribution: A Typology and Bibliography / H. Dyckhoff, U. Finke; with support U. Viethe. — Heidelberg: Physica-Verlag, 1992. — 257 p. — (Contributions to Management Science). DOI: 10.1007/978-3-642-58165-6.

6. Левин, М. Ш. Упаковка в контейнеры (перспективные модели, примеры) // *Информационные процессы*. 2017. Т. 17, № 1. С. 43–60.

7. Karp, R. M. Reducibility Among Combinatorial Problems // *Proceedings of a Symposium on the Complexity of Computer Computations (New York, USA, 20–22 March 1972)* / R. E. Miller, [et al.] (eds). — New York: Plenum Press, 1972. — Pp. 85–103. — (The IBM Research Symposia Series). DOI: 10.1007/978-1-4684-2001-2\_9.

8. Garey, M. R. Computers and Intractability: A Guide to the Theory of NP-Completeness / M. R. Garey, D. S. Johnson. — New York: W. H. Freeman and Co., 1979. — 351 p.

9. Coffman, Jr., E. G. Bin Packing: Maximizing the Number of Pieces Packed / E. G. Coffman, Jr., J. Y.-T. Leung, D. W. Ting // *Acta Infomatica*. 1978. Vol. 9, Is. 3. Pp. 263–271. DOI: 10.1007/BF00288885.

10. Coffman, Jr., E. G. Combinatorial Analysis of an Efficient Algorithm for Processor and Storage Allocation / E. G. Coffman, Jr., J. Y.-T. Leung // *SIAM Journal on Computing*. 1979. Vol. 8, Is. 2. Pp. 202–217. DOI: 10.1137/0208016.

11. Фуремс, Е. М. Обратная задача об упаковке в контейнеры при наличии качественных критериев — постановка и обзор применимых методов // *Искусственный интеллект и принятие решений*. 2016. № 3. С. 31–43.

12. Фуремс, Е. М. Приближенное решение обратной задачи об упаковке в контейнеры с учетом предпочтений лица, принимающего решения // *Искусственный интеллект и принятие решений*. 2018. № 3. С. 112–121. DOI: 10.14357/20718594180321.

13. Wäscher, G. An Improved Typology of Cutting and Packing Problems / G. Wäscher, H. Haußner, H. Schumann // *European Journal of Operational Research*. 2007. Vol. 183, Is. 3. Pp. 1109–1130. DOI: 10.1016/j.ejor.2005.12.047.

14. Chung, F. R. K. On Packing Two-Dimensional Bins / F. R. K. Chung, M. R. Garey, D. S. Johnson // *SIAM Journal on Algebraic Discrete Methods*. 1982. Vol. 3, Is. 1. Pp. 66–76. DOI: 10.1137/0603007.

15. Ullman, J. D. The Performance of a Memory Allocation Algorithm // *Technical Report by Computer Sciences Laboratory of Princeton University*, Vol. 100. — Princeton (NJ): Princeton University, 1971. — 17 p.

16. Garey, M. R. Worstcase analysis of memory allocation algorithms / M. R. Garey, R. L. Graham, J. D. Ullman // *Proceedings of the Fourth Annual ACM Symposium on Theory of Computing (STOC '72) (Denver, CO, USA, 01–03 May 1972)*. — New York (NY): Association for Computing Machinery, 1972. — Pp. 143–150. DOI: 10.1145/800152.804907.

17. Coffman, Jr., E. G. An Application of Bin-Packing to Multiprocessor Scheduling / E. G. Coffman, Jr., M. R. Garey, D. S. Johnson // *SIAM Journal on Computing*. 1978. Vol. 7, Is. 1. Pp. 1–17. DOI: 10.1137/0207001.

# The Double-Packing Algorithm with Limited Inter-Package Exchange

Grand PhD A. G. Basyrov, A. V. Kalyuzhnyi

Mozhaisky Military Space Academy

Saint Petersburg, Russia

alexanderbas@mail.ru, aleksei.kalyuzhnyi@yandex.ru

Grand PhD A. D. Khomonenko

Emperor Alexander I St. Petersburg State

Transport University

Saint Petersburg, Russia

khomon@mail.ru

**Abstract.** The NP-complete problem of packing two satchels with the possibility of limited exchange of packaged objects between them is considered. The verbal and mathematical formulation of the problem of two-ring packing of objects with limited inter-ring exchange is given.

A greedy algorithm has been developed to solve the problem. An indicator of the quality of packing objects by satchels has been introduced. An example of the algorithm's operation and experimental results confirming its quality are given. Packing of knapsacks is schematically presented with zero volume of buffers, with the volume of buffers equal to 1 and 3. The dependence of the average and lower values of the packing quality indicator, its standard deviation from the size of the buffer is obtained.

**Keywords:** knapsacks packing algorithm, quality indicator, exchange between two knapsacks, limited clipboard, confidence interval.

## REFERENCES

1. Goodman S. E., Hedetniemi S. T. Introduction to the design and analysis of algorithms [Vvedenie v razrabotku i analiz algoritmov]. Moscow, Mir Publishers, 1981, 368 p.
2. Fisher M. L. Worst-Case Analysis of Heuristic Algorithms, *Management Science*, 1980, Vol. 26, No. 1, Pp. 1–17. DOI: 10.1287/mnsc.26.1.1.
3. Coffman, Jr. E. G., Garey M. R., Johnson D. S. Approximation Algorithms for Bin-Packing — An Updated Survey. In: Ausiello G., et al. (eds) *Algorithm Design for Computer System Design*. Wien, Springer-Verlag, 1984, Pp. 49–106. DOI: 10.1007/978-3-7091-4338-4\_3.
4. Johnson D. S. Fast Algorithms for Bin Packing, *Journal of Computer and System Sciences*, 1974, Vol. 8, Is. 3, Pp. 272–314. DOI: 10.1016/S0022-0000(74)80026-7.
5. Dyckhoff H., Finke U., Viethe U. Cutting and Packing in Production and Distribution: A Typology and Bibliography. Heidelberg, Physica-Verlag, 1992, 257 p. DOI: 10.1007/978-3-642-58165-6.
6. Levin M. Sh. Bin Packing Problem (prospective models, examples) [Upakovka v konteynery (perspektivnye modeli, primery)], *Information Processes [Informatsionnye protsessy]*, 2017, Vol. 17, No. 1, Pp. 43–60.
7. Karp R. M. Reducibility Among Combinatorial Problems. In: Miller R. E., et al. (eds) *Proceedings of a Symposium on the Complexity of Computer Computations*, New York, USA, March 20–22, 1972. New York, Plenum Press, 1972, Pp. 85–103. DOI: 10.1007/978-1-4684-2001-2\_9.
8. Garey M. R., Johnson D. S. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, W. H. Freeman and Co., 1979, 351 p.
9. Coffman, Jr. E. G., Leung J. Y.-T., Ting D. W. Bin Packing: Maximizing the Number of Pieces Packed, *Acta Informatica*, 1978, Vol. 9, Is. 3, Pp. 263–271. DOI: 10.1007/BF00288885.
10. Coffman, Jr. E. G., Leung J. Y.-T. Combinatorial Analysis of an Efficient Algorithm for Processor and Storage Allocation, *SIAM Journal on Computing*, 1979, Vol. 8, Is. 2, Pp. 202–217. DOI: 10.1137/0208016.
11. Furems E. M. Inverse Bin Packing Problem with Multiple Qualitative Criteria — Formulation and Survey of Applicable Approaches [Obratnaya zadacha ob upakovke v konteynery pri nalichii kachestvennykh kriteriev — postanovka i obzor primenimykh metodov], *Artificial Intelligence and Decision-Making [Iskusstvennyy intellekt i prinyatie resheniy]*, 2016, No. 3, Pp. 31–43.
12. Furems E. M. Approximate Solution Scheme for Inverse Bin-Packing Problem Subject to Decision Making Preferences [Priblizhennoe reshenie obratnoy zadachi ob upakovke v konteynery s uchetom predpochteniy litsa, prinyamayushchego resheniya], *Artificial Intelligence and Decision-Making [Iskusstvennyy intellekt i prinyatie resheniy]*, 2018, No. 3, Pp. 112–121. DOI: 10.14357/20718594180321.
13. Wäscher G., Haußner H., Schumann H. An Improved Typology of Cutting and Packing Problems, *European Journal of Operational Research*, 2007, Vol. 183, Is. 3, Pp. 1109–1130. DOI: 10.1016/j.ejor.2005.12.047.
14. Chung F. R. K., Garey M. R., Johnson D. S. On Packing Two-Dimensional Bins, *SIAM Journal on Algebraic Discrete Methods*, 1982, Vol. 3, Is. 1, Pp. 66–76. DOI: 10.1137/0603007.
15. Ullman J. D. The Performance of a Memory Allocation Algorithm, *Technical Report by Computer Sciences Laboratory of Princeton University, Vol. 100*. Princeton (NJ), Princeton University, 1971, 17 p.
16. Garey M. R., Graham R. L., Ullman J. D. Worstcase analysis of memory allocation algorithms, *Proceedings of the Fourth Annual ACM Symposium on Theory of Computing (STOC '72)*, Denver, CO, USA, May 01–03, 1972. New York (NY), Association for Computing Machinery, 1972, Pp. 143–150. DOI: 10.1145/800152.804907.
17. Coffman, Jr. E. G., Garey M. R., Johnson D. S. An Application of Bin-Packing to Multiprocessor Scheduling, *SIAM Journal on Computing*, 1978, Vol. 7, Is. 1, Pp. 1–17. DOI: 10.1137/0207001.

# Методы анализа больших данных надежности восстанавливаемых систем

к.ф.-м.н. В. А. Проурзин

Институт проблем машиноведения  
Российской академии наук  
Санкт-Петербург, Россия  
proursin@gmail.com

к.т.н. О. В. Проурзин

Петербургский государственный университет  
путей сообщения Императора Александра I  
Санкт-Петербург, Россия  
pvo777@yandex.ru

**Аннотация.** Компьютерные системы мониторинга технического состояния транспортных систем накапливают в том числе и большие данные по надежности отдельных устройств и элементов. Это позволяет вычислить показатели надежности систем, не прибегая к дорогостоящим испытаниям. Здесь рассмотрены методы анализа больших данных надежности восстанавливаемых многоканальных систем. Большие данные содержат значения наработки до отказа и значения времени восстановления элементов системы, полученных при мониторинге функционирования аналогичных систем в процессе эксплуатации. Законы распределения отказов и восстановлений элементов системы неизвестны и могут носить произвольный характер. Рассмотрены алгоритмы оценки показателей надежности восстанавливаемых систем с учетом разнообразия, неоднородности и изменчивости данных. В случае монотонных систем с независимым восстановлением элементов оценка коэффициента готовности и средней наработки между отказами системы сводится к оценке средней наработки до отказа и среднего времени восстановления каждого элемента системы для произвольных законов распределений.

**Ключевые слова:** большие данные, сложная система, робастность, коэффициент готовности, средняя наработка между отказами, среднее время восстановления.

## ВВЕДЕНИЕ

Сложные системы с точки зрения теории надежности представляют собой совокупность технических устройств, взаимодействующих в процессе функционирования и связанных между собой. Математическим моделям надежности сложных, многоканальных, кластерных систем посвящено большое количество научных публикаций, например [1–7].

Важным вопросом является оценка характеристик надежности сложной системы в целом: средней наработки на отказ и коэффициента готовности. Известен логико-вероятностный подход [4, 6], основанный на представлении отказов и восстановлений системы как случайных бинарных событий. Отказы и восстановления системы зависят от некоторого числа первичных бинарных случайных событий (отказ и восстановление элементов). К настоящему времени разработан достаточно эффективный аппарат для решения задач такого рода. Для этого требуется информация о законах распределений времени наработки до отказа и времени восстановления каждого элемента.

Традиционно оценка показателей надежности элементов производится в результате испытаний, которые характеризуются высокой стоимостью и требуют больших за-

трат времени. С другой стороны, компьютерный мониторинг эксплуатации уже существующих объектов позволяет собрать огромную базу данных показателей надежности, в частности данных по наработкам до отказа и времени восстановления. Подходы и методы работы с такими огромными базами содержат содержание компьютерных технологий работы с большими данными [8]. Разнообразие, достоверность и изменчивость больших данных составляют основную сложность анализа больших данных. Основная проблема анализа эксплуатационных данных надежности систем состоит, во-первых, в том, что значения получены при различных нагрузках и различных законах распределений отказов и восстановлений, и, во-вторых, в наличии недостоверных и аномальных данных.

Здесь рассматривается тот случай, когда условия эксплуатации систем регламентированы, и в целом их можно считать близкими, а данные по надежности изделий, полученные из различных источников, в среднем будут однородны по условиям эксплуатации. Данные, связанные с нестандартными условиями эксплуатации и другие аномальные данные требуется выявить и исключить в ходе отбраковки при анализе общей выборки. Этот случай рассмотрен при разработке методов оценки показателей надежности восстанавливаемых многоканальных систем. Показатели оцениваются по данным компьютерного мониторинга отказов и восстановлений эксплуатации аналогичных изделий.

Цель настоящей работы заключается в разработке компьютерных методов оценки основных показателей надежности восстанавливаемых многоканальных систем, а именно: коэффициента готовности, средней наработки между отказами и среднего времени восстановления системы. Показатели оцениваются по данным компьютерного мониторинга отказов и восстановлений эксплуатации аналогичных изделий. При этом учитывается разнообразие реальных законов распределения отказов и восстановлений элементов, разнообразие эксплуатационных нагрузок и наличие недостоверных данных.

## ПОСТАНОВКА ЗАДАЧИ

Рассматривается модель системы, состоящей из  $n$  узлов (элементов). Узлы образуют монотонную структурную схему работоспособности системы, например схему с последовательно-параллельным соединением. Отказы и восстановления каждого узла независимы и образуют альтернирующий процесс восстановления с некоторыми функциями распределения.

Пусть в результате мониторинга эксплуатации самой системы или аналогов рассматриваемых узлов получен набор данных о наработках до отказа и длительностях восстановлений каждого элемента системы. Для  $j$ -го элемента заданы  $N_j$  значений наработок до отказа  $t_{ji}$ ,  $i = 1, \dots, N_j$  и  $M_j$  значений длительности восстановлений  $s_{ki}$ ,  $i = 1, \dots, M_j$ ,  $j = 1, \dots, n$ . Законы распределений отказов и восстановлений, а также данные об эксплуатационных нагрузках неизвестны.

Ставится задача оценки коэффициента готовности  $K$  системы, средней наработки между отказами (средней наработки на отказ) системы  $T_C$  и среднего времени восстановления системы  $T_R$ .

МЕТОДЫ РЕШЕНИЯ ЗАДАЧИ

*Отбраковка аномальных данных.* Далее будем исходить из того, что условия эксплуатации рассматриваемых систем регламентированы, и в целом их можно считать близкими. Это позволяет утверждать, что данные по надежности изделий, полученные из различных источников, в среднем будут однородны по условиям эксплуатации. Данные, связанные с нестандартными условиями эксплуатации и другие аномальные данные требуется выявить и исключить в ходе отбраковки при анализе общей выборки.

Рассмотрим задачу оценивания средней наработки до отказа  $T$  некоторого элемента системы по выборке значений его наработок до отказа  $\{t_i\}$ ,  $i = 1, \dots, N$ . В условиях наличия недостоверных данных и «дрейфа» законов распределений, генерирующих данные, оценка параметра положения выборки (среднего значения), произведенная с помощью среднего арифметического, является неустойчивой. Для решения этой проблемы используют процедуры отбраковки аномальных данных и методы робастного оценивания параметра положения выборки [9, 10].

Простейший классический алгоритм отбраковки подозрительного на выброс значения  $t$  элемента выборки называется *правилом трех сигм*. Элемент выборки  $t$  считается аномальным, если выполнено неравенство

$$|t - \bar{t}| > 3s,$$

где  $\bar{t} = \frac{1}{N} \sum_1^N t_i$  — выборочное среднее;

$s = \sqrt{\frac{1}{N} \sum_1^N (t_i - \bar{t})^2}$  — среднеквадратическое отклонение.

Сравнительно новые подходы к решению задачи отбраковки аномальных данных основаны на алгоритмах разведочного анализа данных Тьюки, а именно боксплота Тьюки и его модификаций [10]. Нижний  $t_L$  и верхний  $t_U$  пороги отбраковки в боксплоте Тьюки задаются следующим образом:

$$t_L = \max \left\{ t_{(1)}, LQ - \frac{3}{2} IQR \right\},$$

$$t_U = \min \left\{ t_{(N)}, UQ + \frac{3}{2} IQR \right\}.$$

Здесь  $t_{(1)}$  и  $t_{(N)}$  — экстремальные порядковые статистики выборки ( $k$ -я порядковая статистика  $t_{(k)}$  есть  $k$ -е по порядку значение в упорядоченной по возрастанию исходной выборке),  $IQR = UQ - LQ$  — выборочная интерквартильная ширина,  $LQ = t_{[N/4]}$  и  $UQ = t_{N-[N/4]}$  — выборочные

нижний и верхний квартили. Правило отбраковки имеет вид: значение  $t$  является аномальным, если  $t > t_U$  или  $t < t_L$ .

*Робастные оценки параметра положения.* В статистическом анализе больших данных для обеспечения устойчивости оценки параметра положения используются робастные методы оценки [10]. Робастность — это свойство статистической процедуры быть устойчивой к неконтролируемым отклонениям от принятых моделей распределений данных.

Двухэтапная процедура робастного оценивания состоит в следующем. На первом этапе производится отбраковка выбросов с помощью правила трех сигм или боксплота Тьюки. На втором этапе производится оценивание параметра положения путем вычисления выборочного среднего для оставшихся элементов выборки.

Известны методы оценки параметра положения, устойчивые к наличию выбросов – робастные методы математической статистики. Простейшей из известных робастных оценок параметра положения является выборочная медиана:

$$\hat{T} = \text{med } t = \begin{cases} t_{(k)}, & N = 2k + 1 \\ \frac{t_{(k)} + t_{(k+1)}}{2}, & N = 2k. \end{cases} \quad (1)$$

Известный подход к построению робастных оценок был предложен Хубером, в основе которого лежит минимаксный принцип построения наилучшего решения в наихудшей ситуации. Оценка параметра положения по Хуберу:

$$\hat{T} = \frac{1}{N} \left( (n_2 - n_1)k + \sum_{|t_i - \hat{T}| < k} t_i \right), \quad (2)$$

где  $k$  — величина, которая допускается в качестве отклонения от центра совокупности (например,  $k = 1,5s$ );

$n_1$  — число наблюдений из выборки, лежащих в интервале  $(-\infty; \hat{T} - k)$ ;

$n_2$  — число наблюдений, лежащих в интервале  $(\hat{T} + k; +\infty)$ .

При расчетах по формуле (2) в качестве начальной оценки  $\hat{T}$  может применяться обычная средняя арифметическая или медиана (1). Затем на каждой итерации производится разделение выборочной совокупности на три части и вычисление по формуле (2) до сходимости процедуры.

Использование описанных методов дает нам устойчивые несмещенные оценки среднего значения  $T_j$  наработки до отказа  $j$ -го элемента системы. Аналогично получают оценки среднего времени  $S_j$  восстановления  $j$ -го элемента системы. По этим значениям строится оценка коэффициента готовности  $j$ -го элемента, которая не зависит от вида законов распределения отказов и восстановлений [7]:

$$K_j = \frac{T_j}{T_j + S_j}. \quad (3)$$

*Логическая структурная функция работоспособности системы.* При анализе надежности сложных систем удобно пользоваться структурными схемами работоспособности системы [4, 6, 7]. Например, если отказ системы наступает при отказе хотя бы одного элемента, то такой схемой будет схема последовательного соединения элементов. Если система работоспособна при работоспособности хотя бы одного элемента, то имеем случай параллельного соединения

элементов (схема нагруженного резерва). Можно рассмотреть и более сложные схемы, включающие в себя последовательные и параллельные подсистемы, например мостиковую схему соединения (рис. 1).

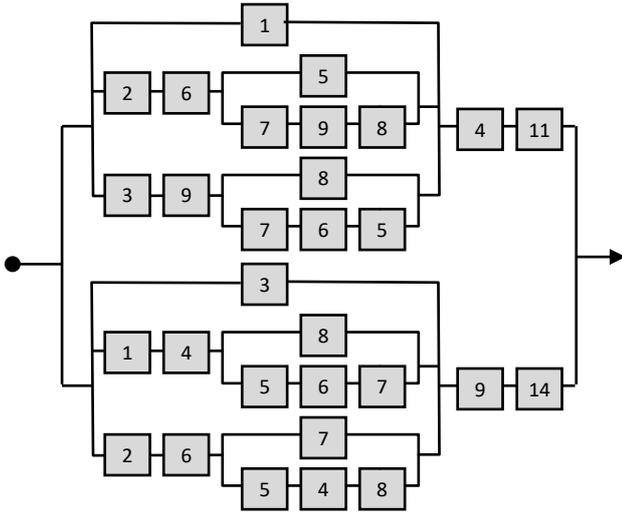


Рис. 1. Структурная схема работоспособности на примере системы электроснабжения

Отказ системы есть случайное событие, описываемое бинарной (булевой) переменной  $X$ , которая принимает одно из двух значений — 0 (отказ) или 1 (работа). Это событие зависит от  $n$  простых независимых событий, описываемых бинарными переменными  $x_j$  (работа или отказ элементов). Вводится структурная функция работоспособности системы, задающая зависимость состояния системы  $X$  от состояний ее элементов:  $X = \varphi(x_1, x_2, \dots, x_n)$ . Например, для схемы  $n$  последовательно соединенных элементов (рис. 2) структурная функция есть произведение всех бинарных переменных  $x_j$ :  $X = x_1 x_2 \dots x_n$ .

