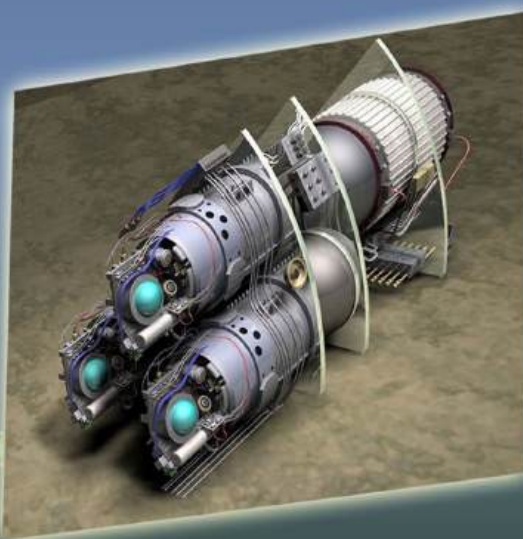
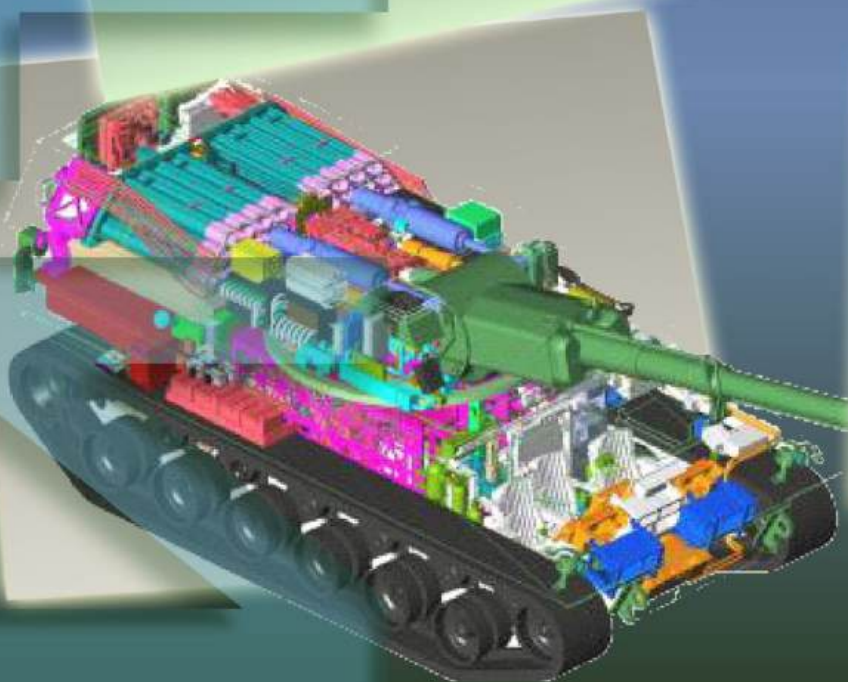
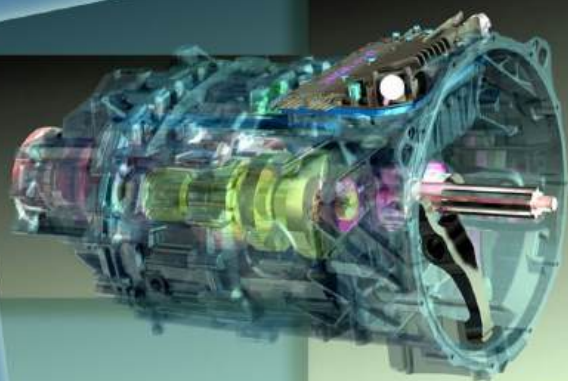
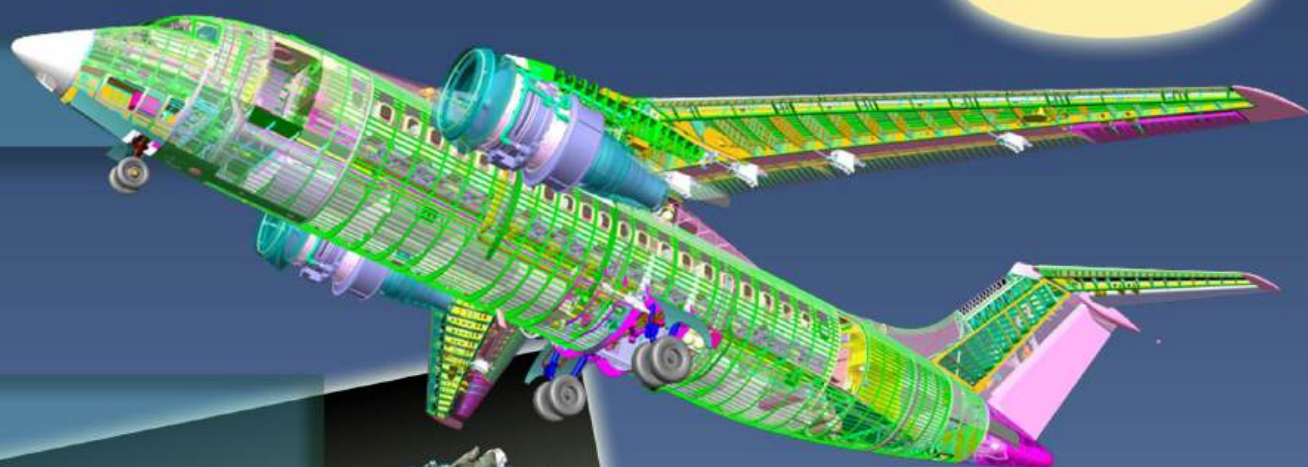


# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

в проектировании и производстве

№ 2' 2023





## Все новейшие достижения и современные разработки предприятий оборонного комплекса — в научно-технических журналах ФГУП «НТЦ оборонного комплекса «Компас»

ФГУП «Научно-технический центр оборонного комплекса «Компас» является издателем следующих научных журналов:



Межотраслевой научно-технический журнал  
**Оборонный комплекс — научно-техническому прогрессу России**  
(4 выпуска)  
Подписной индекс **79379**  
**Издается с 1984 года**



Межотраслевой научно-технический журнал  
**Конструкции из композиционных материалов**  
(4 выпуска)  
Подписной индекс **80089**  
**Издается с 1981 года**



Научно-технический журнал  
**Информационные технологии в проектировании и производстве**  
(4 выпуска)  
Подписной индекс **79378**  
**Издается с 1976 года**



Межотраслевой научно-практический журнал  
**Экология промышленного производства**  
(4 выпуска)  
Подписной индекс **80090**  
**Издается с 1993 года**



Научно-практический журнал  
**Вопросы защиты информации**  
(4 выпуска)  
Подписной индекс **79187**  
**Издается с 1974 года**

Все издания ФГУП «Научно-технический центр оборонного комплекса «Компас»:

✓ включены решением ВАК Министерства науки и высшего образования России в перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные результаты диссертаций на соискание ученой степени кандидата и доктора наук;

✓ метаданные выпусков включены в базу данных Российского индекса научного цитирования (РИНЦ).