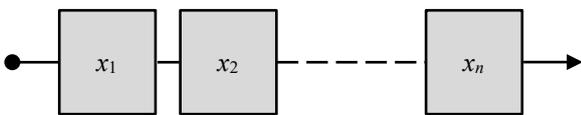


Рис. 2. Структурная схема последовательно соединенных элементов

Для схемы  $n$  параллельно соединенных элементов (рис. 3) структурная функция есть

$$X = 1 - (1 - x_1)(1 - x_2) \dots (1 - x_n).$$

В дальнейшем рассматриваются только системы, структурные функции которых обладают свойством монотонности [7].

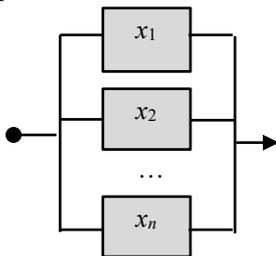


Рис. 3. Структурная схема параллельно соединенных элементов

Известны [7] следующие важные результаты для монотонных систем с независимыми отказами и восстановлением элементов. Пусть для каждого элемента системы известны средние значения  $T_j$  наработки до отказа, средние значения  $S_j$  времени восстановления и соответствующие им значения коэффициента готовности  $K_j$ . Тогда:

1. Коэффициент готовности  $K$  монотонной системы равен значению структурной функции от коэффициентов готовности элементов системы

$$K = \varphi(K_1, K_2, \dots, K_n). \quad (4)$$

2. Средняя наработка между отказами  $T_C$  системы с индивидуальным независимым восстановлением элементов вычисляется по формуле

$$T_C = \frac{K}{\lambda_C} = \frac{\varphi(K_1, K_2, \dots, K_n)}{\lambda_C}, \quad (5)$$

где  $\lambda_C$  — приведенная интенсивность отказов системы:

$$\lambda_C = \sum_{j=1}^n \frac{1}{T_j + S_j} (\varphi|_{K_j=1} - \varphi|_{K_j=0}).$$

3. Среднее время восстановления системы  $T_R$  определяется следующим выражением:

$$T_R = T_C \frac{1 - K}{K} = T_C \frac{1 - \varphi(K_1, K_2, \dots, K_n)}{\varphi(K_1, K_2, \dots, K_n)}. \quad (6)$$

Все приведенные выражения не зависят от вида законов распределения отказов и восстановлений элементов.

*Пример.* Пусть кластерная вычислительная система состоит из трех одинаковых компьютеров, работающих параллельно,  $n = 3$ . Структурная функция системы имеет вид  $X = 1 - (1 - x_1)(1 - x_2)(1 - x_3)$ . Пусть средняя наработка до отказа одного компьютера равна году:  $T_0 = 8760$  ч. Среднее время восстановления после отказа равно календарному месяцу:  $S_0 = 720$  ч.

Используя формулы (3)–(5) получим следующее. Коэффициент готовности каждого компьютера  $K_0 = 8760 / (8760 + 720) = 0,9240506329$ . Коэффициент готовности системы равен  $K = 1 - (1 - K_0)^3 = 0,9995619008$ . Приведенная интенсивность отказов системы:

$$\lambda_C = \frac{3}{T_0 + S_0} (1 - (1 - (1 - K_0)^2)) = 1,8254 \times 10^{-6}.$$

Средняя наработка системы между отказами

$$T_C = K / \lambda_C = 547\,581 \text{ ч.}$$

Среднее время восстановления системы  $T_R = 240$  ч.

#### АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ ОЦЕНКИ ПОКАЗАТЕЛЕЙ НАДЕЖНОСТИ

1. Определение состава элементов, входящих в систему, построение схемы работоспособности и логической структурной функции работоспособности системы.

2. Извлечение из больших данных мониторинга элементов системы и их аналогов значений наработок и времени восстановления этих элементов. Составление начальных выборок  $\{t_{ji}\}$  и  $\{s_{ji}\}$ .

3. Получение робастных оценок параметров положения для каждой выборки: средние значения  $T_j$  наработки до отказа и средние значения  $S_j$  времени восстановления. Для этого используется либо двухэтапная процедура оценивания, либо робастные методы (1), (2).

4. Вычисление коэффициента готовности  $K_j$ ,  $j = 1, \dots, n$  каждого элемента по формуле (3).

5. Вычисление коэффициента готовности  $K$  системы по формуле (4).

6. Вычисление средней наработки между отказами системы  $T_C$  по формуле (5).

7. Вычисление среднего времени восстановления системы  $T_R$  по формуле (6).

#### ЗАКЛЮЧЕНИЕ

Компьютерные методы мониторинга технического состояния сложных систем предоставляют нам данные о надежности этих систем в процессе эксплуатации. Эти данные представляют из себя огромные массивы информации. Анализ и обработка таких массивов составляют содержание науки о больших данных. Здесь рассмотрены алгоритмы оценки основных показателей надежности восстанавливаемых систем в условиях разнообразия реальных законов распределения отказов и восстановлений элементов, разнообразия эксплуатационных нагрузок и наличия недоверенных данных. Приведены алгоритмы оценки коэффициента готовности, средней наработки между отказами и среднего времени восстановления системы по реальным данным эксплуатации элементов системы. Показано, что при этом не требуется оценивать законы распределения отказов и восстановлений элементов.

#### ЛИТЕРАТУРА

1. Shooman, M. L. Reliability of Computer Systems and Networks: Fault Tolerance, Analysis, and Design. — New York: John Wiley & Sons, 2002. — 552 p.

2. Черкесов, Г. Н. Надежность аппаратно-программных комплексов: Учебное пособие. — Санкт-Петербург: Питер, 2005. — 479 с.

3. Гуров, С. В. Анализ надежности технических систем с произвольными законами распределений отказов и восстановлений // Качество и надежность изделий: сборник статей. 1992. № 2 (18). С. 3–37. — (В помощь слушателям лекций Консультационного центра по качеству и надежности).

4. Prourzin, V. A. Techno-Economic Risk in Designing Complex Systems: Algorithms for Analysis and Optimization // Automation and Remote Control. 2003. Vol. 64, No. 7. Pp. 1054–1062. DOI: 10.1023/A:1024773916089.

5. Prourzin, V. A. The Dynamic Reliability Model under Variable Loads and Accelerated Tests // Journal of Machinery Manufacture and Reliability. 2020. Vol. 49, No. 5. Pp. 395–400. DOI: 10.3103/S1052618820050118.

6. Рябинин, И. А. Логико-вероятностные методы исследования надежности структурно-сложных систем / И. А. Рябинин, Г. Н. Черкесов. — Москва: Радио и связь. Редакция литературы по радиоэлектронике, 1981. — 264 с. — (Библиотека инженера по надежности).

7. Байхельт, Ф. Надежность и техническое обслуживание. Математический подход = Zuverlässigkeit und Instandhaltung. Mathematische Methoden / Ф. Байхельт, П. Франкен; перевод с нем. М. Г. Коновалова; под ред. И. А. Ушакова. — Москва: Радио и связь. Редакция переводной литературы, 1988. — 392 с.

8. Лесковец, Ю. Анализ больших наборов данных = Mining of Massive Datasets / Ю. Лесковец, А. Раджараман, Д. Д. Ульман; перевод с англ. А. А. Слинкина. — Москва: ДМК Пресс, 2016. — 498 p.

9. Barnett, V. Outliers in Statistical Data. Third Edition / V. Barnett, T. Lewis. — Chichester: John Wiley & Sons, 1994. — 601 p.

10. Shevlyakov, G. L. Robustness in Data Analysis: Criteria and Methods / G. L. Shevlyakov, N. O. Vilchevski. — Utrecht: VSP Publishers, 2002. — 318 p. — (Modern Probability and Statistics).

# Methods for Big Data Analysis of Reliability of Recoverable Systems

PhD V. A. Prourzin

Institute for Problems in Mechanical Engineering  
of the Russian Academy of Sciences  
Saint Petersburg, Russia  
proursin@gmail.com

PhD O. V. Prourzin

Emperor Alexander I St. Petersburg  
State Transport University  
Saint Petersburg, Russia  
pvo777@yandex.ru

**Abstract.** Computer systems for monitoring the technical condition of transport accumulate, among other things, large data on the reliability of individual devices and elements. This allows us to calculate system reliability metrics without resorting to costly testing. Methods for analyzing big data of reliability of recoverable multichannel systems are considered here. Big data contains values of mean time between failures and values of the recovery time of system elements obtained by monitoring the functioning of similar systems during operation. The distribution laws of failures and restorations of system elements are unknown and can be arbitrary. Algorithms for assessing the reliability indicators of recoverable systems are considered, taking into account the diversity, unreliability and variability of data. In the case of monotonic systems with independent recovery of elements, the estimation of the availability factor and the mean time between failures of the system is reduced to evaluating the mean time to failure and the mean time to recover each element of the system for arbitrary distribution laws.

**Keywords:** big data, complex system, robustness, availability factor, mean time between failures, mean time to recovery.

## REFERENCES

1. Shooman M. L. Reliability of Computer Systems and Networks: Fault Tolerance, Analysis, and Design. New York, John Wiley & Sons, 2002, 552 p.
2. Cherkesov G. N. Reliability of hardware and software systems: Study guide [Nadezhnost' apparatno-programmnykh kompleksov: Uchebnoe posobie]. Saint Petersburg, Piter Publishing House, 2005, 479 p.
3. Gurov S. V. Analysis of The Reliability of Technical Systems with Arbitrary Laws of Distribution of Failures and Restorations [Analiz nadezhnosti tekhnicheskikh sistem s proizvol'nymi zakonami raspredeleniy otkazov i vosstanovleniy], *Quality and Reliability of Rproducts: Collection of Articles [Kachestvo i nadezhnost' izdeliy: sbornik statey]*, 1992. No. 2 (18), Pp. 3–37.
4. Prourzin V. A. Techno-Economic Risk in Designing Complex Systems: Algorithms for Analysis and Optimization, *Automation and Remote Control*, 2003, Vol. 64, No. 7, Pp. 1054–1062. DOI: 10.1023/A:1024773916089.
5. Prourzin V. A. The Dynamic Reliability Model under Variable Loads and Accelerated Tests // *Journal of Machinery Manufacture and Reliability*. 2020. Vol. 49, No. 5. Pp. 395–400. DOI: 10.3103/S1052618820050118.
6. Ryabinin I. A., Cherkesov G. N. Logical-probabilistic methods for studying the reliability of structurally complex systems [Logiko-veroyatnostnye metody issledovaniya nadezhnosti strukturno-slozhnykh sistem]. Moscow, Radio and Communications Publishers, 1981, 264 p.
7. Beichelt F., Franken P. Reliability and maintenance. Mathematical approach [Nadezhnost' i tekhnicheskoe obsluzhivanie. Matematicheskiy podkhod]. Moscow, Radio and Communications Publishers, 1988, 392 p.
8. Leskovec J., Rajaraman A., Ullman J. D. Mining of Massive Datasets [Analiz bol'shikh naborov dannykh]. Moscow, DMK Press, 2016, 498 p.
9. Barnett V., Lewis T. Outliers in Statistical Data. Third Edition. Chichester, John Wiley & Sons, 1994, 601 p.
10. Shevlyakov G. L., Vilchevski N. O. Robustness in Data Analysis: Criteria and Methods. Utrecht, VSP Publishers, 2002, 318 p.

# Модель масштабируемого микросервиса на основе системы массового обслуживания с «охлаждением»

д.т.н. В. А. Лохвицкий, к.т.н. В. А. Гончаренко, Э. С. Левчик  
Военно-космическая академия имени А. Ф. Можайского  
Санкт-Петербург, Россия  
lokhv\_va@mail.ru, vlango@mail.ru

**Аннотация.** Предложена формализация процесса функционирования масштабируемого сервиса с учетом временных затрат на конфигурирование вычислительных узлов на основе многоканальной неэкспоненциальной системы массового обслуживания с «разогревом» и «охлаждением» каналов. Выполнено снижение сложности модели путем представления длительностей «разогрева» и «охлаждения» в виде обобщенной случайной задержки перед обслуживанием. Описаны основные этапы метода расчета вероятностно-временных характеристик системы массового обслуживания. Марковизация немарковских компонент рассматриваемой СМО осуществлена с использованием  $H_2$ -распределения. Получено распределение вероятностей состояний рассматриваемой СМО. Выполнено сопоставление результатов, полученных численным методом, с результатами имитационного моделирования.

**Ключевые слова:** микросервис, система массового обслуживания, «охлаждение» каналов, марковизация, обобщенная задержка обслуживания.

## ВВЕДЕНИЕ

Современные информационные системы разрабатываются и функционируют с использованием широкого спектра технологий. Среди них, как одну из наиболее перспективных, необходимо выделить технологию, базирующуюся на микросервисной архитектуре с контейнерной виртуализацией [1].

Инфраструктура микросервисной информационной системы с контейнерной виртуализацией характеризуется динамическим изменением вычислительной структуры и параметров узлов (рис. 1).

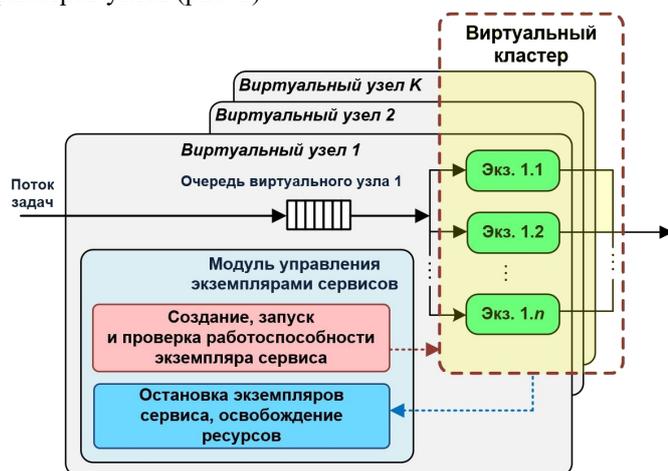


Рис. 1. Представление процесса обработки задач масштабируемым сервисом

В частности, при рассмотрении процессов масштабирования сервисов, реализуемых на основе контейнеризованных приложений, следует учитывать временные затраты на запуск необходимого числа контейнеров, проверку их работоспособности, а также остановку контейнеров, использование которых в данный момент не предполагается [2].

Длительности отдельных операций запуска, остановки контейнеров, проверки работоспособности (с учетом типа запущенного в контейнере приложения-сервиса) представляют собой случайные величины, учет которых необходимо реализовывать в соответствующих моделях.

## ФОРМАЛИЗАЦИЯ ПРОЦЕССА ФУНКЦИОНИРОВАНИЯ МАСШТАБИРУЕМОГО СЕРВИСА

Традиционно для моделирования процессов обработки случайного потока задач используются модели теории очередей. В частности, рассматриваемый нами процесс обработки задач может быть представлен в виде многоканальной системы массового обслуживания (СМО) с дополнительным учетом временных затрат на конфигурирование сервиса.

Указанное конфигурирование может осуществляться в нескольких случаях:

- при запуске сервиса (экземпляров сервиса), в случае если в момент прибытия очередной заявки на обслуживание сервис был остановлен;
- при изменении числа запущенных экземпляров сервиса (масштабирование в сторону увеличения или уменьшения числа экземпляров).

В зависимости от типов распределений, характеризующих случайные процессы прибытия заявок в систему, их обслуживания, масштабирования сервиса с проверкой работоспособности, могут применяться модели, основанные на различных типах СМО. Анализ работ [3–6] показал, что моделирование подобных процессов возможно на основе использования СМО с «разогревом» или «охлаждением» каналов обслуживания. Существенным ограничением данных работ является невозможность одновременного учета в одной модели процессов «разогрева» и «охлаждения» каналов. Принципиальная схема описания искомого процесса представлена на рисунке 2.

В стационарном режиме работы системы обслуживания расположение на временной оси момента наступления указанной задержки (до или после обслуживания) принципиального значения не имеет. Важен именно учет суммарной случайной длительности задержки, которая непо-

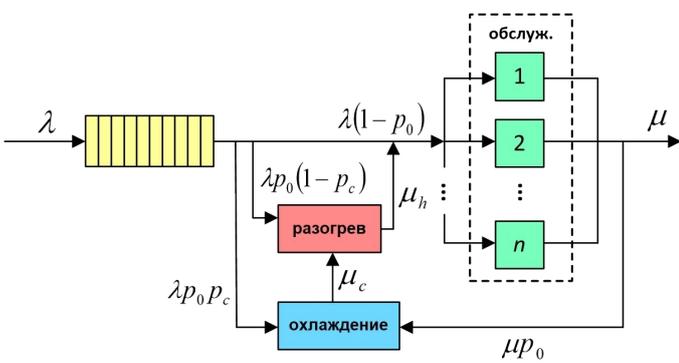


Рис. 2. Представление процесса обработки задач масштабируемым сервисом в виде многоканальной СМО с «разогревом» и/или «охлаждением» каналов

средственно влияет на время пребывания заявки в системе. Здесь необходимо учесть наличие случайных задержек обоих типов, поэтому рассматриваемый процесс возможно представить с помощью модели многоканальной СМО с «охлаждением» общего вида, предложенной в [6].



Рис. 3. Временная диаграмма работы СМО с «разогревом» и «охлаждением» каналов обслуживания

Перечисленные выше условия требуют пояснений.

Во-первых, режим «разогрева» активизируется в случае, если система была пустой (на обслуживании и в очереди не было ни одной заявки) и прибыла очередная заявка. В этом случае обслуживание прибывшей заявки начнется после случайного периода «разогрева» системы. Кроме того, все прибывающие во время «разогрева» системы заявки попадают в очередь и также не обслуживаются до окончания данного периода.

Во-вторых, режим «охлаждения» активизируется после окончания обслуживания последней заявки (обслуживание очередной заявки завершилось и в очереди нет ни одной заявки). Прибывающие во время «охлаждения» системы заявки также попадают в очередь, а их обслуживание может начаться не ранее чем закончится период «охлаждения», а затем и режим «разогрева».

Для этого на первоначальном этапе необходимо найти обобщенную случайную задержку перед обслуживанием (или после него) на основе двух распределений случайных величин — длительности «разогрева» и длительности «охлаждения».

ПРЕДСТАВЛЕНИЕ ПРОЦЕССОВ РАЗОГРЕВА И ОХЛАЖДЕНИЯ В ВИДЕ ОБОБЩЕННОЙ СЛУЧАЙНОЙ ЗАДЕРЖКИ

Для расчета искомой задержки выполним построение временной диаграммы указанного процесса (рис. 3).

При построении диаграммы определим следующие условия:

1. Режим «разогрева» инициируется только событием поступления заявки в пустую систему.
2. Режим «охлаждения» инициируется только событием окончания обработки последней заявки при отсутствии заявок в очереди.
3. Начало обработки поступающей в пустую систему заявки определяется длительностью этапа «разогрева» и остатком времени «охлаждения» при условии, что система «охлаждалась» в данный момент.

Таким образом, искомая длительность случайной задержки перед обслуживанием находится путем свертки первого исходного распределения с остаточным вторым (например, «разогрева» с остаточным распределением случайной длительности «охлаждения»).

Так как преобразование Лапласа — Стильтеса (ПЛС) свертки равно произведению ПЛС составляющих [7], то искомое распределение времени «задержки» перед обслуживанием в терминах ПЛС определяется как

$$d(s) = h(s)f(s),$$

где  $h(s)$  — ПЛС распределения длительности «разогрева»,  $f(s)$  — ПЛС остаточного распределения длительности «охлаждения».

Согласно [7] начальные моменты остаточного распределения на основе начальных моментов распределения длительности «охлаждения» могут быть найдены по формуле:

$$f_k = \frac{c_{k+1}}{(k+1)c_1}, \quad k = 1, 2, \dots,$$

где  $k$  — порядок начального момента распределения.

Далее искомая случайная задержка до или после обслуживания может быть найдена на основе свертки распределений непосредственно в моментах [7]. Для этого воспользуемся символическим разложением

$$d_k = (h + f)^k,$$

в котором после разворачивания бинома показатели степени переводятся в индексы соответствующих моментов. В частности, первые три начальных момента находятся согласно

$$\begin{aligned} d_1 &= h_1 + f_1, \\ d_2 &= h_2 + 2h_1f_1 + f_2, \\ d_3 &= h_3 + 3h_2f_1 + 3h_1f_2 + f_3. \end{aligned}$$

Найденная аппроксимация случайных длительностей позволяет снизить трудоемкость моделирования на основе применения модели СМО только с «разогревом» или только с «охлаждением» каналов обслуживания вместо использования моделей с одновременным учетом как процесса «разогрева», так и «охлаждения».

ОПИСАНИЕ МОДЕЛИ

Рассмотрим модель многоканальной СМО с «охлаждением» типа  $M/M/n - H_2^c$ . Напомним, что гиперэкспоненциальное распределение второго порядка относится к распределениям фазового типа и предполагает выбор случайным процессом одной из двух альтернативных фаз [8]. С вероятностью  $y_1$  процесс «охлаждения» попадает в первую фазу и задерживается в ней случайное время, распределенное по экспоненциальному закону с параметром  $\mu_1^c$ . С вероятностью  $y_2 = 1 - y_1$  процесс попадает во вторую фазу, где экспоненциальная задержка имеет параметр  $\mu_2^c$ . Диаграмма переходов между состояниями марковского процесса, описывающего систему  $M/M/n - H_2^c$ , представлена на рисунке 4.

Крайний левый столбец на диаграмме показывает текущее число заявок в системе и указывает на номер яруса диаграммы. На каждом ярусе система находится в одном из трех состояний: обслуживание и охлаждение одного из двух типов. Состояние охлаждения характеризует некоторую работу, которая выполняется системой после окончания обслуживания последней заявки. В данной модели длительность охлаждения системы характеризуется двухфазным гиперэкспоненциальным распределением.

После обслуживания последней заявки система переходит в режим (фазу) охлаждения 1-го типа с вероятностью  $y_1$  и в режим охлаждения 2-го типа с вероятностью  $y_2$ . Каждая из фаз охлаждения характеризуется своей интенсивностью —  $\mu_1^c$  и  $\mu_2^c$  соответственно. Обслуживание заявок в системе осуществляется с интенсивностью  $\mu$ , умноженной на число занятых каналов обслуживания. При полной занятости (на ярусах с номерами больше  $n$ ) диаграмма стабилизируется на всех нижележащих ярусах.

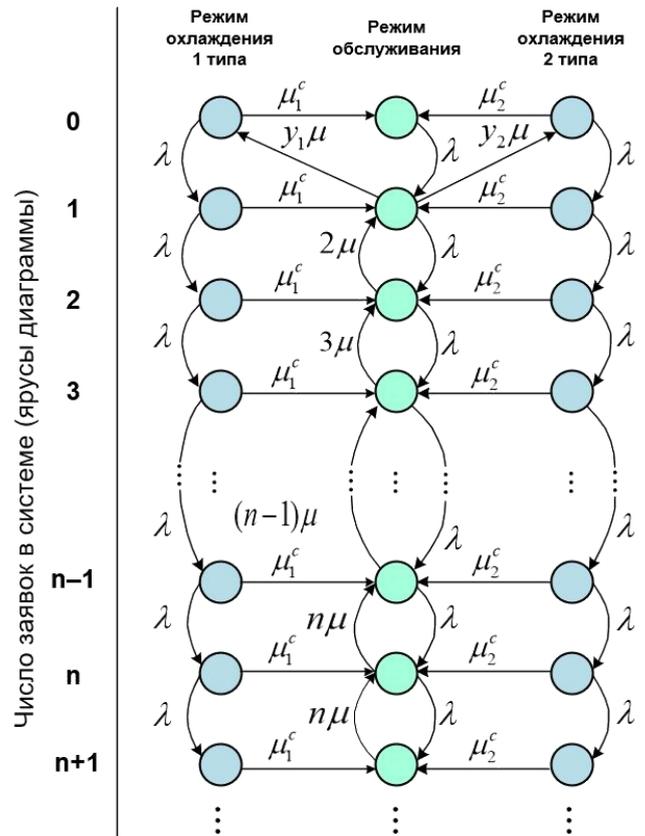


Рис. 4. Диаграмма переходов для системы  $M/M/n - H_2^c$

На основе диаграммы переходов марковизированной СМО строятся матрицы интенсивностей переходов и решаются векторно-матричные уравнения баланса переходов между микросостояниями.

МЕТОДИКА РАСЧЕТА ВЕРОЯТНОСТНО-ВРЕМЕННЫХ ХАРАКТЕРИСТИК СМО

Согласно [7] обозначим через  $S_j$  множество всех возможных микросостояний системы, при которых на обслуживании находится ровно  $j$  заявок, а через  $\sigma_j$  — количество элементов в  $S_j$ . Далее, в соответствии с диаграммой переходов построим матрицы интенсивностей инфинитезимальных переходов:

- $A_j[\sigma_j \times \sigma_{j+1}]$  — в  $S_{j+1}$  (по прибытию заявок),
- $B_j[\sigma_j \times \sigma_{j-1}]$  — в  $S_{j-1}$  (по завершению обслуживания),
- $C_j[\sigma_j \times \sigma_j]$  — в  $S_j$  (по завершению охлаждения),
- $D_j[\sigma_j \times \sigma_j]$  — ухода из микросостояний  $j$ -го яруса (диагональная матрица).

В квадратных скобках здесь и далее указывается размер матриц. Элемент  $(i, k)$  любой из этих матриц представляет интенсивность перехода из  $i$ -го состояния  $j$ -го яруса в  $k$ -е состояние смежного (по переходам рассматриваемого типа) яруса.

Для СМО  $M/M/n - H_2^c$  матрицы переходов будут иметь следующий вид:

$$A_j = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}, \quad j = \overline{0, N-1};$$

$$B_1 = \begin{bmatrix} 0 & 0 & 0 \\ \mu y_1 & 0 & \mu y_2 \\ 0 & 0 & 0 \end{bmatrix}, B_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2\mu & 0 \\ 0 & 0 & 0 \end{bmatrix}, \dots,$$

$$\dots, B_j = \begin{bmatrix} 0 & 0 & 0 \\ 0 & n\mu & 0 \\ 0 & 0 & 0 \end{bmatrix}, j = \overline{1, N};$$

$$C_j = \begin{bmatrix} 0 & \mu_1^c & 0 \\ 0 & 0 & 0 \\ 0 & \mu_2^c & 0 \end{bmatrix};$$

$$D_j = \begin{bmatrix} \lambda + \mu_1^c & 0 & 0 \\ 0 & \lambda + \min(n, j)\mu & 0 \\ 0 & 0 & \lambda + \mu_2^c \end{bmatrix}, j = \overline{0, N}.$$

Очевидно, матрицы  $A$  и  $C$  имеют одинаковый вид для всех ярусов диаграммы, а матрицы  $B$  и  $D$  зависят от номера яруса, но при  $j > n$  стабилизируются.

Введем векторы-строки  $\gamma_j = [\gamma_{j,1}, \gamma_{j,2}, \gamma_{j,3}]$  нахождения СМО в микросостояниях  $j$ -го яруса. Теперь запишем векторно-матричные уравнения баланса переходов между состояниями [7], указанными на диаграмме.

$$\gamma_0 D_0 = \gamma_0 C_0 + \gamma_1 B_1,$$

$$\gamma_j D_j = \gamma_{j-1} A_{j-1} + \gamma_j C_j + \gamma_{j+1} B_{j+1}, j = \overline{1, N},$$

где  $N$  — число обчитываемых ярусов диаграммы.

Для решения данной системы уравнений воспользуемся итерационным методом, предложенным в [8] и доработанным в [7]. Направление прогонки при расчете определим «сверху вниз» — в сторону увеличения номера яруса. В результате расчета получим отношения смежных вероятностей числа заявок в системе:  $x_j = p_{j+1}/p_j, j = \overline{1, N}$ .

После окончания итераций, используя значения  $x_j$ , осуществим переход к вероятностям состояний системы по следующему алгоритму:

- вероятность свободного состояния  $p_0$  положить равной 1;
- рассчитать  $p_{j+1} = p_j x_j, j = \overline{1, N}$ ;
- вычислить сумму  $S = \sum_{j=0}^N p_j$ ;
- нормировать полученные значения:  $p_j = p_j/S, j = \overline{0, N}$ .

Расчет возможен и для систем без ограничения буфера. В этом случае предполагается, что «хвост» распределения числа заявок представляет собой бесконечно убывающую геометрическую прогрессию, знаменатель которой равен отношению двух последних рассчитанных вероятностей.

Время ожидания вновь прибывшей заявки определяется состоянием системы непосредственно перед ее прибытием. В соответствии с теоремой PASTA (Poisson Arrival See Time Average) [9] распределение числа заявок перед прибытием очередной совпадает со стационарным. Перед прибытием очередной заявки система находится в одном из трех микросостояний — охлаждение 1-го или 2-го типа или обслуживание.

Пусть перед прибытием очередной заявки в системе уже находилось  $j$  заявок. Вероятности соответствующих микросостояний записаны в векторе-строке  $\gamma_j = [\gamma_{j,1}, \gamma_{j,2}, \gamma_{j,3}]$ . Если система находилась в первом микросостоянии  $j$ -го яруса (рис. 4), вновь прибывшая заявка будет ожидать завершения охлаждения первого типа плюс  $(j - n + 1)$  продвижений очереди. ПЛС соответствующего времени:

$$\omega_{j,1}(s) = \frac{\mu_1^c}{\mu_1^c + s} \left( \frac{n\mu}{n\mu + s} \right)^{j-n+1}.$$

В третьем микросостоянии происходит охлаждение второго типа. ПЛС соответствующего ожидания:

$$\omega_{j,3}(s) = \frac{\mu_2^c}{\mu_2^c + s} \left( \frac{n\mu}{n\mu + s} \right)^{j-n+1}.$$

Второе микросостояние соответствует режиму обслуживания. ПЛС времени ожидания вновь прибывшей заявки составит

$$\omega_{j,2}(s) = \left( \frac{n\mu}{n\mu + s} \right)^{j-n+1}.$$

Введем вектор-столбец  $\omega_j(s) = [\omega_{j,1}(s), \omega_{j,2}(s), \omega_{j,3}(s)]^T$ . Тогда ПЛС искомого времени ожидания начала обслуживания

$$\omega(s) = \sum_{j=n}^{N-1} \gamma_j \omega_j(s). \quad (1)$$

Для получения начальных моментов времени ожидания можно выполнить дифференцирование в нуле выражения (1), либо заменить все входящие в него ПЛС наборами начальных моментов, а их произведения — свертками в моментах [7].

#### РЕЗУЛЬТАТЫ ЧИСЛЕННОГО ЭКСПЕРИМЕНТА

Выполним расчет системы  $M/M/n - H_2^c$  на имитационной модели и численным методом через  $H_2$ -аппроксимацию для следующего набора исходных данных: число каналов обслуживания  $n = 3$ , средняя интенсивность входящего потока  $\lambda = 2,5$ , среднее время обслуживания заявок  $b_1 = 1$ , среднее время охлаждения системы  $c_1 = 2$ . Параметры  $H_2$ -распределения подбирались по трем моментам различных исходных распределений длительности охлаждения:

- вырожденного  $D$  (коэффициент вариации  $v = 0$ );
- равномерного  $U$  на интервале  $[0; 4]$  ( $v \approx 0,577$ );
- экспоненциального  $M$  ( $v = 1$ );
- гамма  $\Gamma$  с параметром формы  $0,5$  ( $v \approx 1,41$ );
- Вейбулла  $W$  с параметром формы  $0,46$  ( $v = 2,5$ ).

Результаты расчета распределения числа заявок в системе для  $D, M$  и  $W$  распределений приведены на рисунке 5.

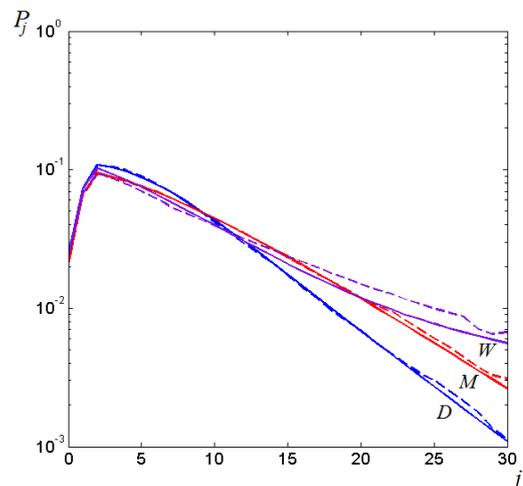


Рис. 5. Распределение числа заявок в системе  $M/M/n - H_2^c$

На рисунке 5 штриховкой показаны графики, полученные с помощью имитационной модели, сплошными линиями — на основе численного метода расчета.

Из графиков видно согласие результатов даже в области комплексных и парадоксальных параметров гиперэкспоненты. Расстояние Колмогорова между результатами, полученными методом имитационного моделирования и с помощью аппроксимации  $D$ ,  $U$ ,  $M$ ,  $\Gamma$  и  $W$  распределений времени охлаждения составило  $\{0,006; 0,005; 0,006; 0,004; 0,043\}$  соответственно, что говорит о приемлемой точности аппроксимации.

Увеличение расстояния Колмогорова для  $W$ -распределения может быть обусловлено тем, что оно относится к классу распределений с «толстым хвостом» и, следовательно, описываемая им случайная величина имеет очень большой разброс. В этой связи возрастают требования к качеству используемых в имитационной модели программных датчиков случайных чисел.

В таблице 1 представлены результаты расчета начальных моментов распределения времени ожидания  $\{w_i\}$ ,  $i = \overline{1,3}$ , полученные численно (Числ.) и с помощью имитационной модели (ИМ).

Таблица 1

Начальные моменты времени ожидания

$B(t)$	$w_1$		$w_2$		$w_3$	
	Числ.	ИМ	Числ.	ИМ	Числ.	ИМ
$D$	1,788	1,709	6,674	6,495	37,19	34,16
$U$	1,971	1,870	7,864	7,230	40,46	37,22
$M$	2,272	2,225	1,035	1,035	62,88	64,02
$\Gamma_{0,5}$	2,582	2,494	1,406	1,356	108,0	103,9
$W$	2,600	2,802	2,586	2,476	767,0	687,7

Из представленных результатов следует, что с увеличением коэффициента вариации распределения времени «охлаждения» среднее время ожидания возрастает. Сходство результатов, полученных численно и с помощью имитационного моделирования, позволяет сделать вывод о корректности предложенной методики численного расчета вероятностно-временных характеристик СМО с «охлаждением».

#### ЗАКЛЮЧЕНИЕ

На основе проведенного исследования можно сделать вывод, что процесс функционирования масштабируемого микросервиса с контейнерной виртуализацией может быть представлен моделью СМО с «охлаждением» каналов обслуживания. В модели временные задержки на выполнение операций масштабирования микросервиса могут быть представлены в виде обобщенной задержки перед началом обслуживания заявок. Расчет вероятностно-временных характеристик рассматриваемого процесса возможен на основе применения итерационного метода, точность которого подтверждена результатами имитационного моделирования.

Представленные результаты могут быть использованы при обосновании требований к характеристикам инфраструктуры масштабируемого сервиса, способного устойчиво функционировать в условиях динамического изменения нагрузки.

#### ЛИТЕРАТУРА

1. Ньюмен, С. Создание микросервисов = Building Microservices / Пер. с англ. Н. Вильчинского. — Санкт-Петербург: Питер, 2016. — 304 с. — (Бестселлеры O'Reilly).
2. Khazaee, H. Performance Modeling of Cloud Computing Centers: A Thesis of the Degree of Doctor of Philosophy. — Winnipeg: The University of Manitoba, 2012. — 218 p.
3. Гиндин, С. И. Численный расчет многоканальной системы массового обслуживания с рекуррентным входящим потоком и «разогревом» / С. И. Гиндин, А. Д. Хомоненко, С. Е. Адауров // Известия Петербургского университета путей сообщения. 2013. № 4 (37). С. 92–101.
4. Khalil, M. M. Testing of Software for Calculating a Multi-channel Queuing System with «Cooling» and E2-approximation / M. M. Khalil, A. A. Andruk // Интеллектуальные технологии на транспорте. 2016. № 4 (8). С. 22–28.
5. Khomonenko, A. D. A Cloud Computing Model Using Multi-Channel Queuing System with Cooling / A. D. Khomonenko, S. I. Gindin, M. M. Khalil // Proceedings of the XIX International Conference on Soft Computing and Measurements (SCM'2016) (Saint Petersburg, Russia, 25–27 May 2016). — Institute of Electrical and Electronics Engineers, 2016. — Pp. 103–106. DOI: 10.1109/SCM.2016.7519697.
6. Лохвицкий, В. А. Численный анализ системы массового обслуживания с гиперэкспоненциальным «охлаждением» / В. А. Лохвицкий, А. В. Уланов // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2016. № 4 (37). С. 36–43. DOI: 10.17223/19988605/37/4.
7. Рыжиков, Ю. И. Алгоритмический подход к задачам массового обслуживания: Монография. — Санкт-Петербург: ВКА им. А. Ф. Можайского, 2013. — 496 с.
8. Takahashi, Y. A Numerical Method for the Steady-State Probabilities of a G1/G/C Queuing System in a General Class / Y. Takahashi, Y. Takami // Journal of the Operations Research Society of Japan. 1976. Vol. 19, No. 2. Pp. 147–157. DOI: 10.15807/jorsj.19.147.
9. Рыжиков, Ю. И. Итерационный метод расчета многоканальных систем с произвольным законом обслуживания / Ю. И. Рыжиков, А. Д. Хомоненко // Проблемы управления и теория информации. 1980. Т. 9, № 3. С. 203–213.

# A Scalable Microservice Model Based on a Queuing System with «Cooling»

Grand PhD V. A. Lokhvitsky, PhD V. A. Goncharenko, E. S. Levchik

Mozhaisky Military Space Academy

Saint Petersburg, Russia

lokhv\_va@mail.ru, vlango@mail.ru

**Abstract.** The formalization of a scalable service functioning is proposed, taking into account the time spent on configuring computing nodes based on a multichannel non-exponential queuing system (QS) with «heating» and «cooling» of channels. The complexity of the model is reduced by representing the durations of «heating» and «cooling» in the form of a generalized random delay before servicing. The main stages of the method of calculating the probability-time characteristics of the QS are described. The markovization of the non-Markov components of the QS under consideration was carried out using the H2-distribution. The probability distribution of the states of the QS under consideration is obtained. The results obtained by the numerical method are compared with the results of simulation.

**Keywords:** microservice, queuing system, channels «cooling», markovization, generalized service delay.

## REFERENCES

1. Newman S. Building Microservices [Sozdanie mikroservisov]. Saint Petersburg, Piter Publishing House, 2016, 304 p.
2. Khazaei H. Performance Modeling of Cloud Computing Centers: A Thesis of the Degree of Doctor of Philosophy. Winnipeg, The University of Manitoba, 2012, 218 p.
3. Gindin S. I., Khomonenko A. D., Adadurov S. E. Numerical Calculations of Multichannel Queuing System with Recurrent Input and «Warm Up» [Chislenny raschet mnogokanal'noy sistemy massovogo obsluzhivaniya s rekurrentnym vkhodyashchim potokom i «razogrevom»], *Proceedings of Petersburg Transport University [Izvestiya Peterburgskogo universiteta putey soobshcheniya]*, 2013, No. 4 (37), Pp. 92–101.
4. Khalil M. M., Andruk A. A. Testing of Software for Calculating a Multichannel Queuing System with «Cooling» and E2-approximation, *Intellectual Technologies on Transport [Intellektual'nye tekhnologii na transporte]*, 2016, No. 4 (8), Pp. 22–28.

5. Khomonenko A. D., Gindin S. I., Khalil M. M. A Cloud Computing Model Using Multi-Channel Queuing System with Cooling, *Proceedings of the XIX International Conference on Soft Computing and Measurements (SCM`2016), Saint Petersburg, Russia, May 25–27, 2016*. Institute of Electrical and Electronics Engineers, 2016, Pp. 103–106. DOI: 10.1109/SCM.2016.7519697.

6. Lokhvitsky V. A., Ulanov A. V. The Numerical Analyses of Queuing System with Hyperexponential Distribution of Cooling Time [Chislenny analiz sistemy massovogo obsluzhivaniya s giperekspontsial'nym «okhlazhdeniem»], *Tomsk State University Journal of Control and Computer Science [Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika]*, 2016, No. 4 (37), Pp. 36–43. DOI: 10.17223/19988605/37/4.

7. Ryzhikov Yu. I. Algorithmic approach to queuing tasks: Monograph [Algoritmicheskiy podkhod k zadacham massovogo obsluzhivaniya: Monografiya]. Saint Petersburg, Mozhaisky Military Space Academy, 2013, 496 p.

8. Takahashi Y., Takami Y. A Numerical Method for the Steady-State Probabilities of a G1/G/C Queuing System in a General Class, *Journal of the Operations Research Society of Japan*, 1976, Vol. 19, No. 2, Pp. 147–157. DOI: 10.15807/jorsj.19.147.

9. Ryzhikov Yu. I., Khomonenko A. D. Iterative Method for Analysis of Multichannel Queueing Systems with General Service Time Distribution [Iteratsionnyy metod rascheta mnogokanal'nykh sistem s proizvol'nym zakonom obsluzhivaniya], *Problems of Control and Information Theory [Problemy upravleniya i teoriya informatsii]*, 1980, Vol. 9, No. 3, Pp. 203–213.

# Программная модель маскировки скрытого сообщения в задачах стеганографии

д.т.н. В. Н. Кустов, А. И. Грохотов, Е. В. Головков

Петербургский государственный университет путей сообщения Императора Александра I

Санкт-Петербург, Россия

kvnvika@mail.ru, grohotov.aleksei@mail.ru, jyk22@mail.ru

**Аннотация.** Рассматривается проблема разработки программной модели маскировки скрытого сообщения в стегосистемах HUGO в условиях естественного шума в канале связи с использованием дискретной хаотической карты кота Арнольда и карты Бейкера, которые являются итеративными обратимыми дискретными хаотическими преобразованиями. Подробно рассмотрены все этапы разработки программной модели, такие как выбор среды разработки, состав программных модулей и их назначение, скриншоты графических интерфейсов для различных режимов работы модели. В программной модели для оценки хаотического состояния скрытого сообщения, представленного цифровым неподвижным изображением, введен коэффициент хаотичности — числовой показатель энтропии вероятности неупорядоченных пикселей. В разработанной программной модели предложен метод определения максимального значения коэффициента хаотичности, соответствующего максимальному хаотическому состоянию скрытого изображения. Тестирование программной модели показывает практическую неотличимость преобразованного скрытого сообщения от естественного шума в канале связи и значительно повышает безопасность стегосистем HUGO.