Более подробную информацию об изданиях, подписке, дополнительных услугах можно получить по тел.: 8 (495) 491-43-17, 8 (495) 491-77-67, 8 (495) 491-77-20 (подписка);  
факс: 8 (495) 491-44-80.  
E-mail: [izdanie@ntckompas.ru](mailto:izdanie@ntckompas.ru)



# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ПРОЕКТИРОВАНИИ И ПРОИЗВОДСТВЕ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ  
УНИТАРНОЕ ПРЕДПРИЯТИЕ  
«НАУЧНО-ТЕХНИЧЕСКИЙ ЦЕНТР  
ОБОРОННОГО КОМПЛЕКСА  
«КОМПАС»

## № 2 (190) 2023

НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Основан в 1976 г.

### Главный редактор

И. А. Шеремет, акад. РАН, проф., заместитель директора Российского фонда фундаментальных исследований по научной работе

### Заместитель главного редактора

В. Н. Гридин, д-р техн. наук, проф., научный руководитель ЦИТИП РАН

### Ответственный секретарь

Э. Ю. Есипова, канд. техн. наук, ФГУП «НТЦ оборонного комплекса «Компас»

### Редакционная коллегия:

В. В. Бордюже, д-р техн. наук, проф., Координационный совет по информационным технологиям предприятий ОПК РФ; А. И. Громов, канд. хим. наук, НИУ «Высшая школа экономики»; В. В. Кондратьев, чл.-кор. РАН, д-р техн. наук, проф., Нижегородский государственный технический университет им. Р. А. Алексеева; Ю. Н. Кофанов, д-р техн. наук, проф., МИЭМ НИУ ВШЭ; С. А. Матвеев, канд. техн. наук, ФГУП «ЦНИИмаш»; Е. И. Митрушкин, д-р техн. наук, проф., АО «НИИИА»; С. В. Назаров, д-р техн. наук, проф., ЗАО «Московский научно-исследовательский телевизионный институт»; М. А. Пирогова, канд. техн. наук, НИУ МЭИ; С. В. Попов, д-р техн. наук, Холдинг «Швабе»; А. Ф. Ронжин, д-р физ.-мат. наук, проф., Вычислительный центр им. А. А. Дородницына; П. О. Скобелев, д-р техн. наук, НАО «Группа компаний «Генезис знаний», СамГТУ; В. В. Соломин, канд. воен. наук, ФГУП «НТЦ оборонного комплекса «Компас»; А. П. Солнцев, д-р воен. наук, проф., ФГУП «НТЦ оборонного комплекса «Компас»; Е. В. Судов, д-р техн. наук, проф., НИЦ CALS-технологий «Прикладная логистика»; А. В. Цырков, д-р техн. наук, проф., ГИИП им. М. В. Хруничева; В. А. Шахнов, чл.-кор. РАН, д-р техн. наук, проф., МГТУ им. Н. Э. Баумана; Е. И. Шульгин, д-р техн. наук, НИЦ при президиуме АИИ им. А. М. Прохорова

© Федеральное государственное унитарное предприятие «Научно-технический центр оборонного комплекса «Компас» (ФГУП «НТЦ оборонного комплекса «Компас»), 2023

## СОДЕРЖАНИЕ

### Инновационные технологии в решении комплексных задач управления и проектирования, моделирования и производства

Жабоев Т. К., Саченко А. С., Сумароков С. В., Попов А. В., Кулаков Е. Н. Подбор настроек регуляторов модели энергоблока на базе расчетного комплекса для решения оптимизационных задач..... 3

Прудников В. А. О перспективности применения Data Mining в проектировании..... 13

### Информационные технологии в решении задач управления

Жалялов Р. Р. Использование свёрточных нейронных сетей для решения задач распознавания образов в промышленности ..... 21

Белов П. Д., Махмадзиёев А. О., Павленко А. А., Пителинский К. В. Применение программных средств для автоматизации тестирования безопасности объектов контейнеризации..... 32

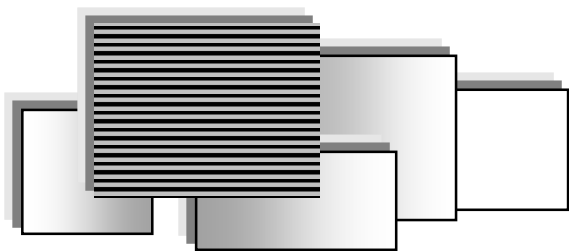
### Прикладные вопросы и задачи применения информационных технологий

Жданова О. В., Столяров О. Н. Опыт применения машиночитаемых требований в проектном направлении «Прорыв» ..... 42

# **INFORMATION TECHNOLOGY OF CAD/CAM/CAE**

## **THE SCIENTIFIC JOURNAL**

Information on recent advances in CAD/CAM/CAE, industry & business & complex information technology, including methods and tools of computer graphing, geometrical and mathematical modeling, database control methods on estimation and provision of control and reliability.



## ИННОВАЦИОННЫЕ ТЕХНОЛОГИИ В РЕШЕНИИ КОМПЛЕКСНЫХ ЗАДАЧ УПРАВЛЕНИЯ И ПРОЕКТИРОВАНИЯ, МОДЕЛИРОВАНИЯ И ПРОИЗВОДСТВА

УДК 519.688, 004.942, 519.876.5  
DOI: 10.52190/2073-2597\_2023\_2\_3  
EDN: IUHSF

### ПОДБОР НАСТРОЕК РЕГУЛЯТОРОВ МОДЕЛИ ЭНЕРГБЛОКА НА БАЗЕ РАСЧЕТНОГО КОМПЛЕКСА ДЛЯ РЕШЕНИЯ ОПТИМИЗАЦИОННЫХ ЗАДАЧ

Т. К. Жабоев; А. С. Саченко; С. В. Сумароков

Институт проблем безопасного развития атомной энергетики РАН, Москва, Россия

А. В. Попов; Е. Н. Кулаков

Научно-производственное объединение по исследованию и проектированию  
энергетического оборудования им И. И. Ползунова (НПО ЦКТИ), Санкт-Петербург, Россия

*Приведены результаты апробации расчётного комплекса для решения задач оптимизации режимов эксплуатации реакторной установки. Апробация проведена на примере решения задачи оптимизации настроек регулятора оборудования на базе динамической модели второго контура энергоблока БРЕСТ-ОД-300. Применена технология подбора настроек регулятора уровня смешивающего подогревателя питательной воды для различных режимов работы энергоблока.*

**Ключевые слова:** расчётный комплекс, многокритериальная оптимизация, АЭС, динамическая модель, переходной процесс, регулирование, *pSeven*, *SimInTech*.