**Ключевые слова:** программная модель, стегосистема HUGO, карта кота Арнольда, карта Бейкера, коэффициент корреляции Пирсона, естественный шум, коэффициент хаотичности, хаотические карты, стеганология.

## ВВЕДЕНИЕ

Стеганография — это специфическая область научной деятельности, связанная с созданием и исследованием методов сокрытия и обнаружения передачи информации. Как и криптография, стеганография также существует с древних времен. Тем не менее на этом аналогии заканчиваются, особенно в области теории. За последние четверть века была создана и успешно развивается новая дисциплина — математическая стеганография, или стеганология, в которой изучаются математические модели стегосистем.

Если рассматривать новые тенденции в современной стеганографии, то можно перечислить следующие ее отличительные особенности:

1. Все более широкое использование неподвижных цифровых фотографий в качестве покрывающих объектов с их предварительным преобразованием с использованием дискретных преобразований в сочетании с уже известными методами стеганографии.

2. Исследования по созданию новых стегосистем с повышенной секретностью передачи данных.

3. Повышенное внимание к учету наличия помех в каналах связи и способов борьбы с ними.

4. Активное использование дискретных хаотических преобразований скрытых сообщений для маскировки их

под естественный шум в канале передачи данных с целью повышения безопасности стегосистем HUGO.

5. Широкое применение новых самоортогональных помехоустойчивых кодов для кодирования покрывающих объектов со встроенными скрытыми сообщениями при передаче их по каналам связи с высоким уровнем шума.

6. Повышенное внимание к новым разработкам эффективных многопороговых декодеров с оптимальными характеристиками энергопотребления, скоростью и надежностью процессов декодирования.

7. Переход от предварительного криптографического преобразования скрытого сообщения перед встраиванием его в покрывающий объект к использованию криптографических преобразований непосредственно в алгоритме встраивания скрытого сообщения.

8. Активное расширение исследований в области синтеза высокочастотных стегосистем (так называемых HUGO-систем).

В этой статье рассмотрены дискретные хаотические преобразования скрытого сообщения, маскирующие его под естественный шум в канале связи. Пользователи в системах электронного документооборота могут реализовать такие методы для повышения уровня безопасности конфиденциальных документов, передаваемых по каналу связи.

## ДИСКРЕТНЫЕ ПРЕОБРАЗОВАНИЯ ПОКРЫВАЮЩИХ ОБЪЕКТОВ И СКРЫТЫХ СООБЩЕНИЙ В ЗАДАЧАХ СТЕГАНОГРАФИИ

Методы стеганографии, использующие неподвижное цифровое изображение в качестве покрывающего объекта, состоят из двух широких классов: первый класс методов вводит секретное сообщение в пространственную область покрывающего объекта, второй класс методов делает это в частотной области. Первый класс методов изменяет пространственную область покрывающего объекта, например его пиксели. Однако эти методы подвержены определенным графическим преобразованиям покрывающего объекта, таким как уменьшение или увеличение масштаба, шум, сжатие и другие изменения, которые делают невозможным извлечение скрытого сообщения.

Вторая группа методов встраивает скрытые сообщения в частотный спектр покрывающего объекта. В этой группе методов используются предварительные дискретные преобразования частотного спектра покрывающего объекта в низкочастотную и высокочастотную части спектра. Внедрение секретного сообщения обычно осуществляется в высокочастотной области спектра. Исследователь может использовать эту группу методов для различных форматов

покрывающих объектов. Применяются следующие дискретные преобразования:

- дискретное косинусное преобразование (англ. *Discrete Cosine Transform, DCT*);
- дискретное преобразование Адамара (англ. *Discrete Hadamard Transform, DHT*);
- дискретное вейвлет-преобразование (англ. *Discrete Wavelet Transform, DWT*);
- дискретное преобразование Фурье (англ. *Discrete Fourier Transform, DFT*);
- разложение по сингулярным значениям (англ. *Singular Value Decomposition, SVD*).

Следует отметить, что наиболее часто используются дискретные преобразования DCT и DWT. Эти методы были успешно применены для разработки широко используемых алгоритмов сжатия неподвижных изображений JPEG2000 и JPEG.

Как уже отмечено выше, в современной стеганографии для повышения безопасности стегосистем исследователи все чаще стали использовать дискретные хаотические преобразования для скрытых сообщений. Хаотические дискретные преобразования относятся к области исследований теории хаоса.

Теория хаоса — это раздел математики, который имеет дело с системами, проявляющими хаотическое поведение при определенных условиях. Даже для случайного поведения некоторых детерминированных систем применима концепция хаоса. В данном случае такое поведение называется детерминированным хаосом, а соответствующий раздел математики называется теорией детерминированного хаоса.

Общепринято использовать понятие энтропии в качестве основной меры случайности системы. Понятие энтропии играет важную роль в физике и математике. Существуют различные подходы к определению энтропии, разработанные и подробно описанные в работах [1–7]. Имеются известные приложения, в которых энтропия используется в качестве статистической оценки данных, построенных с использованием наблюдений. Например, такие оценки полезны в теории выбора объектов и обнаруживают неоднородности текстуры, отношения сигнал/шум и т. д. [1–7].

Обратимость хаотических процессов в дискретном времени в настоящее время хорошо изучена и доказана. Любой хаотический процесс может протекать как в прямом, так и в обратном направлении, то есть обратимо.

Системы с высоким значением энтропии более дезорганизованы, чем системы с низким значением. В неупорядоченных системах существует множество возможных состояний, а в упорядоченных системах существует небольшая вероятность упорядочения конфигурации.

На первый взгляд, согласно второму закону термодинамики, замкнутая система должна стремиться ко все более неупорядоченному состоянию, то есть к хаосу. Однако наряду с этими процессами можно наблюдать обратные переходы от хаоса к порядку на более высоком уровне.

Нарушение порядка ведет к разрушению упорядоченности и в конечном счете приводит к бесструктурной, неупорядоченной форме существования системы с максимальной энтропией системы, то есть к хаосу. Однако ка-

кой бы хаотичной она ни казалась, система всегда представляет собой некую структуру. Тогда хаос может действовать как чрезмерно сложная упорядоченность.

Таким образом, в замкнутой системе всегда происходит переход от Порядка к Хаосу ( $\Pi \rightarrow X$ ) в процессе хаотических преобразований. То есть происходит чередование изменений:  $X \rightarrow \Pi \rightarrow X \rightarrow \Pi \rightarrow \dots$  и т. д.

Хаотическое преобразование (декомпозиция или отображение) обычно используется для формального описания детерминированного хаоса. Существует около 200 различных хаотических преобразований (хаотических карт) и их число постоянно растет. Список дискретных хаотических карт, которые можно использовать для хаотических преобразований скрытых сообщений, приведен в таблице 1.

Таблица 1

Список хаотических карт, которые можно использовать для хаотических преобразований скрытых сообщений

Наименование хаотической карты	Характеристики	
	Размерность пространства	Количество параметров
2D Lorentz System	2	1
Arnold's Cat Map	2	2
Baker's Map	2	2
Chaotic Pool Map	2	1
Bogdan map	2	3
Circle Map	1	2
Clifford Fractal Map	2	4
The De Jong Fractal Map	2	4
Discretized Circular Van der Pol System	2	1
Duffing Map	2	5
Dyadic Transformation	1	0
Fractal-Dream System	2	0
Gingerbread Map	2	0
Map of Oenone	2	2
Horseshoe Map	2	1
Poincare Return Map	2	3

Хаотические карты классифицируются в соответствии со следующими признаками классификации (рис. 1):

- временная область (непрерывная, дискретная);
- пространственная область (реальная, рациональная, комплексная, дискретная);
- количество пространственных измерений (фиксированное (1, 2, 3), произвольное);
- количество параметров (фиксированное (0, 1, 2, 3, 4, 6, 9, 10, 18), переменное).

Дискретные хаотические преобразования скрытых сообщений, представленных неподвижными изображениями, маскируют их под естественный шум в стеганографическом канале связи.

Для преобразования скрытого сообщения, представленного в виде неподвижного цифрового изображения, необходимо перевести его в форму, которая очень напоминает шум в канале связи. Обычно это делается с использованием дискретных хаотических карт.

Следует подчеркнуть, что для использования хаотического дискретного отображения для маскировки скрытого сообщения под естественным шумом оно должно, помимо прочего, обладать свойством шифрования (обеспечения

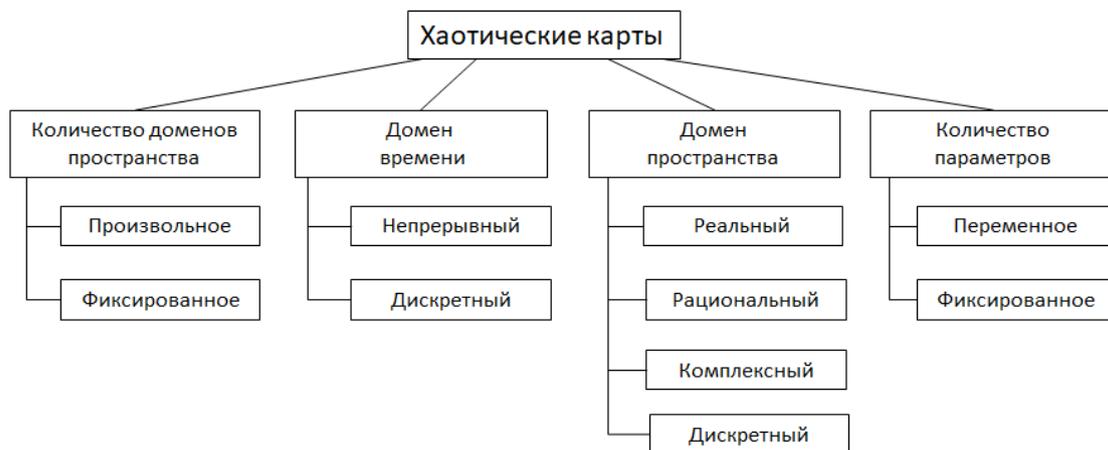


Рис. 1. Классификация хаотических карт

конфиденциальности) двумерного объекта. При выполнении пользователем хаотического преобразования необходимо сгенерировать секретный ключ, использование этого ключа гарантирует, что преобразование является обратимым. Кроме того, хаотическое отображение должно быть детерминированным, конечным и легко вычислимым.

Следует подчеркнуть, что для использования хаотического дискретного отображения для маскировки скрытого сообщения под естественным шумом оно должно, помимо прочего, обладать свойством шифрования (обеспечения конфиденциальности) двумерного объекта. При выполнении пользователем хаотического преобразования необходимо сгенерировать секретный ключ, использование этого ключа гарантирует, что преобразование является обратимым. Кроме того, хаотическое отображение должно быть детерминированным, конечным и легко вычислимым.

Для исследования были выбраны хаотичные отображения карты кота Арнольда (Arnold's Cat Map, ACM) и карты Бейкера (Baker's Map, BM), чтобы замаскировать скрытые сообщения под естественный шум.

АСМ и ВМ имеют следующие значения для перечисленных классификационных признаков:

- временная область — дискретная;
- пространственная область — дискретная;
- количество пространственных измерений — 2;
- количество параметров — 2.

Рассмотрим подробно эти два дискретных хаотических преобразования.

#### ХАОТИЧЕСКОЕ ОТОБРАЖЕНИЕ КАРТЫ КОТА АРНОЛЬДА

В дискретных хаотических картах АСМ, первым предложенная В. И. Арнольдом, известна как хаотическое преобразование на поверхности тора.

Исследователи могут формально описать АСМ как хаотическое двумерное преобразование с помощью следующих формул:

$$\begin{aligned} p' &= (2p + q) \bmod 1, \\ q' &= (p + q) \bmod 1. \end{aligned}$$

В матричной форме эти отношения имеют следующий вид:

$$\begin{pmatrix} p' \\ q' \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix} \bmod 1 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix} \bmod 1.$$

В этих соотношениях знак штриха показывает динамику изменения значений параметров на последовательном временном шаге.

Поскольку поверхность тора обыкновенно представляет собой пространство АСМ, изменение значений каждого параметра ограничено диапазоном от нуля до единицы. Обычно это квадрат со сторонами, равными единице, с координатами  $q$  и  $p$ . Графическая иллюстрация АСМ приведена на рисунке 2. АСМ получил свое название потому, что Владимир Арнольд проиллюстрировал его изображением, напоминающим кошачью голову.

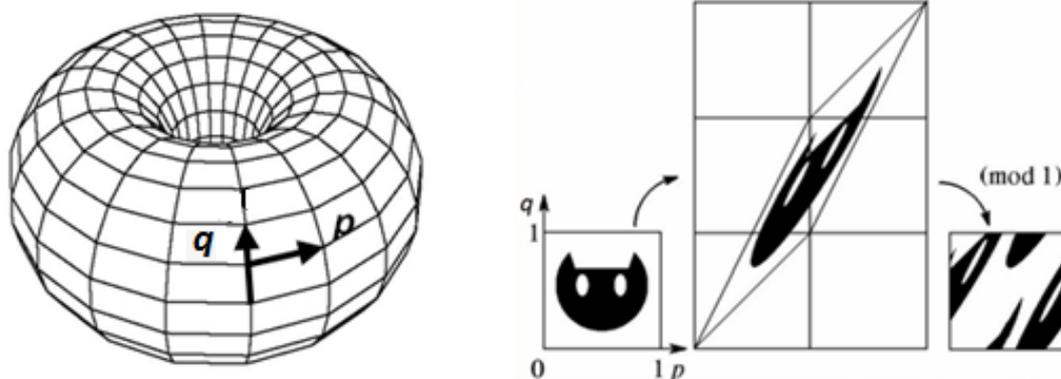


Рис. 2. Графическое представление АСМ

Стоит упомянуть, что изображение, подвергнутое АСМ, всегда сохраняет свою площадь и обычно используется для описания динамики хаотических процессов.

Преобразуем скрытое сообщение с помощью алгоритма АСМ. Как показано на рисунке 3, на итерации 35

скрытое сообщение почти неотличимо от шума. Номер итерации может быть использован в качестве ключа для восстановления скрытого сообщения на стороне получателя.

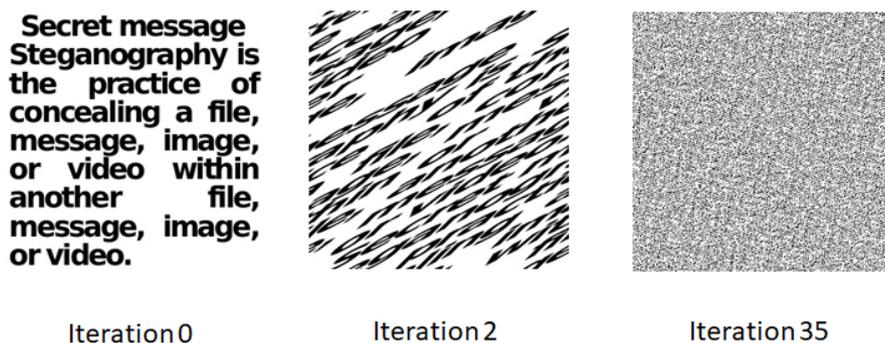


Рис. 3. Итеративные преобразования скрытого сообщения с использованием АСМ до 35 итерации

ХАОТИЧЕСКОЕ ОТОБРАЖЕНИЕ КАРТЫ БЕЙКЕРА

Карта Бейкера — это хаотическое отображение единичного квадрата в себя в теории динамических систем. Трансформация названа в честь операции замешивания, которую пекари применяют к тесту. Тесто разрезают пополам, две половинки складывают друг на друга и сжимают. ВМ может быть представлена в виде оператора двустороннего сдвига решетчатой модели с двумя состояниями. Она топологически сопряжена с картой Подковы. В физике цепочка связанных преобразований ВМ используется для моделирования детерминированной диффузии. Как и во многих детерминированных динамических системах, ВМ изучается по ее действию на пространство функций, определенных на единичном квадрате. ВМ определяет оператор в функциональном пространстве, известный как оператор переноса карты. ВМ — это точно

разрешимая модель детерминированного хаоса. В этой модели собственные функции и собственные значения оператора переноса могут быть определены явно в виде следующих соотношений:

$$(p', q') = \begin{cases} (2p, q/2) & \text{если } 0 \leq p < 1/2 \\ (2 - 2p, 1 - q/2) & \text{если } 1/2 \leq p < 1. \end{cases}$$

ВМ — это обратимое преобразование. Легко показать, что обратное преобразование для карты Бейкера выполняется в соответствии со следующими соотношениями:

$$(p, q) = \begin{cases} (p'/2, 2q') & \text{если } 0 \leq p' < 1/2 \\ (1 - p'/2, 2 - 2q') & \text{если } 1/2 \leq p' < 1. \end{cases}$$

Динамика преобразований скрытого сообщения картой Бейкера показана на рисунке 4.

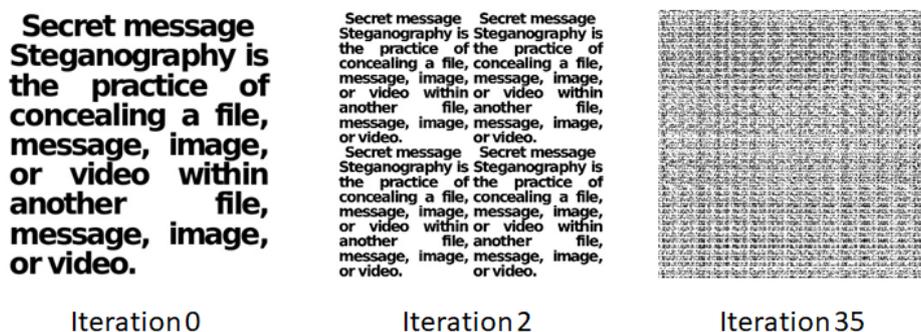


Рис. 4. Динамика преобразований скрытого сообщения в соответствии с картой Бейкера до 35 итерации

КОЭФФИЦИЕНТ СЛУЧАЙНОСТИ КАК МЕРА ХАОТИЧНОСТИ

При выполнении хаотических преобразований важной проблемой является поиск оптимального уровня хаотического изменения скрытого сообщения. Для этой цели предлагается использовать коэффициент хаотичности  $K_{hi}$ . Задача поиска сводится к определению номера итерации хаотического преобразования скрытого сообщения, при котором  $K_{hi}$  достигает своего максимума (состояние максимальной хаотичности), то есть

$$K_h^{opt} = \max K_{hi} \text{ for } i = 1, \dots, m,$$

где  $K_{hi}$  — значение коэффициента хаотичности при номере итерации  $i$ ;  
 $m$  — общее количество выполненных итераций.

$$K_{hi} = \frac{K_{success_i}}{K_{pixel}},$$

где  $K_{success_i}$  — общее количество успешных проверок хаотичности окружения для всех пикселей изображения скрытого сообщения на итерации  $i$ ;

$K_{pixel}$  — общее количество пикселей в скрытом сообщении.

Проверка считается успешной для проверки хаотичности окружения, в результате которой выполняется следующее условие для проверяемого пикселя  $B_p$  (расположенного в центре окружения):

$$B_p \neq \frac{\sum_{i=1}^n B_i}{n}, \quad (1)$$

где  $B_i$  — пиксель  $i$  из центра окружения;  $n$  — количество пикселей в окружении.

Проверяя условие (1), можно определить, является ли окружение однородным (упорядоченным или турбулентным (хаотичным)).

Различные формы окружения (окрестности), использованные в эксперименте, показаны на рисунке 5.

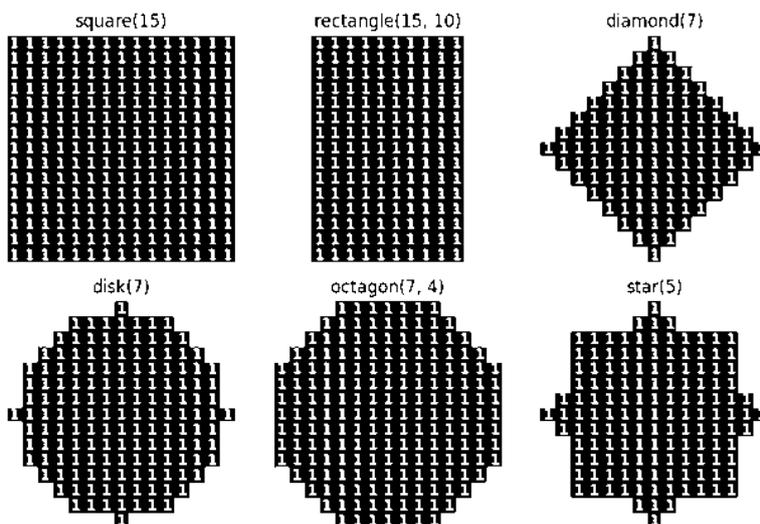


Рис. 5. Используемые формы окружения

Перед выполнением первой итерации вычисляется начальное значение  $K_{success_i}$  для начального состояния скрытого сообщения. Затем в каждом случае условия (1) значение  $K_{success_i}$  увеличивается на 1.