Автоматизация технологических процессов теплоэнергетических установок играет определяющую роль в обеспечении безопасной работы станций. Неправильно выбранные настройки систем регулирования могут приводить к излишней колебательности параметров оборудования и большим отклонениям их от заданной величины, что влияет как на ресурс отдельных элементов, так и на возникновение нарушений нормальных условий эксплуатации.

Настройка системы регулирования представляет собой весьма непростую задачу, которая обусловлена как выбором метода определения оптимальных параметров настроек, так и количеством взаимосвязей различных систем регулирования и неопределённостью их работы.

Существует значительное количество методов настройки систем регулирования, однако многие из них чрезмерно трудоемки или не обладают достаточной точностью. Как правило, использование этих методов позволяет обеспечить

устойчивость управления технологическими процессами, но не гарантирует качественную работу регулирующего органа.

Для повышения качества и увеличения эффективности, устойчивости и надёжности работы системы автоматического регулирования, должны быть найдены оптимальные настройки ее регуляторов.

В целях анализа режимов работы установок создают математические модели технологических процессов и самого оборудования. Это делает возможным проведение анализа процессов, протекающих в оборудовании энергоблока при различных режимах работы еще на стадии проектирования, что позволяет снизить затраты на натурные испытания и минимизировать последствия ошибочных решений, отработать технологию пусконаладочных работ, предотвратить возможные внештатные ситуации и выполнить расчёт параметров в различных режимах эксплуатации, не прибегая к эмпирическим исследованиям.

В настоящий момент одной из новых и перспективных ядерных энергетических установок, строящихся в России, является БРЕСТ-ОД-300 [1, 2]. Особенностью установки является использование в качестве теплоносителя первого контура жидкого свинца, что накладывает ряд эксплуатационных ограничений, которые ранее не предъявлялись к энергоблокам ТЭС и АЭС [3].

Для комплексного анализа работы энергоустановки разработана и применяется динамическая модель турбоустановки и систем II контура энергоблока с реакторной установкой (РУ) БРЕСТ-ОД-300 [4]. Модель реализована в программном комплексе *SimInTech* [5] с интегрированным теплогидравлическим аттестованным кодом *TRP*, используемым для моделирования динамики переходных процессов в системе сложных теплогидравлических контуров произвольной конфигурации и показывающим достаточно высокую точность [6, 7]. Модель позволяет исследовать различные переходные процессы энергоблока как при нормальных условиях эксплуатации, так и при внештатных ситуациях [8].

Поиск оптимальных настроек регуляторов можно осуществлять посредством автоматизированных многовариантных исследований, сопровождаемых моделированием переходных режимов системы с различными настройками

регуляторов. Для данной цели может быть использован расчётный комплекс, представляющий собой специализированную программную платформу, адаптированную к соответствующей предметной области и расчётную систему, отвечающую за моделирование соответствующих процессов. При этом, программная платформа реализует алгоритмы оптимизации, а также сервисы управления запуском и выполнением расчётной системы, обеспечивая автоматизированный поиск оптимальных настроек регулятора для заданного исследователем режима и предъявляемых к нему требований.

Для проведения автоматизированных многовариантных исследований разработан расчётный комплекс для решения задач оптимизации режимов эксплуатации реакторных установок (далее РКО РЭ), который позволяет проводить параметрические и оптимизационные исследования режимов работы РУ на этапах НИОКР.

#### Расчётный комплекс для решения задач оптимизации режимов эксплуатации РУ

Общая структура расчётного комплекса для решения задач оптимизации режимов эксплуатации реакторных установок представлена на рис. 1.



Рис. 1. Общая структура РКО РЭ

Реализация РКО РЭ построена на базе программной платформы *pSeven*, предоставляющей необходимые алгоритмы и методы анализа данных, а также выполняющей роль интегратора для связывания расчётных систем в единый комплекс [9]. Выбор *pSeven* в качестве платформы обусловлен наличием в данном программном обеспечении широкого функционала для решения задач инженерной оптимизации и анализа данных, наличием средств по интеграции с внешними расчётными программами, удобным конструктором расчётных схем, а также наличием функционала для построения суррогатных моделей и их исследования.

Для подключения программного комплекса *SimInTech* к платформе *pSeven* был разработан специализированный программный интерфейс, предоставляющий возможности обмена данными между платформой *pSeven* и *SimInTech*, а также запуска и управления расчётами в *SimInTech*.

Для проведения исследований по оптимизации настроек регуляторов была подготовлена расчётная схема в *pSeven*, состоящая из двух уровней. На верхнем уровне (рис. 2) расположены блок параметрической оптимизации *DSE* и блок *Composite*.

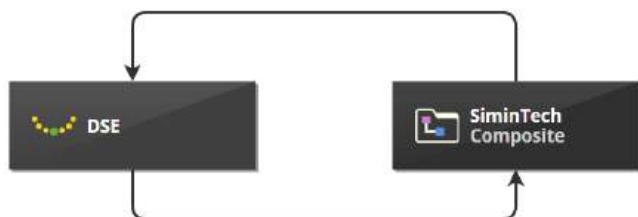


Рис. 2. Верхний уровень расчётной схемы

Блоки — это функциональные гибко настраиваемые элементы, в которых выполняется обработка входных и формирование выходных данных. При помощи связей осуществляется взаимодействие между блоками, а также задается очередность их запуска. Для проведения многовариантных оптимизационных расчётов расчётная схема должна содержать оптимизационный цикл. Управление циклом расчётов осуществляется блоком параметрической оптимизации *DSE*, являющимся ключевым элементом схемы, который обеспечивает "умный перебор" варьируемых параметров, оценивает отклики модели на предмет достижения целевых критериев и удовлетворения ограничений постановки задачи.

В рамках циклического расчёта блок *DSE* позволяет:

- накапливать и анализировать результаты исследования каждого цикла и соответствующие им значения входных и выходных параметров;
- на основании анализа накопленных данных и заданных критериев и ограничений генерировать входные параметры для каждого цикла;
- определять и отслеживать критерии остановки для своевременного завершения расчёта.

Важно отметить, что для блока *DSE* исследуемый объект является "черным ящиком". В ходе исследования блок выполняет вычисления, основываясь на входных и выходных данных, а также на заданных критериях и ограничениях задачи.

Блок *DSE* отвечает за выполнение итерационного циклического расчёта с обратной связью, обеспечивая генерацию значений варьируемых параметров, а также оценку целевых функций на предмет критериев и ограничений постановки задачи.

Блок *Composite* содержит вложенную схему, предоставляя возможность параллельного запуска нескольких ее экземпляров. Другой важной особенностью блока *Composite* является возможность кэширования значений входных и выходных параметров завершённых итераций расчёта, что позволяет возобновлять расчеты с последней успешной итерации в случае возникновения технического сбоя.

Нижний уровень расчётной схемы, реализованный посредством вложенной схемы блока *Composite*, содержит интеграционный блок *SimInTech* и вспомогательные блоки, использование которых обусловлено спецификой выполнения расчётов (рис. 3).

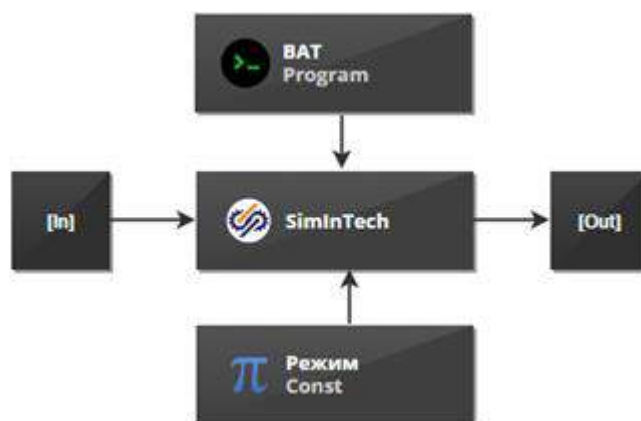


Рис. 3. Нижний уровень расчётной схемы

В рамках циклического расчёта связь портов блоков обеспечивает строгую очередность запуска и выполнения блоков:

- 1) Шаг 1 — генерация значений варьируемых параметров (значений коэффициентов регуляторов) в блоке *DSE*;
- 2) Шаг 2 — подготовка данных для расчёта, в частности, создание в изолированной папке копии исходной расчётной модели блоком *Program (BAT)*;
- 3) Шаг 3 — инициализация условий моделируемого переходного процесса в блоке *Const*;
- 4) Шаг 4 — запуск и отслеживание выполнения блоком *SimInTech* расчёта переходного режима РУ с измененными значениями коэффициентов регуляторов, полученных из блока *DSE*. По завершении расчёта блок *SimInTech* возвращает значения целевых функций в блок *DSE*;

5) Шаг 5 — анализ рассчитанных значений целевых функций блоком *DSE*, генерация новых значений варьируемых параметров либо завершение циклического расчёта.

### Регулятор модели энергоблока

Во всех режимах работы энергоблока переходные процессы различных систем регулирования должны отвечать определенным требованиям, таким, как степень устойчивости и колебательности переходного процесса, длительность процесса регулирования, динамическая погрешность регулирования и прочие, которые в сумме отражают качество регулирования системы. Обеспечение требуемого качества регулирования

выполняется путем выбора оптимальных настроек системы автоматического регулирования (САР).

Регулятор, исследуемый в данной работе, выполнен по схеме регулирующего блока Р-27 с отрицательной исчезающей обратной связью по изменению положения органа регулирования и дополнительной обратной связью по материальному дебалансу, фактически являющейся дифференциальной составляющей (рис. 4).

В рамках апробации подхода рассмотрена САР регулирующего клапана (РК) уровня смешивающего подогревателя питательной воды (СППВ). На рис. 5 представлена принципиальная схема системы питательной воды энергоблока БРЕСТ-ОД-300.

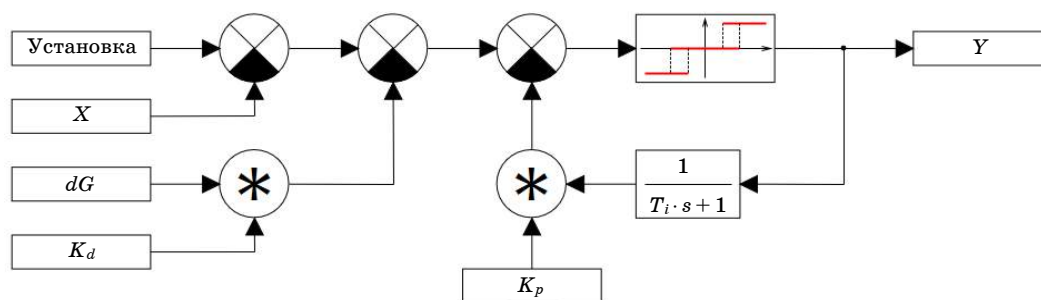


Рис. 4. Схема регулятора типа Р-27:

*X* — входной сигнал от датчика уровня; *dG* — сигнал материального дебаланса, вычисляемый по датчикам расхода; *Y* — выходной сигнал на открытие/закрытие регулирующего органа;  $K_p$  — коэффициент усиления обратной связи;  $T_i$  — постоянная времени фильтра обратной связи;  $K_d$  — коэффициент усиления сигнала *dG*