Результаты расчета функций  $K_{h_i} = f(i)$  и  $E = f(i)$  для алгоритма АСМ с использованием окрестностей disk(7) показаны на рисунке 6.

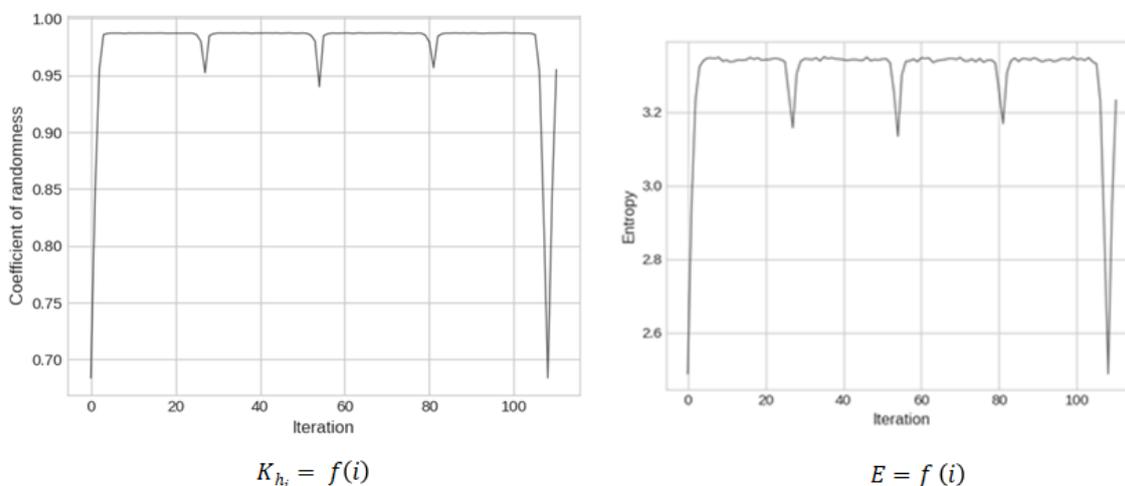


Рис. 6. Результаты вычисления функций  $K_{h_i} = f(i)$  и  $E = f(i)$

Те же функциональные зависимости для алгоритма ВМ имеют схожий внешний вид. Максимальные значения  $K_{h_i} = f(i)$  и  $E = f(i)$  составляют 0,9873422728531308 и 3,349067973898346 соответственно для скрытого сообщения в форме, показанной на рисунке 3 для итерации 35.

Скрытое сообщение в этой итерации неотлично от естественного шума. Коэффициент корреляции Пирсона для этих зависимостей (АСМ и ВМ) составляет 0,9904321037457637.

### РАЗРАБОТКА ПРОГРАММНОЙ МОДЕЛИ ДЛЯ МАСКИРОВКИ НЕПОДВИЖНЫХ ЦИФРОВЫХ ИЗОБРАЖЕНИЙ

В этом разделе представлен прототип программной модели для реализации процесса маскирования цифровых неподвижных изображений на пути от отправителя к получателю с использованием их дискретных преобразований.

В данной модели рекомендуется выполнить предварительное преобразование файла в форму, во многом напоминающую шум, что позволяет повысить безопасность системы. В случае утечки секретного сообщения восстановить его можно будет только с использованием закрытого ключа, с помощью которого изображение было преобразовано изначально.

### ОПИСАНИЕ ПРОГРАММНОЙ СРЕДЫ

Для реализации программной модели была использована интегрированная среда разработки NetBeans IDE.

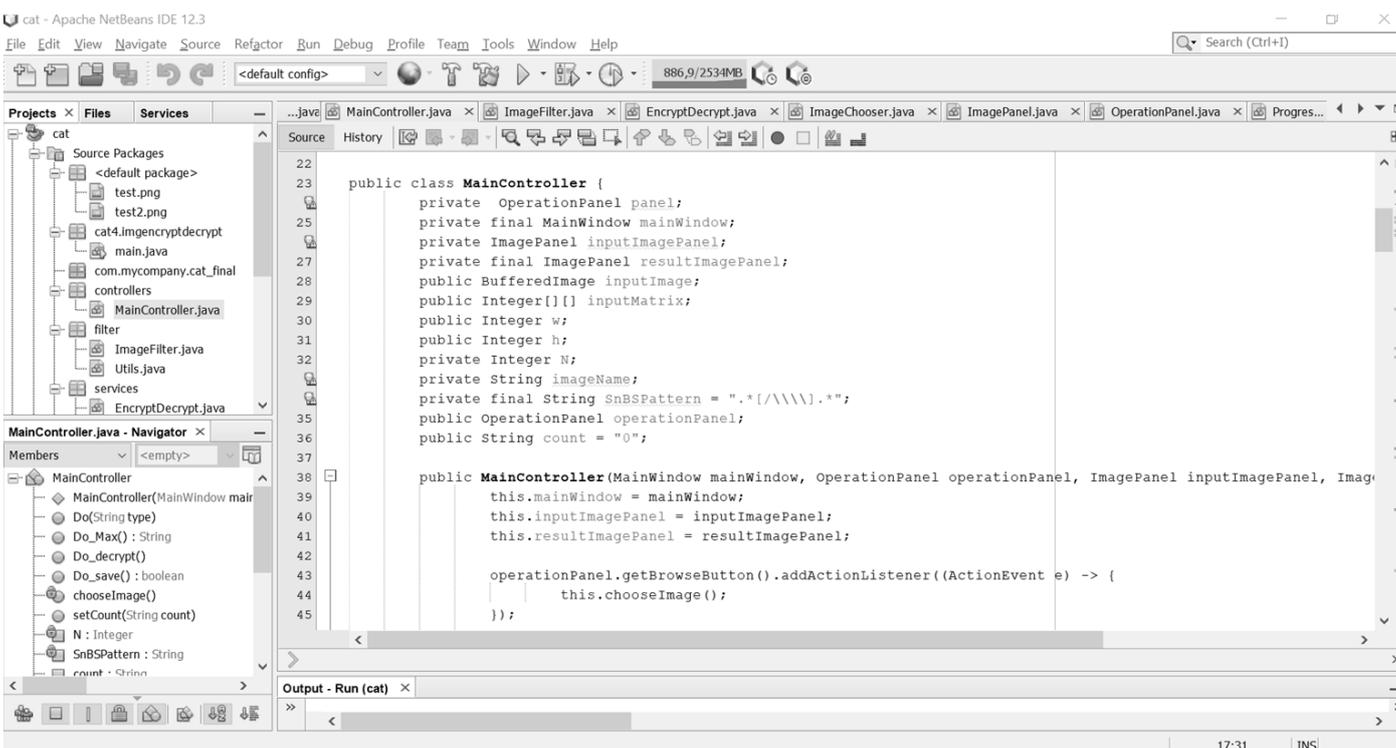


Рис. 7. Общий графический интерфейс среды разработки

### ОПИСАНИЕ ЯЗЫКА ПРОГРАММИРОВАНИЯ

Java — это строго типизированный объектно-ориентированный язык программирования. Его разработала компания Sun Microsystems, впоследствии поглощенная Oracle.

Характерные особенности языка Java:

- язык прост в использовании, в нем исправлены ошибки из других языков (например, не используются указатели);
- объектно-ориентированный;
- устойчивый язык, программы Java застрахованы от многих ошибок;
- безопасный;
- сетевой (распределенный);
- независимый от архитектуры (независимый от платформы);
- интерпретированный;

Проект NetBeans IDE поддерживается и спонсируется Oracle, но разрабатывают NetBeans независимое сообщество разработчиков-энтузиастов (NetBeans Community) и компания NetBeans Org.

Последние версии IDE NetBeans поддерживают рефакторинг, профилирование, выделение синтаксических конструкций цветом, автоматическое заполнение типизированных структур на лету и множество predefined шаблонов кода.

Среда разработки IDE NetBeans позволяет разрабатывать приложения на различных языках программирования (Java, Python, PHP, JavaScript, C, C++, Ada и некоторых других).

Для написания кода программной модели был использован язык программирования Java. Общий графический интерфейс среды разработки показан на рисунке 7.

- портативный;
- высокая производительность;
- многопоточность;
- динамичный.

### СТРУКТУРА РАЗРАБАТЫВАЕМОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Структура разработанного программного комплекса показана на рисунке 8.

Main.java является основным классом, создает объекты вспомогательных классов и вызывает соответствующие методы.

MainController.java — это класс, который реализует обработку событий для различных действий пользователя. Он также включает методы создания интерфейсных панелей для пользователя. Основные методы класса:

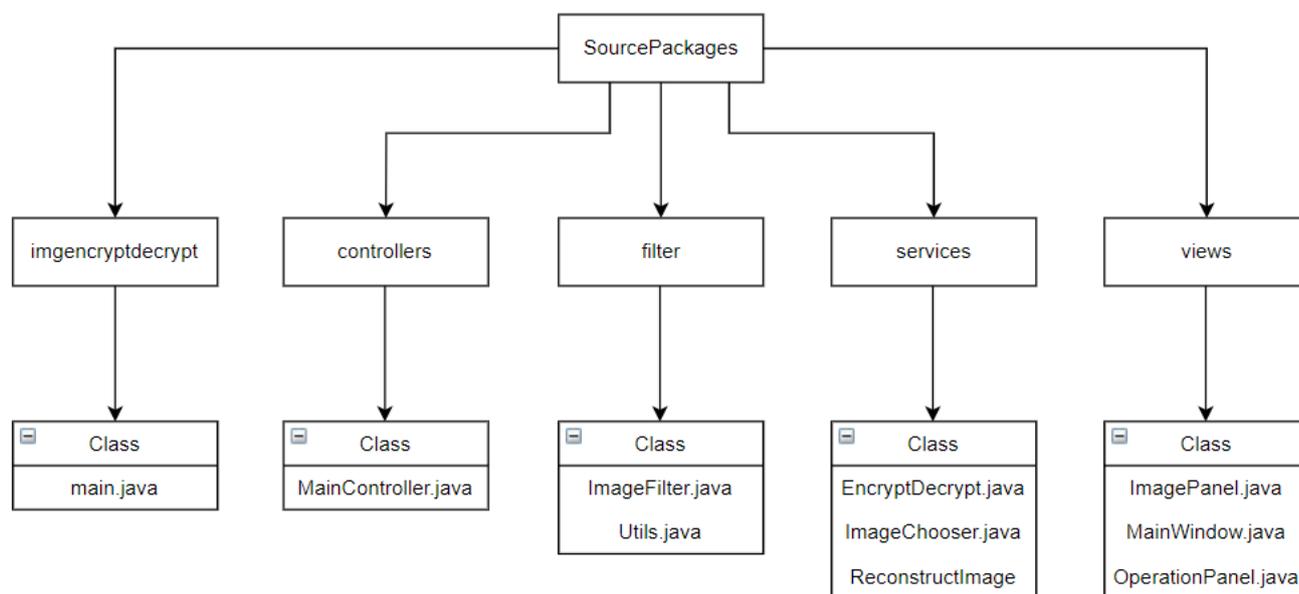


Рис. 8. Структура разработанного программного комплекса

- chooseImage() отвечает за обработку действий пользователя при выборе изображения, также преобразует исходный объект в массив байтов;
- Do() вызывает функцию шифрования encrypt и отображает изображение на панели после каждой итерации;
- Do\_Max() запускается при нажатии кнопки, которая запускает процесс преобразования изображения до тех пор, пока оно не вернется в исходное состояние;
- Do\_decrypt() вызывает функцию дешифрования шифрования и отображает изображение на панели после каждой итерации;
- Do\_save() сохраняет преобразованное изображение в каталог, указанный пользователем.

ImageFilter.java — класс, который проверяет формат выбранного изображения для преобразования.

EncryptDecrypt.java — это основной класс, реализующий алгоритм дискретного хаотического преобразования. Основные методы класса:

- ACM() реализует алгоритм маскирования изображений в шум;
- ACM\_Max() реализует обработку события, которое пользователь инициирует, нажимая на кнопку, для преобразования изображения до тех пор, пока оно не вернется в исходное состояние;
- Compare() является вспомогательным методом для ACM\_Max(), проверяет соответствие преобразованного объекта с исходным изображением после каждой итерации преобразования.

ReconstructImage.java — класс, который реализует методы преобразования изображения в формате PNG или JPEG в массив байтов.

Перейдем к рассмотрению блок-схемы программного интерфейса (рис. 9).

Класс MainWindow.java отвечает за главное окно пользователя. В верхней части главного окна рисуются окна, в которых отображается исходное изображение и изображение после преобразования. Класс, отвечающий за эту

панель — ImagePanel.java. В нижней части окна находится панель инструментов с основными пользовательскими функциями. За эту панель отвечает класс OperationPanel.java.

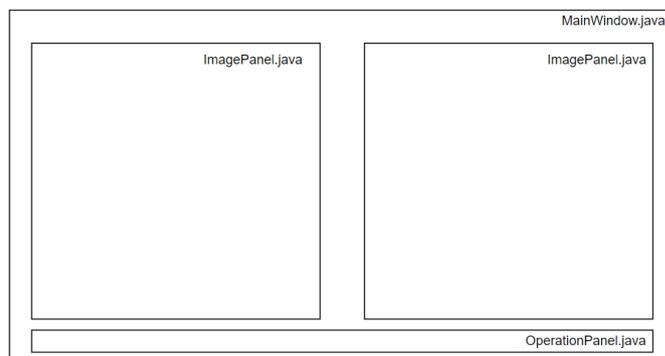


Рис. 9. Структурная схема программного интерфейса

В процессе разработки использовались следующие встроенные библиотеки (рис.10):

1. AWT (Abstract Window Toolkit) — библиотека для работы с графическими объектами, используемая для отрисовки пользовательского интерфейса.
2. IO (ввод/вывод) — библиотека для организации потоков ввода/вывода, используемая для чтения файла изображения и отображения его на панели пользователя.
3. Net — библиотека для организации соединений, используемая для загрузки файла изображения из файловой системы в программу.
4. Util — библиотека для управления системными компонентами.
5. ImageIO — библиотека для загрузки и выгрузки графических изображений в буфер и из буфера, используемая для промежуточного сохранения преобразованного изображения.
6. Swing — библиотека для работы с графической оболочкой.

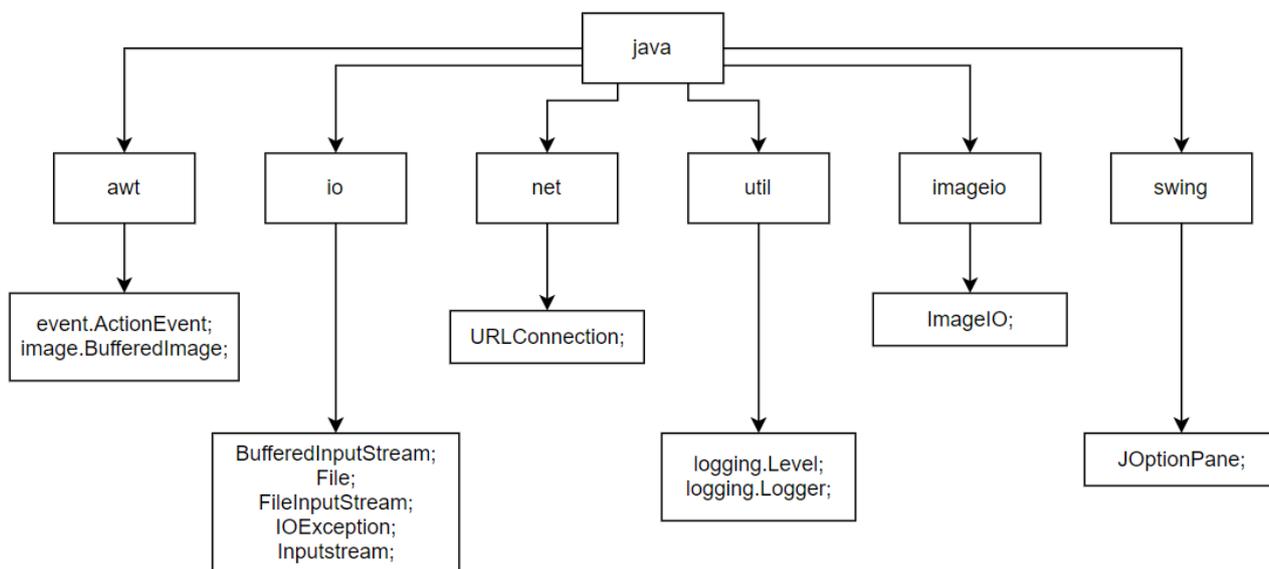


Рис. 10. Список используемых библиотек

СТРУКТУРА РАЗРАБАТЫВАЕМОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Для реализации моделирования был разработан алгоритм, отражающий последовательность действий, необходимых для выполнения.

В программной модели при выборе неподвижного цифрового объекта для маскировки под шум можно использовать два формата изображений: PNG и JPEG. PNG расшифровывается как портативный сетевой графический файл (Portable Network Graphics) и поддерживает сжатие без потерь. В отличие от JPEG, сохранение цифрового файла в формате PNG не снижает его качества. Формат PNG подходит, если изображения имеют прозрачный фон (например, логотип компании). Их можно использовать в качестве наложений видео или при объединении двух или более изображений в качестве слоев.

Формат файла JPEG (или JPG) — это стандартное расширение для цифровых изображений. Файлы JPEG поступают со всех цифровых камер, которые используются на разных платформах. JPEG является аббревиатурой названия организации-разработчика — Объединенной экспертной группы по фотографии (Joint Photographic Expert Group).

Преимущество JPEG заключается в возможности многократно редактировать необработанные изображения без потери качества.

Файлы изображений JPEG используются для немедленного обмена изображениями.

На рисунке 11 показана блок-схема, описывающая логику работы программного модуля на этапе выбора исходного изображения.

Первым шагом работы с программным модулем является выбор объекта для преобразования. Пользователь имеет возможность выбирать между двумя форматами изображений: PNG или JPEG. Эти действия выполняются через пользовательский интерфейс. После выбора объекта программа проверяет формат изображения. Если он удовлетворяет условию, то объект преобразования передается модулю шифрования.

```

private void choose Image() {
ImageChooser imageChooser = new ImageChooser();
File input Image File = imageChooser.getSelectedFile();
  
```

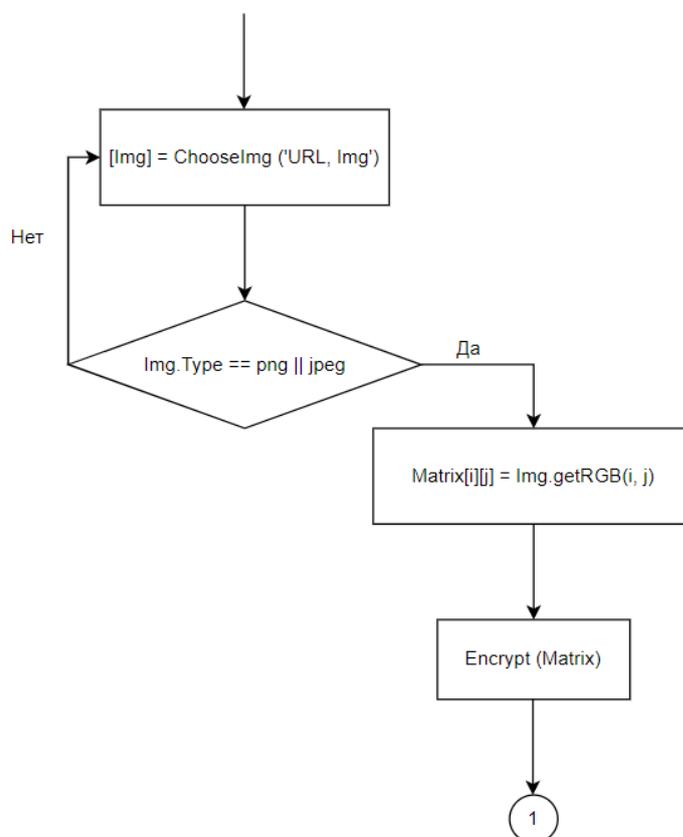


Рис. 11. Считывание изображения

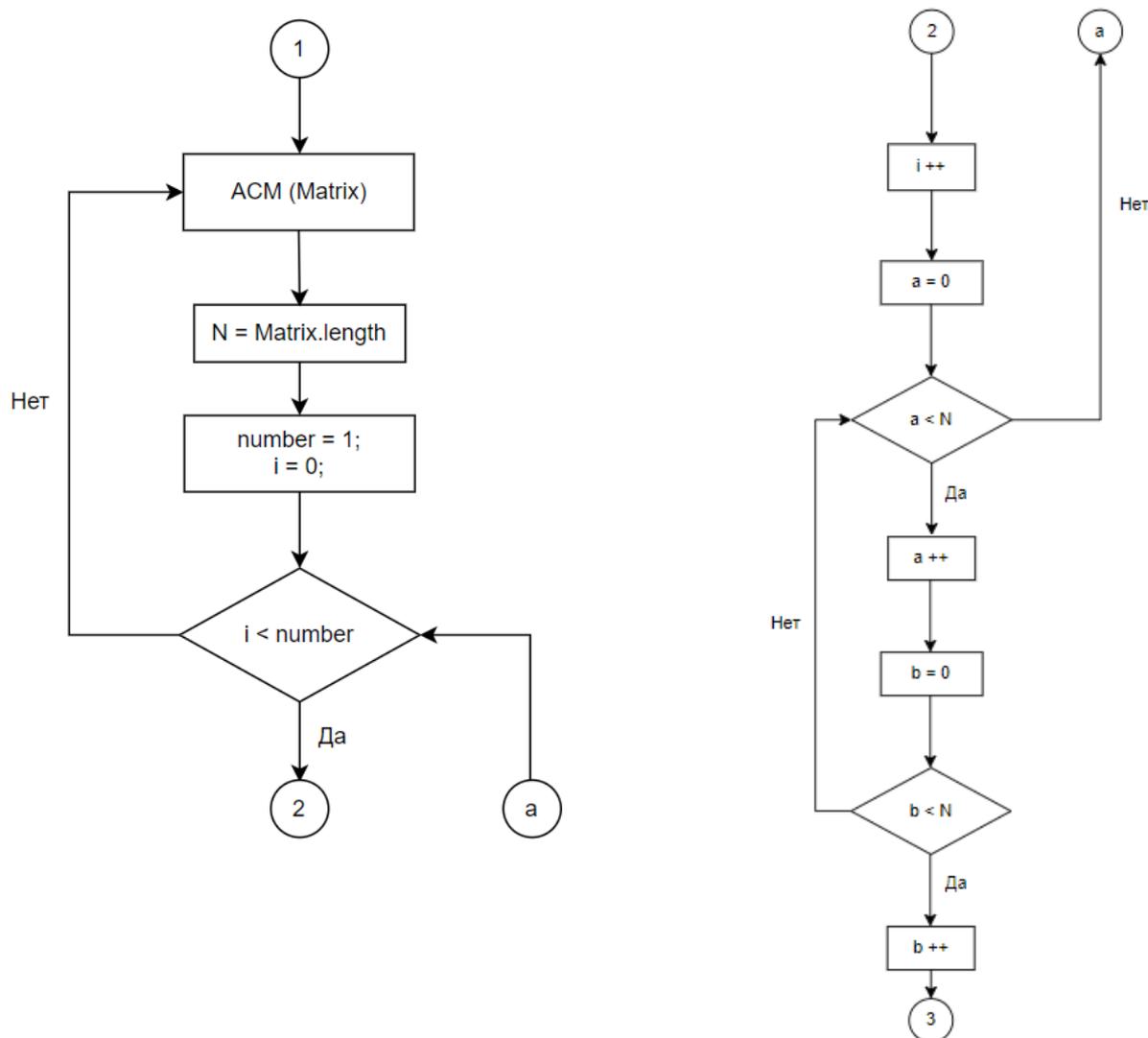


Рис. 12. Цикл, преобразующий исходное изображение в матрицу

На рисунке 12 показана блок-схема, описывающая работу метода ACM(). Этот метод маскирует исходное изображение под шум.

```

...
Integer[] hasil = new Integer[2];
Integer[][] result = new Integer[N][N];
int number;
...
number = 1;
for (int i = 0; i < number; i++) {
for (int a = 0; a < N; a++) {
for (int b = 0; b < N; b++) {
hasil[0] = (1*a + 1*b) % N;
    hasil[1] = (1*a + 2*b) % N;
result[a][b] = matrix[hasil[0]][hasil[1]];
...
Integer[] hasil = new Integer[2];
Integer[][] result = new Integer[N][N];
int number;
  