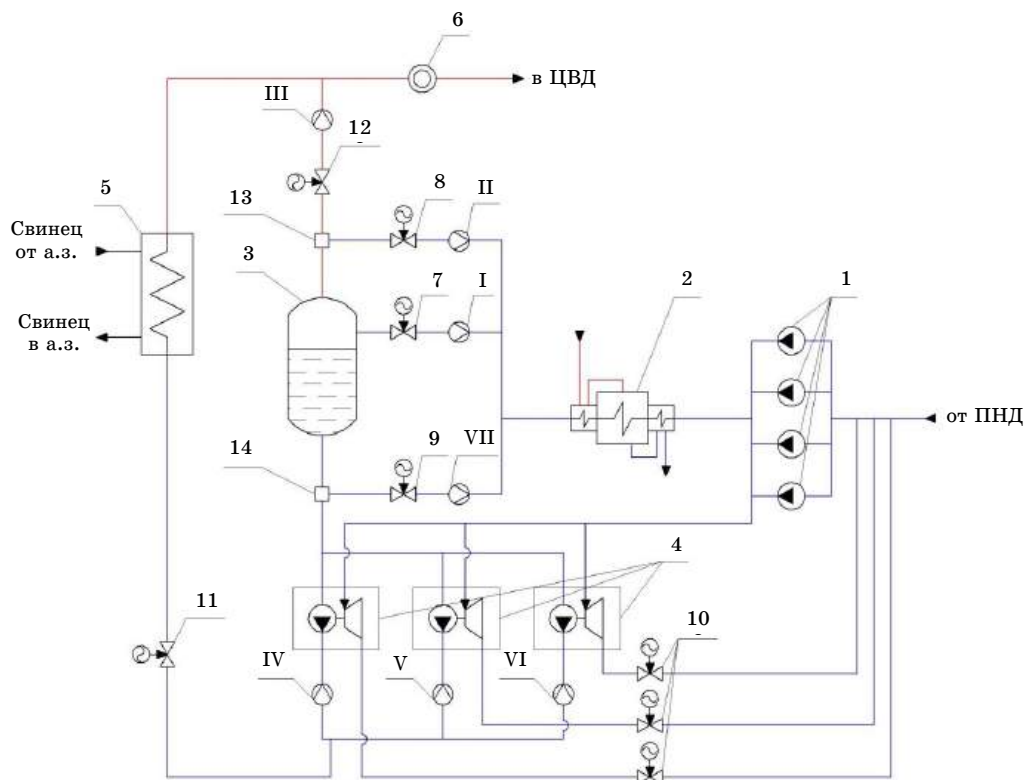


Рис. 5. Схема питательной воды энергоблока БРЕСТ-ОД-300:

1 — питательный электронасос; 2 — подогреватель высокого давления; 3 — СППВ; 4 — ПГТН; 5 — ПГ; 6 — стопорно-регулирующий клапан паровой турбины; 7 — РК уровня СППВ; 8 — РК греющего пара СППВ; 9 — РК захлаживания перед ПГТН; 10 — РК управления ПГТН; 11 — регулирующий питательный клапан; 12 — РК давления в СППВ; 13 — охладитель пара СППВ; 14 — смеситель перед ПГТН; I—VII — датчики расхода



Основной конденсат от подогревателей низкого давления (ПНД) поступает к питательным насосам 1, которые прокачивают воду через подогреватели высокого давления 2 в смешивающий подогреватель питательной воды 3. Далее с помощью питательных гидротурбинных насосов (ПГТН) 4 питательная вода направляется в прямоточные парогенераторы 5, охлаждая свинец из активной зоны (а. з.), и далее поступает в цилиндр высокого давления (ЦВД) турбины.

Одной из наиболее важных систем второго контура энергоблока БРЕСТ-ОД-300 является система регулирования уровня смешивающего подогревателя питательной воды высокого давления, который предназначен для подачи в парогенератор (ПГ) воды с температурой не менее 340 °С во всех режимах работы реакторной установки. Принцип его работы в составе системы питательной воды изложен в [9]. Использование ПГ позволяет не только осуществлять нагрев воды до требуемой высокой температуры, но и обеспечивать запас питательной воды перед парогенераторами. При этом стоит отметить, что РК уровня СППВ оказывает непосредственное влияние на гидравлический тракт питательной воды и, соответственно, на другие его системы.

### Постановка задачи

Исходные настройки САР расчётной модели были определены эмпирическим методом посредством анализа кривой разгона объекта регулирования. Поскольку при разработке модели в первую очередь стоит задача достижения устойчивости переходных процессов, такой подход определения настроек использован для всех регуляторов динамической модели энергоблока, которых насчитывается порядка восьмидесяти.

В рамках проводимого исследования подбор оптимальных настроек регулятора уровня СППВ выполняли в переходном режиме, предусматривающем ступенчатое понижение уставки регулятора (режим "ступенька") величиной, равной 10 % от номинального уровня. Направление изменения уставки выбрано из условия недопущения полного открытия регулирующего клапана, что исказило бы результаты расчёта.

Рассматриваемая динамическая модель энергоблока с РУ БРЕСТ-ОД-300 позволяет моделировать различные переходные режимы, и поскольку искомые настройки регулятора должны обеспечивать приемлемое качество регулирования во всех режимах работы энергоблока, то получаемые решения должны быть дополнительно исследованы. В нашем примере выбраны два режима, связанные с глобальными изменениями по энергоблоку, — снижение тепловой мощности реакторной установки со скоростью 1 %/с до 50 %  $Q_{\text{ном}}$  (режим "разгрузка") и отключение питательного электронасоса (ПЭН-1) с последующей разгрузкой РУ с той же скоростью

снижения мощности до 50 %  $Q_{\text{ном}}$  (режим "отключение ПЭН-1"), предусмотренные алгоритмами управления.

В процессе многовариантных исследований коэффициенты настройки САР уровня СППВ варьировались в следующих диапазонах, определенных исходя из начальных настроек и опыта настройки аналогичных регуляторов:

- $K_p$  в диапазоне от 0 до 2,5;
- $T_i$  в диапазоне от 0 до 30;
- $K_d$  в диапазоне от 0 до 1,25.

Для комплексной оценки качества регулирования уровня СППВ приняты следующие критерии:

• Накопленное рассогласование (НР) — интегральный критерий качества, отражающий затраты на ликвидацию возмущения, вычисляемый по формуле:

$$NR = \int_0^t \varepsilon(t) dt$$

$$\varepsilon(t) = \begin{cases} Y_{\phi} - Y_{\text{уст}} & |Y_{\phi} > (Y_{\text{уст}} + \Delta Y / 2) \\ Y_{\text{уст}} - Y_{\phi} & |Y_{\phi} < (Y_{\text{уст}} - \Delta Y / 2) \end{cases} \quad (1)$$

где  $Y_{\phi}$  — фактическое значение регулируемой величины;

$Y_{\text{уст}}$  — уставка регулятора;

$\Delta Y$  — зона нечувствительности регулятора.

• Время хода клапана (ВХК) — критерий, оценивающий колебательность и ресурс органа регулирования:

$$ВХК = \int_0^t (X_{\phi} + X_{\text{м}}) dt, \quad (2)$$

где  $X_{\phi}$  — замыкание контакта реле на открытие регулирующего клапана (РК);

$X_{\text{м}}$  — замыкание контакта реле на закрытие РК.