```

```

...
number = 1;
for (int i = 0; i < number; i++) {
for (int a = 0; a < N; a++) {
for (int b = 0; b < N; b++) {
hasil[0] = (1*a + 1*b) % N;
    hasil[1] = (1*a + 2*b) % N;
result[a][b] = matrix[hasil[0]][hasil[1]];
  
```

На рисунке 13 показан вывод преобразованного изображения в пользовательский интерфейс.

```

converted image to the user interface.
for (int i = 0; i < result Image.getWidth(); i++) {
for (int j = 0; j < resultImage.getHeight(); j++) {
resultImage.setRGB(i, j, result[i][j]);
...
this.result Image Panel.setImage(resultImage);
...
  
```

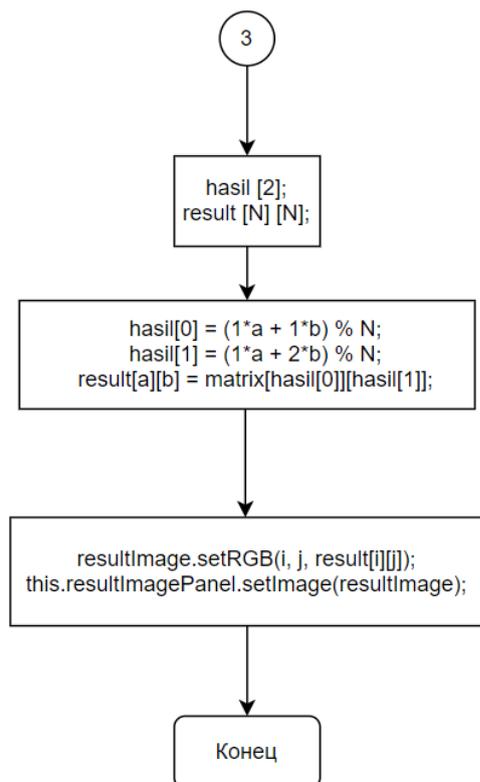


Рис. 13. Вывод полученного изображения на панель

#### СКРИНШОТЫ ПРОГРАММНОГО ПАКЕТА

На рисунке 14 показан пользовательский интерфейс разработанного программного пакета. После запуска программы пользователь попадает в главное окно приложения.

После нажатия на кнопку «Обзор» (Browse) обработчик событий отобразит окно меню на главной панели. Пользователю будет предложено выбрать объект преобразования из файловой системы персонального компьютера (рис. 15).

Предположим, что пользователь выбрал изображение, показанное на рисунке 16. Выбранное изображение будет отображаться в левой части главной панели пользователя.

В нижней части окна находится панель инструментов. Пустое поле служит для ввода количества итераций. После ввода числа в это поле программа начнет преобразование изображения и отобразит результат в правой части панели. При нажатии на стрелки, указывающие вправо и влево, будет выполнена одна итерация преобразования изображения.

На рисунке 17 показано состояние панели изображений после 114 итераций. В правой части панели рисунок превратился в почти естественный шум.

На рисунке 18 пользователь повторно ввел количество итераций 114 в поле ввода, и результат работы программы отображается в правой части панели. Следует отметить, что после 228 итераций это изображение принимает свой первоначальный вид. Таким образом, процесс трансформации является обратимым.



Рис. 14. Программный интерфейс начального пользовательского окна

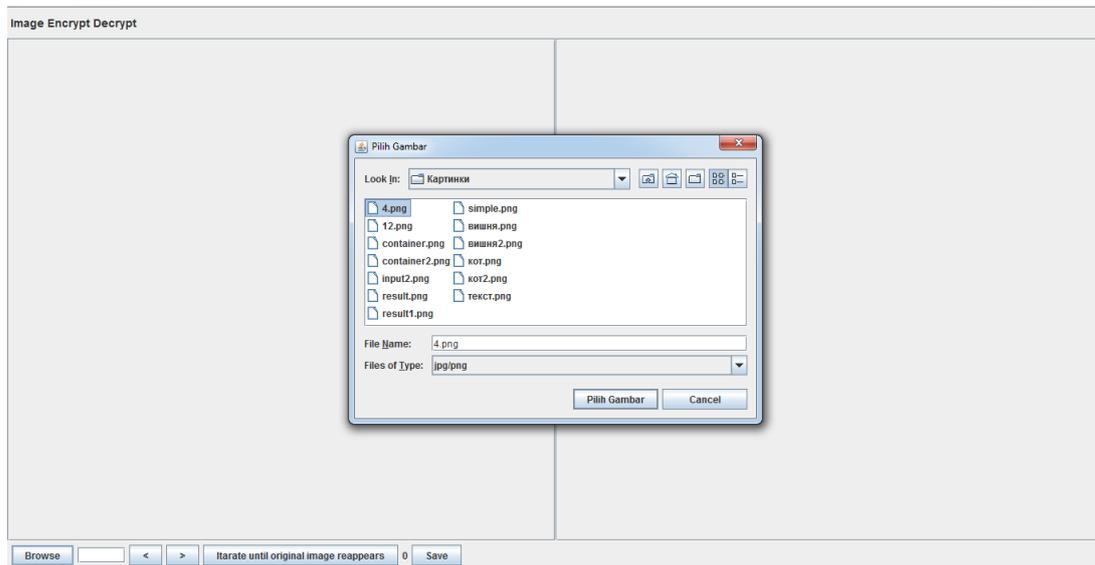


Рис. 15. Выбор изображения



Рис. 16. Исходное изображение

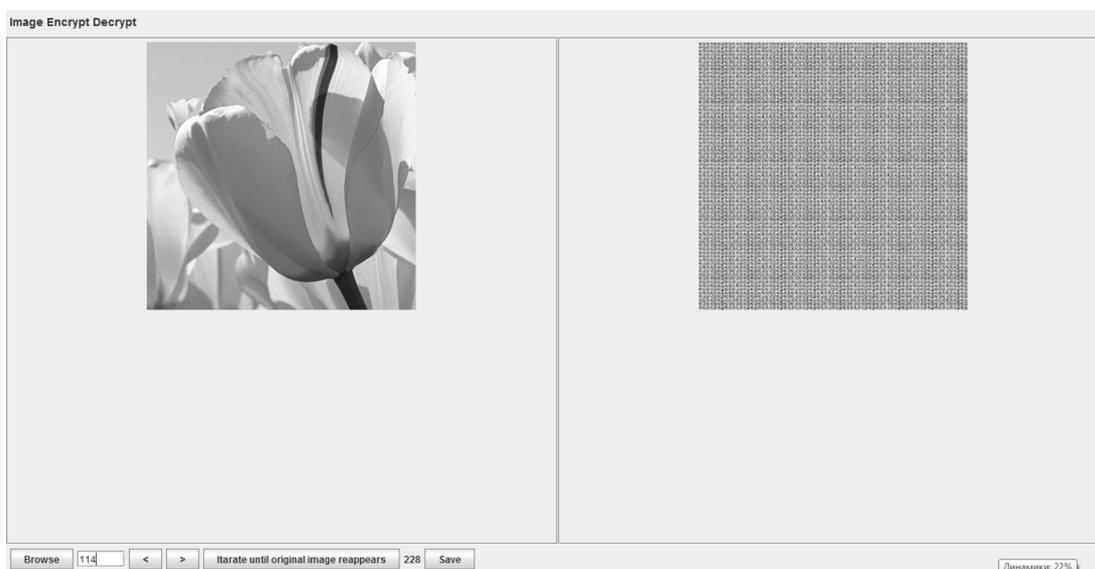


Рис. 17. Состояние панели изображений после 114 итераций

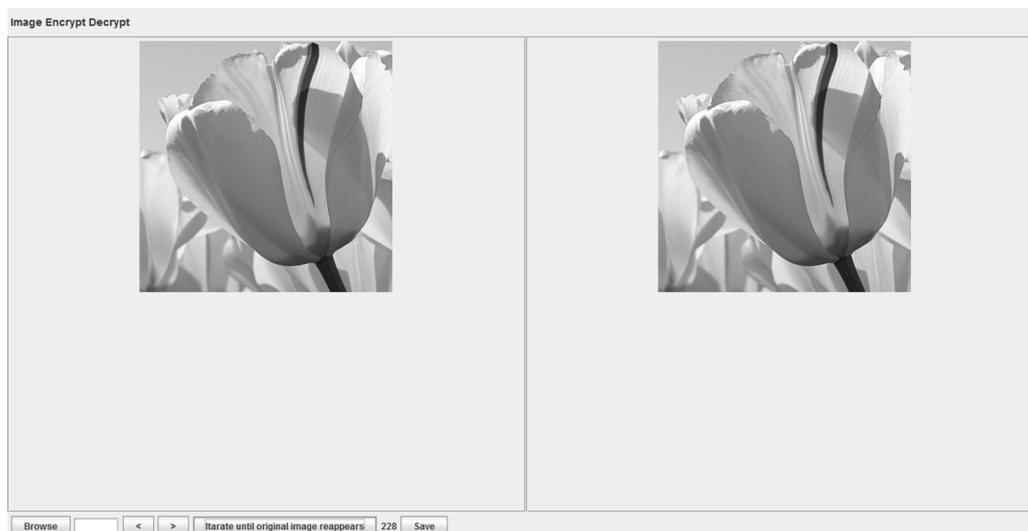


Рис. 18. Состояние панели изображений после 228 итераций

#### ЗАКЛЮЧЕНИЕ

В данной статье рассмотрена и решена задача оптимальной маскировки скрытого сообщения под естественный шум в канале связи. Для этого исследованы два дискретных хаотических преобразования — ACM и VM. В качестве меры хаотичности предлагается использовать коэффициент хаотичности, который легко вычисляется и сопоставим с энтропией. Задача оптимизации маскировки скрытого сообщения, таким образом, сводится к нахождению итерации, при которой коэффициент хаотичности принимает максимальное значение. Конечным результатом работы является разработка программного пакета для дискретных преобразований скрытых сообщений, представленных неподвижными изображениями, в форму, неотличимую от естественного шума. Разработанный программный пакет тщательно проиллюстрирован скриншотами.

#### ЛИТЕРАТУРА

1. Benguigui, L. The Different Paths to Entropy // *European Journal of Physics*. 2013. Vol. 34, No. 2. Pp. 303–321. DOI: 10.1088/0143-0807/34/2/303.

2. Peng, H. Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy / H. Peng, F. Long, C. Ding // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2005. Vol. 27, Is. 8. Pp. 1226–1238. DOI: 10.1109/TPAMI.2005.159.

3. Pál, D. Estimation of Rényi Entropy and Mutual Information Based on Generalized Nearest-Neighbor Graphs / D. Pál, B. Póczos, C. Szepesvári // *Advances in Neural Information Processing Systems 23 (NIPS 2010): Proceedings of the 24th Annual Conference on Neural Information Processing Systems (Vancouver, Canada, 06–09 December 2010)*. — New York: Curran Associates Inc., 2019. — Pp. 1849–1857.

4. Archer, E. W. Bayesian Entropy Estimation for Countable Discrete Distributions / E. W. Archer, I. M. Park, J. W. Pillow // *The Journal of Machine Learning Research*. 2014. Vol. 15. Pp. 2833–2868.

5. Hall, P. On the Estimation of Entropy / P. Hall, S. C. Morton // *Annals of the Institute of Statistical Mathematics*. 1993. Vol. 45, No. 1. Pp. 69–88. DOI: 10.1007/BF00773669.

6. Miller, E. G. A New Class of Entropy Estimators for Multi-Dimensional Densities // *Proceedings of the 28th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03), (Hong Kong, China, 06–10 April 2003)*. — Vol. 3. — IEEE, 2003. — Pp. 297–300. DOI: 10.1109/ICASSP.2003.1199463.

7. Sricharan, K. Ensemble Estimators for Multivariate Entropy Estimation / K. Sricharan, D. Wei, A. O. Hero III // *IEEE Transactions on Information Theory*. 2013. Vol. 59, Is. 7. Pp. 4374–4388. DOI: 10.1109/TIT.2013.2251456.

# A Software Model of Masking Hidden Message in Steganography Tasks

Grand PhD V. N. Kustov, A. I. Grokhotov, E. V. Golovkov  
Emperor Alexander I St. Petersburg State Transport University  
Saint Petersburg, Russia  
kvnvika@mail.ru, grokhotov.aleksei@mail.ru, jyk22@mail.ru

**Abstract.** The authors consider the problem of developing a software model of masking a hidden message in HUGO stegosystems under natural noise in the communication channel using discrete chaotic Arnold cat map and Baker map, which are iterative reversible discrete chaotic transformations. All stages of software model development are considered in detail, such as the choice of the development environment, the composition of software modules and their purpose, screenshots of graphical interfaces for various modes of operation of the model. In the software model to estimate the chaotic state of a hidden message represented by a digital still image, the authors introduce the chaotic coefficient, a numerical indicator of the entropy of the probability of disordered pixels. In the developed software model, the authors propose a method for determining the maximum value of the chaotic coefficient corresponding to the maximum chaotic state of the hidden image. Testing the software model shows the practical indistinguishability of the transformed hidden message from the natural noise in the communication channel and significantly increases the safety of HUGO stegosystems.

**Keywords:** software model, HUGO stegosystem, Arnold's cat map, Baker's map, Pearson correlation coefficient, natural noise, chaotic coefficient, chaotic maps, steganology.

## REFERENCES

1. Benguigui L. The Different Paths to Entropy, *European Journal of Physics*, 2013, Vol. 34, No. 2, Pp. 303–321. DOI: 10.1088/0143-0807/34/2/303.
2. Peng H., Long F., Ding C. Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, Vol. 27, Is. 8, Pp. 1226–1238. DOI: 10.1109/TPAMI.2005.159.

3. Pál D., Póczos B., Szepesvári C. Estimation of Rényi Entropy and Mutual Information Based on Generalized Nearest-Neighbor Graphs, *Advances in Neural Information Processing Systems 23 (NIPS 2010): Proceedings of the 24th Annual Conference on Neural Information Processing Systems, Vancouver, Canada, December 06–09, 2010*. New York, Curran Associates Inc., 2019, Pp. 1849–1857.

4. Archer E. W., Park I. M., Pillow J. W. Bayesian Entropy Estimation for Countable Discrete Distributions, *The Journal of Machine Learning Research*, 2014, Vol. 15, Pp. 2833–2868.

5. Hall P., Morton S. C. On the Estimation of Entropy, *Annals of the Institute of Statistical Mathematics*, 1993, Vol. 45, No. 1, Pp. 69–88. DOI: 10.1007/BF00773669.

6. Miller E. G. A New Class of Entropy Estimators for Multi-Dimensional Densities, *Proceedings of the 28th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03), Hong Kong, China, April 06–10, 2003, Vol. 3*. IEEE, 2003, Pp. 297–300. DOI: 10.1109/ICASSP.2003.1199463.

7. Sricharan K., Wei D. Hero III A. O. Ensemble Estimators for Multivariate Entropy Estimation, *IEEE Transactions on Information Theory*, 2013, Vol. 59, Is. 7, Pp. 4374–4388. DOI: 10.1109/TIT.2013.2251456.

# Использование моделей принятия решений для оптимизации выбора технологий производства геоэкозащитных материалов

к.т.н. Е. В. Русанова, Е. В. Рунев

Петербургский государственный университет путей сообщения Императора Александра I  
Санкт-Петербург, Россия  
rusanovaev@mail.ru, jr\_2010@mail.ru

**Аннотация.** Проблема выбора материалов при использовании в геоэкозащитных технологиях актуальна. В работе предложена математическая модель выбора наилучшего доступного материала, основанная на применении критериев оптимальности, заданных на конечных множествах, — факторов (технических характеристик объектов) и альтернатив (материалов различного химического состава). На основе этой модели разработана методика расчета, позволяющая стандартизировать материалы, которые используются в геоэкозащитных технологиях в соответствии с техническими требованиями, предъявляемыми заказчиком. Эта методика дает возможность принимать решения на основе мониторинга данных использования технологий в течение длительного периода времени. Данные решения, как правило, имеют долгосрочные последствия на экологическую безопасность и экономическую эффективность.

**Ключевые слова:** оптимальные стратегии, множество состояний, множество альтернатив, критерий Гурвица, априорное вероятностное распределение, критерий максимума ожидаемой полезности, критерий Ходжеса — Лемана, наилучшая доступная технология, золопенобетон.

## ВВЕДЕНИЕ

Выбор доступной и эффективной технологии имеет огромное значение в современном мире развивающихся малоотходных технологий. Это особенно важно на фоне кризисных явлений в экономическом развитии транспортной инфраструктуры. Выбор технологии осуществляется по нескольким критериям. Задача может не иметь оптимального решения в чистых стратегиях. Различные подходы существуют при выборе оптимального решения.

Авторы предлагают использовать стратифицированный метод принятия решений на основе широкого круга критериев (МДСМ) [1]. Этот метод позволяет принимать решения в долгосрочной перспективе. Метод стратифицированного МДСМ в сочетании с маркированным стратифицированным методом (БВМ) может использоваться для расчета ранжирования имеющихся технологий. Но метод не указывает множество оптимальных стратегий.

Авторами предлагается использовать метод группового анализа мультиатрибутного решения на основе теории интервальных нечетких множеств, что может успешно помочь заинтересованным сторонам определить приоритетную последовательность альтернатив-

ных сценариев [2–4]. Этот метод анализа также не дает множество оптимальных стратегий.

В работе продемонстрировано применение математической модели выбора наилучшей доступной технологии, основанной на применении критериев оптимальности, заданных на конечных множествах: факторов (технических характеристик объектов) и альтернатив (материалов различного состава). При этом функции полезности критериев оптимального выбора рассматриваются в следующих ситуациях [5–7]:

- модель с неполной информацией;
- модель в условиях риска.

Указанные критерии применены для материалов из золопенобетона, которые отличаются плотностью и содержанием золы от сжигания осадка сточных вод [8–11]. Определение оптимального состава золопенобетона актуально при разработке конструкций шумозащитных экранов на железнодорожном транспорте, а также в других областях инженерного строительства.

Предложенный алгоритм моделирования позволяет сравнить материалы и построить иерархическую лестницу, на основании которой возможно в кратчайшие сроки принять решение о применении той или иной технологии в зависимости от заданных условий эксплуатации.

## ПОСТАНОВКА ЗАДАЧИ

Рассматривается множество  $S = \{s_1, \dots, s_n\}$  — конечное множество факторов по группам: экологическая, технологическая, эксплуатационная ( $n = 8$ ). Каждый фактор  $s_j$  из множества  $S$  имеет влияние на выбор материала, применяемого для технологии производства  $d_k$  из множества  $D$ . Множество  $D$  является конечным множеством технологий (альтернатив)  $D = \{d_1, \dots, d_m\}$ , где  $m = 4$ :

- $d_1$  — зола от сжигания осадка сточных вод;
- $d_2$  — автоклавный золопенобетон D500 плотностью 500 кг/м<sup>3</sup>;
- $d_3$  — автоклавный золопенобетон D600 плотностью 600 кг/м<sup>3</sup>;
- $d_4$  — автоклавный золопенобетон D800 плотностью 800 кг/м<sup>3</sup>.

В таблице 1 представлены экспериментальные измерения по каждому представленному виду материала, приведенные к безразмерным величинам.

Экспериментальные исследования образцов рассматриваемых материалов

Геоэкозащитный материал	Содержание естественных радионуклеидов	Концентрация пыли	Теплопроводность	Прочность	Морозостойкость	Содержание золы	Звукозащита	Снижение шума
	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$	$s_8$
Зола ( $d_1$ )*	0,277	0,244	0,000	0,000	0,000	0,000	0,000	0,000
D500 ( $d_2$ )	0,964	1,000	1,000	0,356	0,213	0,312	0,936	0,868
D600 ( $d_3$ )	0,961	1,000	0,962	0,500	0,524	0,437	0,944	0,876
D800 ( $d_4$ )	0,959	1,000	0,740	0,842	0,860	0,685	0,956	0,891

Примечание: \* — технические испытания не проводились, дано справочно.

Функция полезности определяется как отображение

$$h: D \times S \rightarrow R,$$

сопоставляющее каждому фактору  $s_j \in S$  и каждой альтернативе  $d_i \in D$  выигрыш  $h(d_i, s_j) = h_{ij}$  для конечных или счетных  $D$  и  $S$ .

Наряду с принятием решения в виде чистой стратегии часто имеет смысл задача поиска оптимального решения в смешанных стратегиях. Эта стратегия реализуется в виде вектора

$$p = (p_1, p_2, \dots, p_m),$$

компоненты которого удовлетворяют условиям:

$$\sum_{k=1}^m p_k = 1, \quad p_k \geq 0.$$

В этом случае полезность принятого решения определяется как математическое ожидание случайной величины — полезности с дискретным распределением

$$p = (p_1, p_2, \dots, p_m), \quad p_j = p(d_j), \quad j = 1, \dots, m:$$

$$h(p, s) = \sum_{k=1}^m p_k \times h(d_k, s)$$

Смешанная стратегия может реализовываться различными способами: физическим смешиванием, априорным вероятностным распределением, статистическим (частота при многократном выборе).

МОДЕЛИ ОПТИМАЛЬНОСТИ С НЕПОЛНОЙ ИНФОРМАЦИЕЙ

Модель принятия решения с неполной информацией предполагает выполнение следующих условий:

- множество факторов  $S$  является полным, то есть включает в себя все возможные факторы данной модели;
- один из факторов множества  $S$  является решающим, то есть его влияние оказывает решающее действие на принятие решения.

Задача лица, принимающего решение, состоит в том, чтобы выбрать какую-либо альтернативу из множества  $D$ . Каждая альтернатива есть чистая стратегия. Ниже рассмотрены основные критерии принятия решения в моделях с неполной информацией. Результатом применения этих критериев является чистая стратегия.

ПРИНЯТИЕ РЕШЕНИЯ НА ОСНОВЕ

ПАРАМЕТРИЧЕСКОГО КРИТЕРИЯ ГУРВИЦА

Рассматривается функционал Гурвица [7, 12, 13]:

$$G_\alpha[d] = \alpha \times \underline{h}(d) + (1 - \alpha) \times \bar{h}(d), \quad \alpha \in [0; 1],$$

где  $\bar{h}(d) = \max_{s \in S} h(d, s)$  — верхнее значение функции полезности;

$\underline{h}(d) = \min_{s \in S} h(d, s)$  — нижнее значение функции полезности.

Задача оптимизации имеет следующий вид:

$$G_\alpha[d] \rightarrow \max_{d \in D}, \quad \alpha \in [0; 1].$$

Верхние значения функции полезности для альтернатив из множества  $D$ :

$$\bar{h}(d_1) = 0,277; \quad \bar{h}(d_2) = 1,000;$$

$$\bar{h}(d_3) = 1,000; \quad \bar{h}(d_4) = 1,000.$$

Нижние значения функции полезности для альтернатив из множества  $D$ :

$$\underline{h}(d_1) = 0,000; \quad \underline{h}(d_2) = 0,213;$$

$$\underline{h}(d_3) = 0,437; \quad \underline{h}(d_4) = 0,685.$$

Для частных значений параметра  $\alpha \in [0,1]$  имеем следующие критерии:

- при  $\alpha = 1$  — критерий максимина (Вальда);
- при  $\alpha = 0$  — критерий максимакса.

Значения функционала критерия Гурвица для элементов множества  $D$  как функции параметра  $\alpha \in [0,1]$  получены:

$$d_1: f_1(\alpha) = 0,277 \times \alpha;$$

$$d_2: f_2(\alpha) = 0,787 \times \alpha + 0,213;$$

$$d_3: f_3(\alpha) = 0,563 \times \alpha + 0,437;$$

$$d_4: f_4(\alpha) = 0,315 \times \alpha + 0,685.$$

Из приведенных уравнений видно, что все зависимости имеют линейный характер.

На рисунке 1 даны графики зависимости функционала Гурвица от параметра  $\alpha \in [0,1]$  для всех альтернатив из множества  $D$ .

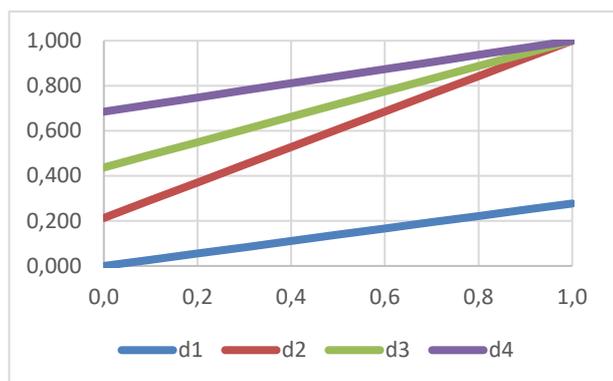


Рис. 1. Графики зависимости функционала Гурвица от параметра  $\alpha$

Видно, что при  $\alpha = 1$  оптимальная стратегия не единственна:  $d^* = d_2 = d_3 = d_4$ , но при  $\alpha \in (0,1)$  имеем единственную оптимальную стратегию:  $d^* = d_4$ .

ПРИНЯТИЕ РЕШЕНИЯ НА ОСНОВЕ КРИТЕРИЯ ЛАПЛАСА

Задача оптимизации на основе критерия Лапласа имеет следующий вид:

$$L[d] = \sum_{i=1}^n h(d, s_i) \rightarrow \max_{d \in D}$$

Ниже приведены значения функционала Лапласа для всех альтернатив из множества  $D$ :

$$L[d_1] = \sum_{j=1}^8 h(d_1, s_j) = 0,521 ; L[d_2] = \sum_{j=1}^8 h(d_2, s_j) = 5,649 ;$$

$$L[d_3] = \sum_{j=1}^8 h(d_3, s_j) = 6,204 ; L[d_4] = \sum_{j=1}^8 h(d_4, s_j) = 6,933 .$$

Видно, что

$$\max_{1 \leq i \leq m} L[d_i] = \max_{1 \leq i \leq m} \left( \sum_{j=1}^8 h(d_i, s_j) \right) = 6,933 .$$

Таким образом, согласно критерию Лапласа оптимальной стратегией является  $d^* = d_4$  — технология, использующая золопенобетон D800.

ПРИНЯТИЕ РЕШЕНИЯ НА ОСНОВЕ КРИТЕРИЯ ВАЛЬДА

Задача оптимизации на основе критерия Вальда имеет следующий вид:

$$\underline{h}(d) = \min_{s \in S} h(d, s) \rightarrow \max_{d \in D}$$

$$\underline{h}_1 = 0; \underline{h}_2 = 0,213; \underline{h}_3 = 0,437; \underline{h}_4 = 0,685 .$$

Таким образом,  $\max(\underline{h}_1; \underline{h}_2; \underline{h}_3; \underline{h}_4) = 0,685$ .

Согласно критерию Вальда оптимальной стратегией является  $d^* = d_4$  — технология, использующая золопенобетон D800.

ПРИНЯТИЕ РЕШЕНИЯ НА ОСНОВЕ

КРИТЕРИЯ СЭВИДЖА (КРИТЕРИЙ МИНИМАКСА)

Сожаление (риск) в теории принятия решений — потеря в результате упущенных возможностей [14–16]:

$$l(s, d) = \bar{h}(s) - h(s, d) ,$$

где  $\bar{h}(s) = \max_{d \in D} h(d, s)$ .

В этом случае задача оптимизации имеет следующую постановку:

$$S[d] = \max_{s \in S} l(s, d) \rightarrow \min_{d \in D}$$

$l_{ij} = \bar{h}(s_j) - h(d_i, s_j)$  — элементы матрицы сожалений Сэвиджа (риски, потери вследствие отсутствия информации о факторе) — представлены в таблице 2.

Вид матрицы сожалений  $l_{ij} = \bar{h}_j - h_{ij}$  представлен в таблице 3.

Таблица 2

Максимальные значения по каждому фактору из множества  $S$

	$h_{11}$	$h_{12}$	$h_{13}$	$h_{14}$	$h_{15}$	$h_{16}$	$h_{17}$	$h_{18}$
$h_{1j}$	0,277	0,244	0,000	0,000	0,000	0,000	0,000	0,000
$h_{2j}$	0,964	1,000	1,000	0,356	0,213	0,312	0,936	0,868
$h_{3j}$	0,961	1,000	0,962	0,500	0,524	0,437	0,944	0,876
$h_{4j}$	0,959	1,000	0,740	0,842	0,860	0,685	0,956	0,891
$\max_{d \in D} h(d, s)$	<b>0,964</b>	<b>1,000</b>	<b>1,000</b>	<b>0,842</b>	<b>0,860</b>	<b>0,685</b>	<b>0,956</b>	<b>0,891</b>

Таблица 3

Максимальные значения по каждой альтернативе из множества  $D$

	$l_{11}$	$l_{12}$	$l_{13}$	$l_{14}$	$l_{15}$	$l_{16}$	$l_{17}$	$l_{18}$	$\max_{s \in S} l(s, d)$
$l_{1j}$	0,687	0,756	1,000	0,842	0,860	0,685	0,956	0,891	<b>1,000</b>
$l_{2j}$	0,000	0,000	0,000	0,486	0,647	0,373	0,020	0,023	<b>0,647</b>
$l_{3j}$	0,003	0,000	0,038	0,342	0,336	0,248	0,012	0,015	<b>0,342</b>
$l_{4j}$	0,005	0,000	0,260	0,000	0,000	0,000	0,000	0,000	<b>0,260</b>

Таким образом,  $\min(1,000; 0,647; 0,342; 0,260) = 0,260$ . Согласно критерию Сэвиджа минимум сожалений достигается на альтернативе  $d^* = d_4$  — технологии, использующей золопенобетон D800.

**МОДЕЛИ ОПТИМАЛЬНОСТИ В УСЛОВИЯХ РИСКА**

Задача принятия решений в условиях риска возникает, если известно априорное распределение [17, 18]

$$p = (p_1, p_2, \dots, p_m), \quad p_j = P(s_j)$$

вероятностей влияния факторов из множества  $S$ .

В случае конечного множества факторов  $S$  ожидаемая полезность альтернативы  $d$  определяется как математическое ожидание полезности, соответствующей альтернативе  $d$ :

$$H(d, p) = E_p h = \sum_{j=1}^n p_j \times h(d, s_j).$$

Этот функционал положен в основу критериев оптимальности в условиях риска.

Выбор оптимальной стратегии в условиях риска осуществляется с помощью следующих критериев [19]:

- максимума ожидаемой полезности,
- Ходжеса — Лемана,
- наивероятнейшего фактора,
- минимума ожидаемых сожалений,
- минимума дисперсии полезности.

Критерии выбора оптимальной стратегии в условиях риска представлены в таблице 4.

Таблица 4

Критерии выбора оптимальной стратегии в условиях риска

Критерий	Максимума ожидаемой полезности	Ходжеса — Лемана	Наивероятнейшего фактора	Минимума ожидаемых сожалений
Функционал критерия	$H(d, p)$	$HL_p[d] = \alpha \times \underline{h}(d) + (1 - \alpha) \times H(d, p),$ $\underline{h}(d) = \min_{s \in S} h(d, s)$	$p(s) \rightarrow \max_{s \in S}$	$L_p[d] = \sum_{j=1}^n p_j \times l(s_j, d),$ $l(s_j, d) = \bar{h}(s_j) - h(s_j, d)$
Условие оптимальности	$H(d, p) \rightarrow \max_{d \in D}$	$HL_p[d] \rightarrow \max_{d \in D}$	$h(d, s^*) \rightarrow \max_{d \in D}$	$L_p[d] \rightarrow \min_{d \in D}$

**АПРИОРНОЕ РАСПРЕДЕЛЕНИЕ  
ВЕРОЯТНОСТЕЙ ВЛИЯНИЯ ФАКТОРОВ**

Построим априорное вероятностное распределение на примере разбиения факторов на три группы факторов, как указано в таблице 5. Введем в рассмотрение полную группу событий  $\{A_k\}_{k=1}^l$ , где  $l$  — количество групп факторов. При этом выполнено очевидное равенство:

$$\sum_{k=1}^l P(A_k) = 1.$$

В каждой группе факторов выделим подгруппы событий:  $\{A_k^j\}_{j=1}^{i_k}$ , где  $i_k$  — количество подгрупп в  $k$ -й группе. Используя теорему умножения вероятностей событий, независимых в совокупности, найдем априорное дискретное распределение влияния факторов:

$$p(s_j) = P(A_k \cap A_k^j) = P(A_k) \times P(A_k^j).$$

Окончательно априорное распределение вероятностей представлено в таблице 6.

Таблица 5

Распределение вероятностей влияния факторов из множества  $S$  по группам

I. Environmental group		II. Technological group					III. Operational group
0,40		0,30					0,30
0,90	0,10	0,02	0,13	0,15	0,35	0,35	1,00

Таблица 6

Априорное вероятностное распределение факторов из множества  $S$

$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$p_6$	$p_7$	$p_8$
0,360	0,040	0,006	0,039	0,045	0,105	0,105	0,300

ПРИНЯТИЕ РЕШЕНИЯ НА ОСНОВЕ КРИТЕРИЯ  
МАКСИМАЛЬНОСТИ ОЖИДАЕМОЙ ПОЛЕЗНОСТИ

В таблице 7 приведены результаты применения критерия максимума ожидаемой полезности к альтернативам множества  $D$ .

Согласно критерию оптимальной стратегией является альтернатива  $d^* = d_4$ , соответствующая золопенобетону плотностью 800 кг/м<sup>3</sup>.

Таблица 7

Значения ожидаемой полезности на множестве альтернатив

Ожидаемая полезность на множестве альтернатив	$(p_j \times h_{ij})$								Значение
$H(d_1, p)$	0,100	0,010	0	0	0	0	0	0	<b>0,109</b>
$H(d_2, p)$	0,347	0,040	0,006	0,014	0,010	0,033	0,098	0,260	<b>0,808</b>
$H(d_3, p)$	0,346	0,040	0,006	0,020	0,024	0,046	0,099	0,263	<b>0,843</b>
$H(d_4, p)$	0,345	0,040	0,004	0,033	0,039	0,072	0,100	0,267	<b>0,901</b>

ПРИНЯТИЕ РЕШЕНИЯ

НА ОСНОВЕ КРИТЕРИЯ ХОДЖЕСА — ЛЕМАНА

В таблице 8 показаны результаты применения критерия Ходжеса — Лемана к альтернативам множества  $D$ .

Таким образом, зависимости функционала Ходжеса-Лемана от параметра  $\alpha \in [0,1]$  для альтернатив из множества  $D$ :

$$d_1: f_1(\alpha) = 0,109 - 0,109 \times \alpha;$$

$$d_2: f_2(\alpha) = 0,808 - 0,595 \times \alpha;$$

$$d_3: f_3(\alpha) = 0,843 - 0,406 \times \alpha;$$

$$d_4: f_4(\alpha) = 0,901 - 0,216 \times \alpha.$$

Все зависимости имеют линейный характер, их графики приведены на рисунке 2.

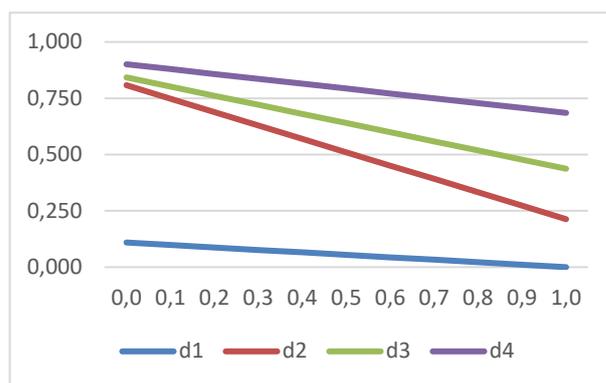


Рис. 2. Графики зависимости функционала Ходжеса-Лемана от параметра  $\alpha$

Видно, что при  $\alpha = 1$  оптимальная стратегия в условиях риска, согласно критерию Ходжеса-Лемана,  $d^* = d_4$ , при этом  $HL_p[d_4] = 0,901$ .

Таблица 8

Минимальные значения по каждой альтернативе из множества  $D$

	$h_{i1}$	$h_{i2}$	$h_{i3}$	$h_{i4}$	$h_{i5}$	$h_{i6}$	$h_{i7}$	$h_{i8}$	$\min_{s \in S} h(d, s)$
$h_{1j}$	0,277	0,244	0,000	0,000	0,000	0,000	0,000	0,000	<b>0,000</b>
$h_{2j}$	0,964	1,000	1,000	0,356	0,213	0,312	0,936	0,868	<b>0,213</b>
$h_{3j}$	0,961	1,000	0,962	0,500	0,524	0,437	0,944	0,876	<b>0,437</b>
$h_{4j}$	0,959	1,000	0,740	0,842	0,860	0,685	0,956	0,891	<b>0,685</b>

Таблица 9

ПРИНЯТИЕ РЕШЕНИЯ НА ОСНОВЕ КРИТЕРИЯ  
НАИВЕРОЯТНЕЙШЕГО ФАКТОРА

В таблицах 9 и 10 приведены результаты применения критерия наиболее вероятного фактора к альтернативам множества  $D$ .

Видно, что оптимальная стратегия в условиях риска, согласно критерию наиболее вероятного фактора,  $d^* = d_2$ .

Группы факторов  
и их вероятностное распределение

I. Environmental group	$p_1 = 0,360$
	$p_2 = 0,040$
II. Technological group	$p_3 = 0,006$
	$p_4 = 0,039$
	$p_5 = 0,045$
	$p_6 = 0,105$
	$p_7 = 0,105$
III. Operational group	$p_8 = 0,300$

Применение критерия наивероятнейшего фактора к альтернативам множества  $D$

	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$	$s_8$
$d_1$	0,277	0,244	0,000	0,000	0,000	0,000	0,000	0,000
$d_2$	<b>0,964</b>	1,000	1,000	0,356	0,213	0,312	0,936	0,868
$d_3$	0,961	1,000	0,962	0,500	0,524	0,437	0,944	0,876
$d_4$	0,959	1,000	0,740	0,842	0,860	0,685	0,956	0,891

ПРИНЯТИЕ РЕШЕНИЯ НА ОСНОВЕ КРИТЕРИЯ МИНИМАЛЬНОСТИ ОЖИДАЕМЫХ СОЖАЛЕНИЙ

На рисунке 3 приведен график минимальных значений из найденных ожидаемых сожалений  $L(d, p)$ .

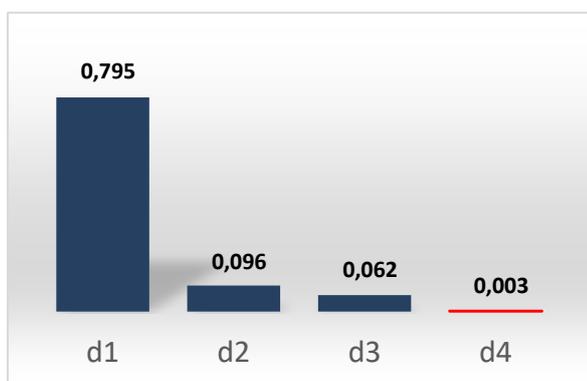


Рис. 3. Значения функционала критерия минимального ожидаемого сожаления на альтернативах из множества  $D$

Из графика видно, что согласно критерию минимума ожидаемых сожалений оптимальной является технология, использующая золопенобетон плотностью  $800 \text{ кг/м}^3$  —  $d^* = d_4$ .