• Максимальное рассогласование (МР) — критерий, равный максимальному отклонению регулируемой величины в переходном процессе от заданного значения

$$MR = \max |Y_{\phi} - Y_{\text{уст}}|. \quad (3)$$

• Время достижения уставки (ВДУ) — время, в течении которого регулируемая величина достигает значения уставки регулятора.

Целью оптимизационного расчёта являлся поиск коэффициентов  $K_p$ ,  $T_i$  и  $K_d$  регулятора уровня СППВ, применение которых обеспечивает минимальное НР, при условии, что ВХК не превышает величину, полученную при расчёте переходного процесса для исходных настроек

регулятора. МР и ВДУ используют в качестве дополнительных показателей для оценки получаемых решений.

### Результаты апробации

В результате проведенного итерационного расчёта с применением РКО РЭ было исследовано 200 конфигураций с различными настройками регулятора, из которых отобраны 10, удовлетворяющих ограничению по ВХК и обеспечивающих минимальное НР. Значения НР, соответствующие всем итерациям расчёта и оптимальным решениям, представлены на рис. 6.

По результатам дополнительных исследований в режимах "разгрузка" и "отключение ПЭН-1" для найденных в режиме "ступенька" настроек регулятора, в качестве оптимальной выбрана конфигурация с наименьшим НР. Оптимальные настройки регулятора представлены в табл. 1.

Сравнение основных параметров тракта питательной воды в переходных режимах при оптимальных настройках регулятора с аналогичными параметрами при настройках, найденных эмпирическим путем, показаны на рис. 7—9.

Из рис. 7 видно, что в режиме "ступенька", на котором производился поиск настроек регулятора, при оптимальных настройках максимальное снижение уровня составило 4996 мм против 4966 мм при исходных настройках. При этом значение уровня после провала вышло на значение уставки быстрее, на 128 с расчёта против 171 с.

В режиме "разгрузка" (рис. 8) максимальный заброс уровня при оптимальных настройках снизился с 5936 до 5835 мм. При этом качество регулирования параметров, таких как температура греющего пара СППВ и температура свинца на выходе за парогенератором (рис. 8, в, г), на которые оказывает непосредственное влияние клапан уровня СППВ, существенным образом не изменилось.

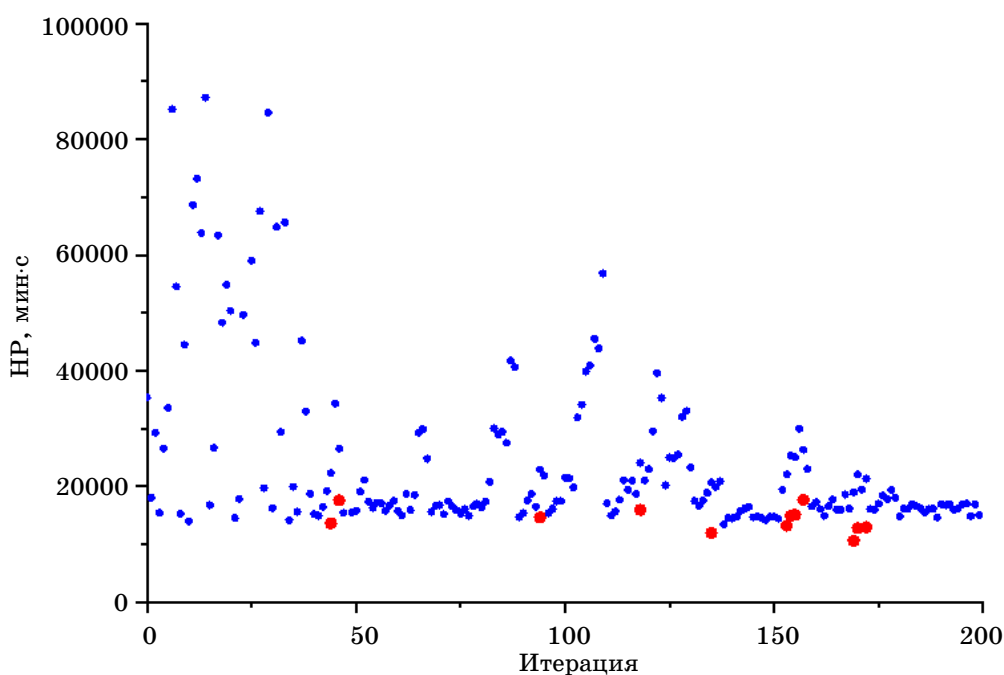


Рис. 6. Значения НР, полученные в результате оптимизационного расчёта:

• — все; ● — опт

Таблица 1

Настройки регулятора, найденные с помощью эмпирического метода и оптимизационного расчёта

Способ подбора настроек регулятора	Настройки регулятора		
	$K_p$	$T_i$	$K_d$
Эмпирический метод	0,75	12	0,5
Оптимизационный расчёт	0,716	8,7	0,675

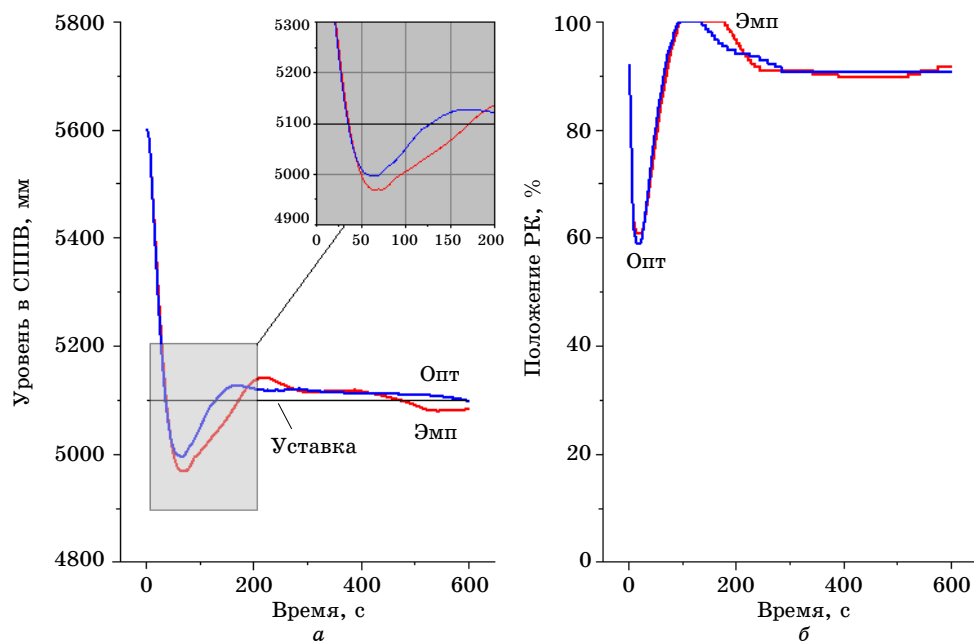


Рис. 7. Уровень и положение РК уровня в режиме "ступенька"

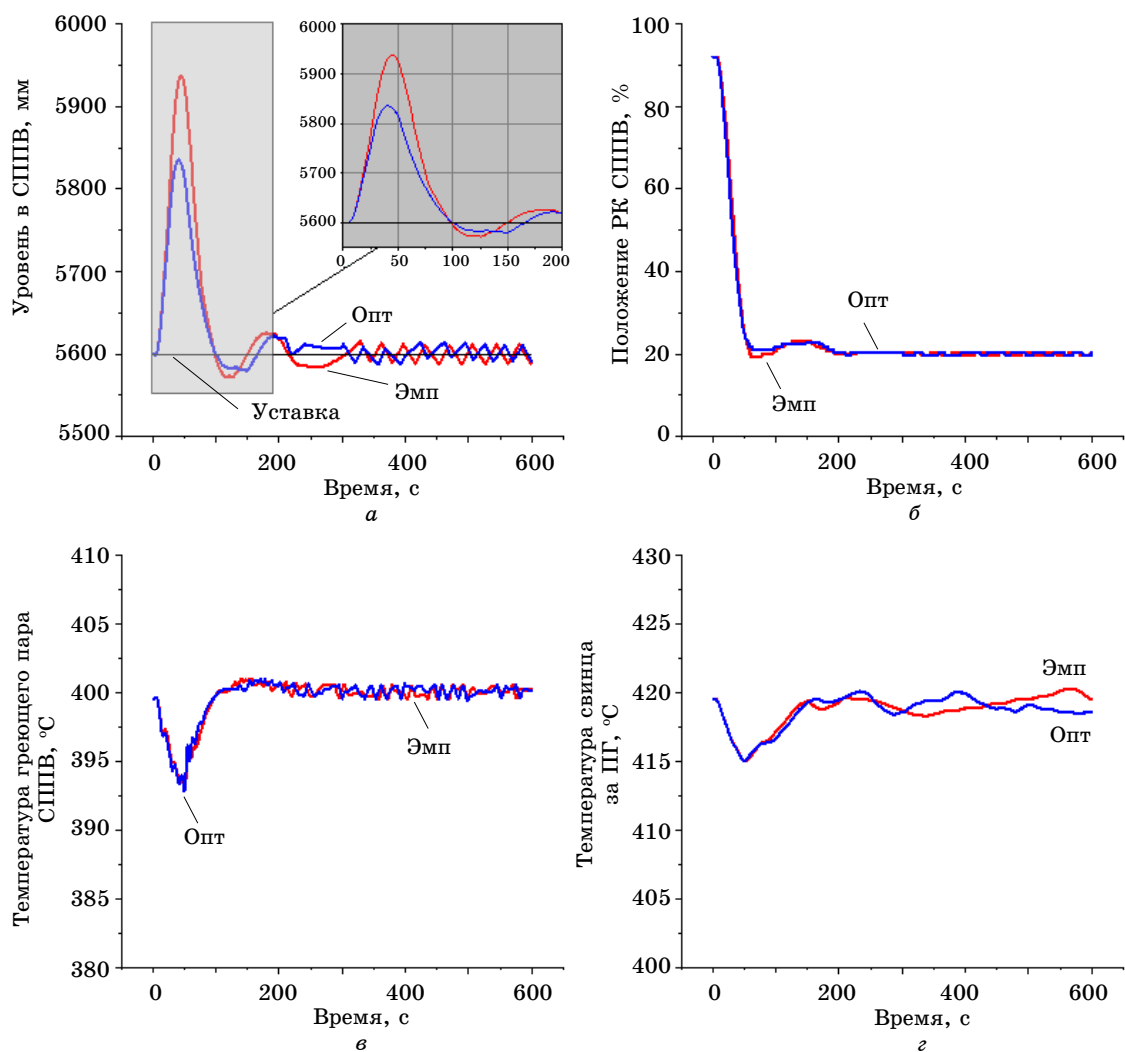


Рис. 8. Основные параметры тракта питательной воды в режиме "разгрузка"

В режиме "отключение ПЭН-1" (рис. 9, а) максимальное снижение уровня для обеих настроек регулятора получается одинаковым за счет того, что отключение питательного насоса вызывает заметное снижение давления перед РК уровня СППВ, что приводит к его полному открытию (рис. 9, б). В дальнейшем уровень восстанавливается, и его заброс из-за перерегулирования составляет 5892 мм для варианта с

оптимальными настройками против 5997 мм. Как и в режиме "разгрузка", ухудшения качества регулирования смежных параметров не наблюдается (рис. 9, в, г).

Значения критериев качества переходного процесса на исследуемых режимах при оптимальных настройках регулятора и при настройках, найденных эмпирическим путем, приведены в табл. 2.