ПРИНЯТИЕ РЕШЕНИЯ НА ОСНОВЕ КРИТЕРИЯ МИНИМАЛЬНОСТИ ДИСПЕРСИИ ПОЛЕЗНОСТИ

В основе критерия лежит квадратичный функционал дисперсии [20–23]:

$$D_p[d] := \sum_{j=1}^n p_j \times (h(d, s_j) - H(d, p))^2.$$

Задача оптимизации имеет вид:

$$D_p[d] \rightarrow \min_{d \in D}.$$

Результаты применения критерия приведены в таблице 11.

Применение критерия минимальности дисперсии полезности к альтернативам множества  $D$

	$p_j \times (h(d, s_j) - H(d, p))^2$								Значение
$D_p[d_1]$	0,010	0,001	0,000	0,000	0,001	0,001	0,001	0,004	<b>0,018</b>
$D_p[d_2]$	0,009	0,001	0,000	0,008	0,016	0,026	0,002	0,001	<b>0,063</b>
$D_p[d_3]$	0,005	0,001	0,000	0,005	0,005	0,017	0,001	0,000	<b>0,034</b>
$D_p[d_4]$	0,001	0,000	0,000	0,000	0,000	0,005	0,000	0,000	<b>0,007</b>

Согласно критерию минимальности дисперсии полезности оптимальной является альтернатива, золопенобетон плотностью  $800 \text{ кг/м}^3$  —  $d^* = d_4$ .

ЗАКЛЮЧЕНИЕ

В работе продемонстрировано применение моделей принятия решений в условиях неполной информации и риска для случая, когда множество факторов и множество альтернатив конечны. Критерии оптимальности в условиях риска и в условиях неопределенности показали, что оптимальной стратегией является альтернатива использования материала, произведенного из золопенобетона плотностью  $800 \text{ кг/м}^3$ . Указанные оптимальные решения получены в чистых стратегиях.

Результаты применения большого количества критериев в условиях неполной информации и риска сделали

возможным создание комплекса программ для мониторинга и анализа данных по разным материалам и технологиям. Это позволяет обрабатывать информацию, поступающую от производителей и потребителей технологий, с целью принятия решения об использовании в производстве той или иной технологии или материала в короткие временные интервалы.

Следует отметить, что указанные во второй части работы критерии оптимальности в условиях риска допускают обобщение в случае, если априорное распределение вероятностей влияния факторов — непрерывное. Это позволяет расширить применение метода до систем с непрерывно меняющимися значениями факторов. В качестве значений случайных величин факторов моделей принимаются значения характеристик, которые разбиваются на интервалы. Такая классификация и вы-

бор является детализацией имеющейся дискретной модели с точечными значениями (оценками факторов). В дальнейшем планируется развивать указанный подход в задачах оптимального выбора из доступных технологий в различных отраслях на транспорте.

ЛИТЕРАТУРА

1. Torkayesh, A. E. Sustainable Waste Disposal Technology Selection: The Stratified Best-Worst Multi-Criteria Decision-Making Method / A. E. Torkayesh, B. Malmir, M. R. Asadabadi // *Waste Management*. 2021. Vol. 122. Pp. 100–112. DOI: 10.1016/j.wasman.2020.12.040.

2. Чепаченко, Н. В. Обоснование выбора приоритетного инновационного решения в строительстве с применением аппарата теории игр / Н. В. Чепаченко, М. М. Луценко, А. М. Дёмин // *Вестник гражданских инженеров*. 2020. № 3 (80). С. 238–244. DOI: 10.23968/1999-5571-2020-17-3-238-244.

3. Демин, А. М. Подход к выбору приоритетного инвестиционного решения в промышленно-гражданском строительстве с учетом риска / А. М. Демин, М. М. Луценко, Н. В. Чепаченко // *Проблемы и достижения в области строительного инжиниринга: Сборник материалов внутрифакультетской научной конференции, посвященной 210-летию Петербургского государственного университета путей сообщения Императора Александра I и 155-летию кафедры «Здания» (Санкт-Петербург, Россия, 17 апреля 2019 г.)* / под общ. ред. Т. А. Белаш и А. В. Кузнецова. — Санкт-Петербург: ПГУПС, 2019. — С. 27–31.

4. The Influence of Wet Ground Fly Ash on the Performance of Foamed Concrete / G. Li, H. Tan, X. He, [et al.] // *Construction and Building Materials*. 2021. Vol. 304. Art. No. 124676. 10 p. DOI: 10.1016/j.conbuildmat.2021.124676.

5. Waste-to-Energy, Municipal Solid Waste Treatment, and Best Available Technology: Comprehensive Evaluation by an Interval-Valued Fuzzy Multi-Criteria Decision Making Method / Z. Wang, J. Ren, M. E. Goodsite, G. Xu // *Journal of Cleaner Production*. 2018. Vol. 172. Pp. 887–899. DOI: 10.1016/j.jclepro.2017.10.184.

6. Hwang, Ch.-L. Group Decision Making Under Multiple Criteria: Methods And Applications / Ch.-L. Hwang, M.-J. Lin. — Berlin: Springer-Verlag, 1987. — 411 p. — (Lecture Notes in Economics and Mathematical Systems, Vol. 281).

7. Козлова, Е. А. Линейные обобщенные критерии в задаче многокритериальной оптимизации / Е. А. Козлова, Л. И. Руденко // *Математика, Информатика, Компьютерные науки, Моделирование, Образование = Mathematics, Informatics, Computer Science, Modeling, Education: Сборник научных трудов научно-практической конференции МИКМО-2017 и Таврической научной конференции студентов и молодых специалистов по математике и информатике* / Крымский федеральный университет имени В. И. Вернадского, Таврическая академия; под ред. В. А. Лукьяненко. — Симферополь: ИП Корниенко А. А., 2017. — С. 138–143.

8. Rusanova, E. V. The Control Waste of Communal Services / E. V. Rusanova, M. S. Abu-Khasan, A. S. Sakharova // *Proceedings of the International Science and Technology Conference «Earth Science» (Vladivostok, Russia, 04–06 March 2019)*. IOP Conference Series: Earth and Environmental

Science. 2019. Vol. 272, Is. 2. Art. No. 022109. 6 p. Published online at 21 June 2019. DOI: 10.1088/1755-1315/272/2/022109.

9. Rusanova, E. V. The Complex Evaluation of Geo Eco-Protective Technologies Taking into Account the Influence of Negative Temperatures / E. V. Rusanova, M. S. Abu-Khasan, V. V. Egorov // *Proceedings of the International Science and Technology Conference «FarEastCon-2019» (Vladivostok, Russia, 01–04 October 2019)*. IOP Conference Series: Materials Science and Engineering. 2020. Vol. 753, Is. 1. Art. No. 022042. 8 p. Published online at 05 March 2020. DOI: 10.1088/1757-899X/753/2/022042.

10. Золопенбетон с использованием золы осадка сточных вод / А. М. Сычева, А. В. Хитров, М. В. Шершнева, Е. В. Русанова // *Цемент и его применение*. 2006. № 3. С. 64.

11. Sychova, A. M. A Method of Obtaining Geonoseprotective Foam Concrete for Use on Railway Transport / A. M. Sychova, M. M. Sychov, E. V. Rusanova // *Proceedings of the International Scientific Conference Transportation Geotechnics and Geocology (TGG 2017), (Saint Petersburg, Russia, 17–19 May 2017)*. *Procedia Engineering*. 2017. Vol. 189. Pp. 681–687. DOI: 10.1016/j.proeng.2017.05.108.

12. Луценко, М. М. Принятие инвестиционных решений в строительстве при неполной информации о функционировании объекта / М. М. Луценко, А. М. Дёмин // *Управление рисками в экономике: проблемы и решения* / отв. ред. С. Г. Опарин. — Санкт-Петербург, Изд-во Политехнического ун-та Петра Великого, 2015. — С. 237–259.

13. Harju, M. Spatial Multi-Attribute Decision Analysis: Axiomatic Foundations and Incomplete Preference Information / M. Harju, J. Liesiö, K. Virtanen // *European Journal of Operational Research*. 2019. Vol. 275, Is. 1. Pp. 167–181. DOI: 10.1016/j.ejor.2018.11.013.

14. Нигматов, А. Н. Математическое моделирование в экологии / А. Н. Нигматов, Г. Н. Назарова // *Евразийский союз ученых*. 2018. № 3-2 (48). С. 48–50.

15. Подиновский, В. В. Парето-оптимальные решения многокритериальных задач / В. В. Подиновский, В. Д. Ногин. — Москва: Наука. Главная редакция физико-математической литературы, 1982. — 256 с. — (Экономико-математическая библиотека).

16. Птускин, А. С. Инструментальные средства моделирования выбора наилучших доступных технологий / А. С. Птускин, Е. В. Левнер // *Научный результат. Серия «Экономические исследования»*. 2016. Т. 2. № 2. С. 58–63. DOI: 10.18413/2409-1634-2016-2-2-58-63.

17. Птускин, А. С. Нечеткая модель выбора альтернативных операций наилучшей доступной технологии на уровне установки / А. С. Птускин, Ю. М. Жукова // *Экономика и математические методы*. 2020. Т. 56. № 4. С. 78–87. DOI: 10.31857/S042473880012417-7.

18. Rusanova, E. V. A Method for Evaluating Geo-Environmental Technologies Based on a Weighted Convolution of Partial Performance Criteria in the MATLAB Environment / E. V. Rusanova, E. V. Runev // *Proceedings of Models and Methods for Researching Information Systems in Transport 2020 (MMRIST 2020), (St. Petersburg, Russia, 11–12 December 2020)*. *CEUR Workshop Proceedings*. 2021. Vol. 2803. Pp. 92–99. DOI: 10.24412/1613-0073-2803-92-99.

19. Рыков, А. С. Модели и методы системного анализа: принятие решений и оптимизация: Учебное пособие для

студентов вузов, обучающихся по направлениям подготовки дипломированных специалистов «Металлургия» и «Физическое моделирование». — Москва: МИСиС: Издательский дом «Руда и металлы», 2005. — 351 с. — (Металлургия и материаловедение XXI века).

20. Vincke, P. *Multicriteria Decision-Aid*. — Chichester: John Wiley & Sons, 1992. — 174 p.

21. A Mathematical Programming-Based Method for Heterogeneous Multicriteria Group Decision Analysis with Aspirations and Incomplete Preference Information / W. Zhang, Y. Ju, X. Liu, M. Giannakis // *Computers and Industrial Engineering*. 2017. Vol. 113. Pp. 541–557.

DOI: 10.1016/j.cie.2017.09.030.

22. Yu, P.-L. *Multiple-Criteria Decision Making: Concepts, Techniques, and Extensions*. — New York: Plenum Press, 1985. — 402 p. — (Mathematical Concepts and Methods in Science and Engineering, Vol. 30).

23. Fan, Z.-P. A Method for Multiple Attribute Decision-Making with the Fuzzy Preference Relation on Alternatives / Z.-P. Fan, G.-F. Hu, S.-H. Xiao // *Selected papers from the XXVII International Conference on Computers and Industrial Engineering (27<sup>th</sup> ICC&IE)*, (Beijing, China, 11–13 October 2000), Part 1. *Computers and Industrial Engineering*. 2004. Vol. 46, Is. 2. Pp. 321–327. DOI: 10.1016/j.cie.2003.12.011.

# Using Decision-Making Models to Optimize the Choice of Technologies for the Production of Geo-Eco-Protective Materials

PhD E. V. Rusanova, E. V. Runev

Emperor Alexander I St. Petersburg State Transport University  
Saint Petersburg, Russia  
rusanovaev@mail.ru, jr\_2010@mail.ru

**Abstract.** The problem of material selection, when used in geo-ecological technologies, is relevant. This paper presents a mathematical model of choosing the best available material, based on the criterion of optimality application, defined on finite sets — factors (technical characteristics of objects) and alternatives (materials of different chemical composition). Based on the model a calculation method was developed, it allows standardizing materials used in geo-ecological technologies according to technical requirements from the client. This method makes it possible to make a decision based on monitoring data of the use of technologies over a long period. These decisions generally have long-term impacts on environmental safety and economic efficiency.

**Keywords:** optimal strategies, set of states, set of alternatives, Hurwitz criterion, a priori probability distribution, maximum expected utility criterion, Hodges — Lehman criterion, best available technology, ash concrete.

## REFERENCES

1. Torkayesh A. E., Malmir B., Asadabadi M. R. Sustainable Waste Disposal Technology Selection: The Stratified Best-Worst Multi-Criteria Decision-Making Method, *Waste Management*, 2021, Vol. 122, Pp. 100–112. DOI: 10.1016/j.wasman.2020.12.040.
2. Chepachenko N. V., Lutsenko M. M., Demin A. M. Substantiation of Selecting the Priority Innovation Solution in Construction Using the Game Theory Apparatus [Obosnovanie vybora prioritetnogo innovatsionnogo resheniya v stroitel'stve s primeneniem apparata teorii igr], *Bulletin of Civil Engineers [Vestnik grazhdanskikh inzhenerov]*, 2020, No. 3 (80), Pp. 238–244. DOI: 10.23968/1999-5571-2020-17-3-238-244.
3. Demin A. M., Lutsenko M. M., Chepachenko N. V. Approach to the Selection of the Priority Investment Solution in Industrial and Civil Construction Taking into Account the Risk [Podkhod k vyboru prioritetnogo investitsionnogo resheniya v promyshlenno-grazhdanskom stroitel'stve s uchetom riska]. In: *Belash T. A., Kuznetsov A. V. (eds) Problems and Achievements in the Field of Construction Engineering: Proceedings of the Intrafaculty Scientific Conference Dedicated to the 210th Anniversary of Emperor Alexander I St. Petersburg State Transport University of Railway Transport of Emperor Alexander I [Problemy i dostizheniya v oblasti stroitel'nogo inzhiniringa: Sbornik materialov vnutrifakul'tetskoy nauchnoy konferentsii, posvyashchennoy 210-letiyu Peterburgskogo gosudarstvennogo universiteta putey soobshcheniya Imperatora Aleksandra I]*, Saint Petersburg, Russia, April 17, 2019. Saint Petersburg, PSTU, 2019, Pp. 27–31.
4. Li G., Tan H., He X., et al. The Influence of Wet Ground Fly Ash on the Performance of Foamed Concrete, *Construction and Building Materials*, 2021, Vol. 304, Art. No. 124676, 10 p. DOI: 10.1016/j.conbuildmat.2021.124676.
5. Wang Z., Ren J., Goodsite M. E., Xu G. Waste-to-Energy, Municipal Solid Waste Treatment, and Best Available Technology: Comprehensive Evaluation by an Interval-Valued Fuzzy Multi-Criteria Decision Making Method, *Journal of Cleaner Production*, 2018, Vol. 172, Pp. 887–899. DOI: 10.1016/j.jclepro.2017.10.184.
6. Hwang Ch.-L., Lin M.-J. Group Decision Making Under Multiple Criteria: Methods And Applications. Berlin, Springer-Verlag, 1987, 411 p.
7. Kozlova E. A., Rudenko L. I. Linear Generalized Criteria in the Problem of Multicriteria Optimization [Lineynye obobshchennyye kriterii v zadache mnogokriterial'noy optimizatsii], *Mathematics, Informatics, Computer Science, Modeling, Education: Proceedings of the Scientific and Practical Conference (MICMO-2017) and the Taurida Scientific Conference of Students and Young Specialists in Mathematics and Informatics [Matematika, Informatika, Komp'yuternyye nauki, Modelirovanie, Obrazovanie: Sbornik nauchnykh trudov nauchno-prakticheskoy konferentsii MIKMO-2017 i Tavricheskoy nauchnoy konferentsii studentov i molodykh spetsialistov po matematike i informatike]*, Simferopol, Russia, April 10–14, 2017. Simferopol, A. A. Kornienko Publishing House, 2017, Pp. 138–143.
8. Rusanova E. V., Abu-Khasan M. S., Sakharova A. S. The Control Waste of Communal Services, *Proceedings of the International Science and Technology Conference «Earth Science»*, Vladivostok, Russia, March 04–06, 2019. *IOP Conference Series: Earth and Environmental Science*, 2019, Vol. 272, Is. 2., Art. No. 022109, 6 p. Published online at June 21, 2019. DOI: 10.1088/1755-1315/272/2/022109.
9. Rusanova E. V., Abu-Khasan M. S., Egorov V. V. The Complex Evaluation of Geo Eco-Protective Technologies Taking into Account the Influence of Negative Temperatures, *Proceedings of the International Science and Technology Conference «FarEastCon-2019»*, Vladivostok, Russia, October 01–04, 2019. *IOP Conference Series: Materials Science and Engineering*, 2020, Vol. 753, Is. 1, Art. No. 022042, 8 p. Published online at March 05, 2020. DOI: 10.1088/1757-899X/753/2/022042.
10. Sycheva A. M., Khitrov A. V., Shershneva M. V., Rusanova E. V. Production of Ash-Foam Concrete Using the

Sewage Sludge Ash [Zolopenobeton s ispol'zovaniem zoly osadka stochnykh vod], *Cement and Its Applications [Tsement i ego primeneniye]*, 2006, No. 3, P. 64.

11. Sychova A. M., Sychov M. M., Rusanova E. V. A Method of Obtaining Geonoseprotective Foam Concrete for Use on Railway Transport, *Proceedings of the International Scientific Conference Transportation Geotechnics and Geology (TGG 2017), Saint Petersburg, Russia, May 17–19, 2017. Procedia Engineering*, 2017, Vol. 189, Pp. 681–687. DOI: 10.1016/j.proeng.2017.05.108.

12. Lutsenko M. M., Demin A. M. Investment Decisions — Making in Construction with Incomplete Information About the Facility's Operation [Prinyatie investitsionnykh resheniy v stroitel'stve pri nepolnoy informatsii o funktsionirovaniy ob'ekta]. In: *Oparin S. G. (ed.) Risk management in the economy: Problems and solutions [Upravlenie riskami v ekonomike: problemy i resheniya]*. St. Petersburg, Peter the Great St. Petersburg Polytechnic University, 2015, Pp. 237–259.

13. Harju M., Liesiö J., Virtanen K. Spatial Multi-Attribute Decision Analysis: Axiomatic Foundations and Incomplete Preference Information, *European Journal of Operational Research*, 2019, Vol. 275, Is. 1, Pp. 167–181. DOI: 10.1016/j.ejor.2018.11.013.

14. Nigmatov A. N., Nazarova G. N. Mathematical Modeling in Ecology [Matematicheskoe modelirovanie v ekologii], *Eurasian Union of Scientists [Evraziyskiy soyuz uchenykh]*, 2018, No. 3-2 (48), Pp. 48–50.

15. Podinovskiy V. V., Nogin V. D. Pareto-optimal solutions of multi-criteria problems [Pareto-optimal'nye resheniya mnogokriterial'nykh zadach]. Moscow, Nauka Publishers, 1982, 256 p.

16. Ptuskin A. S., Levner E. V. Instrumental Tools for Modeling the Problem of Selecting the Best Available Technologies [Instrumental'nye sredstva modelirovaniya vybora nailuchshikh dostupnykh tekhnologiy], *Scientific Result. Economic Research Series [Nauchnyy rezul'tat. Seriya «Ekonomichekские issledovaniya»]*, 2016. Vol. 2., No. 2, Pp. 58–63. DOI: 10.18413/2409-1634-2016-2-2-58-63.

17. Ptuskin A. S., Zhukova Yu. M. A Fuzzy Model of the Selection of Alternative Operations of the Best Available Techniques at the Installation Level [Nechetkaya model' vybora al'ternativnykh operatsiy nailuchshey dostupnoy tekhnologii na urovne ustanovki], *Economics and Mathematical Methods [Ekonomika i matematicheskie metody]*, 2020, Vol. 56, No. 4, Pp. 78–87.

DOI: 10.31857/S042473880012417-7.

18. Rusanova E. V., Runev E. V. A Method for Evaluating Geo-Environmental Technologies Based on a Weighted Convolution of Partial Performance Criteria in the MATLAB Environment, *Proceedings of Models and Methods for Researching Information Systems in Transport 2020 (MMRIST 2020), St. Petersburg, Russia, December 11–12, 2020. CEUR Workshop Proceedings*, 2021, Vol. 2803, Pp. 92–99. DOI: 10.24412/1613-0073-2803-92-99.

19. Rykov A. S. Models and methods of systems analysis: Decision-making and optimization: Study guide for university students [Modeli i metody sistemnogo analiza: prinyatie resheniy i optimizatsiya: Uchebnoe posobie dlya studentov vuzov]. Moscow, National University of Science and Technology «MISiS», Ore and Metals Publishing House, 2005, 351 p.

20. Vincke P. Multicriteria Decision-Aid. Chichester, John Wiley & Sons, 1992, 174 p.

21. Zhang W., Ju Y., Liu X., Giannakis M. A Mathematical Programming-Based Method for Heterogeneous Multicriteria Group Decision Analysis with Aspirations and Incomplete Preference Information, *Computers and Industrial Engineering*, 2017, Vol. 113, Pp. 541–557. DOI: 10.1016/j.cie.2017.09.030.

22. Yu P.-L. Multiple-Criteria Decision Making: Concepts, Techniques, and Extensions. New York, Plenum Press, 1985, 402 p.

23. Fan Z.-P., Hu G.-F., Xiao S. H. A Method for Multiple Attribute Decision-Making with the Fuzzy Preference Relation on Alternatives, *Selected papers from the XXVII International Conference on Computers and Industrial Engineering (27<sup>th</sup> ICC&IE), Beijing, China, October 11–13, 2000, Part 1. Computers and Industrial Engineering*, 2004, Vol. 46, Is. 2, Pp. 321–327. DOI: 10.1016/j.cie.2003.12.011.