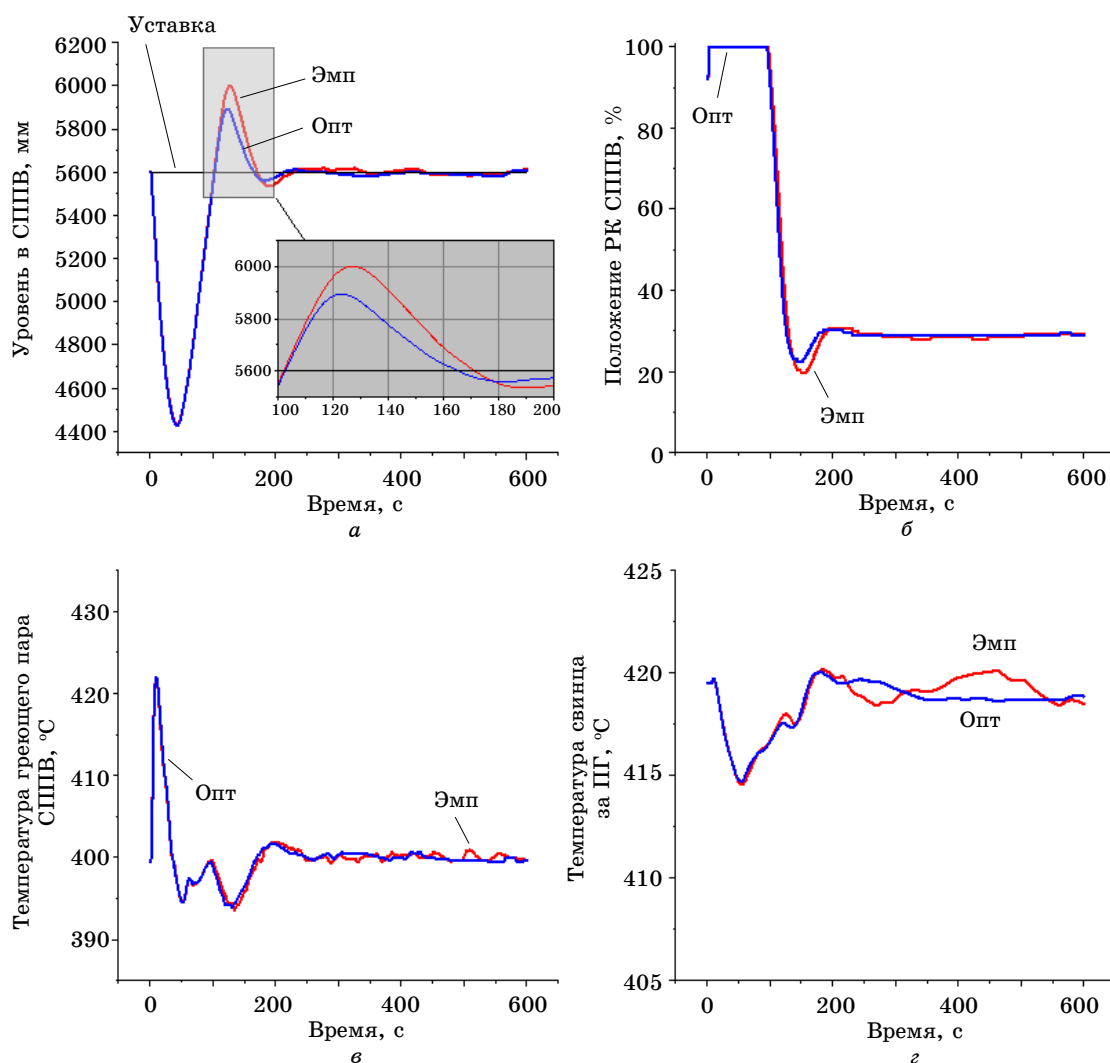


Рис. 9. Основные параметры тракта питательной воды в режиме "отключение ПЭН-1"

Таблица 2

Значения критериев качества переходного процесса на исследуемых режимах

Режим исследования	Эмпирический метод				Оптимизационный расчёт			
	НР, мм·с	ВХК, с	МР, мм	ВДУ, с	НР, мм·с	ВХК, с	МР, мм	ВДУ, с
Ступенька	17683	82,4	131	36	12721	83,3	104	35
Разгрузка	13667	90,5	336	98	9109	87,8	235	99
Отключение ПЭН-1	83987	104,3	666	102	78510	95,7	673	102



В результате выполненного оптимизационного расчёта удалось получить следующие результаты:

- в режиме "ступенька" снижение НР и МР составило 28 и 20 %, ВХК и ВДУ остались на прежнем уровне;
- в режиме "разгрузка" НР, ВХК и МР уменьшилось на 33, 3 и 30 %, соответственно, ВДУ осталось на прежнем уровне;
- в режиме "отключение ПЭН-1" НР и ВХК снизились на 7 и 8 %, МР возросло на 1 %, ВДУ осталось без изменений. Это объясняется тем, что отключение ПЭН-1 приводит к значительному снижению давления за ним и, соответственно, перепаду на клапане уровня СППВ, в результате чего регулирующий орган выходит на полное открытие и теряет регулирующие свойства;
- новые настройки регулятора уровня СППВ не оказали негативного влияния на другие регулируемые параметры, связанные с ним по гидравлической системе, такие как температура греющего пара СППВ и температура свинца на выходе из парогенератора.

### Заключение

В работе выполнена апробация технологии подбора настроек САР для различных режимов работы энергоблока, в рамках которой исследована динамическая модель турбоустановки и систем II контура энергоблока с реакторной установкой БРЕСТ-ОД-300. Рассмотрена задача по оптимизации настроек регулятора уровня СППВ. Полученное решение обеспечивает более высокое качество регулирования в сравнении с настройками регулятора, полученными эмпирическим путем.

Результаты апробации свидетельствуют о высоком потенциале применения рассматриваемого подхода и разрабатываемого расчётного комплекса на практике, в частности, в рамках НИОКР по разработке новых энергоблоков.

Стоит отметить, что РКО РЭ позволяет производить подбор настроек регуляторов и по целевым функциям, не рассматриваемым в данной

работе, если это требуется для исследования объекта регулирования.

В рамках дальнейших работ планируется расширение функционала РКО РЭ и его испытания на базе решения оптимизационных задач для систем регулирования проектируемых РУ с жидкометаллическим теплоносителем.

### Литература

1. Адамов Е. О. Быстрый реактор со свинцовым теплоносителем БРЕСТ: от концепции к реализации технологии // Атомная энергия. 2020. Т. 129. № 4. С. 185—194.
2. Драгунов Ю. Г., Лемехов В. В., Смирнов В. С., Чернецов Н. Г. Технические решения и этапы разработки реакторной установки БРЕСТ-ОД-300 // Атомная энергия. 2012. Т. 113. № 1. С. 58—64.
3. Филин А. И., Цикунов В. С., Попов С. В. Разработка схем, оборудования и режимов работы второго контура перспективной АЭС // Теплоэнергетика. 2008. № 6. С. 27—31.
4. Расчётная модель турбоустановки и систем II контура энергоблока с РУ БРЕСТ-ОД-300 (РМТУ БРЕСТ-ОД-300). Свидетельство РФ о государственной регистрации программы для ЭВМ №2018611038 от 22.01.2018 г.
5. Баум Ф. И., Козлов О. С., Паршиков И. А., Петухов В. Н., Тимофеев К. А., Щекатуров А. М. Программное обеспечение *SimInTech* для программирования приборов систем управления // Атомная энергия. 2006. Т. 113. № 6. С. 354—357.
6. Кавун О. Ю., Исполатов Д. Н., Куно М. Я., Фейман В. Г. Расширение возможностей теплогидравлического программного комплекса для моделирования процессов, происходящих в оборудовании АЭС // Теплоэнергетика. 2005. № 1. С. 34—40.
7. Попов А. В., Федорович Е. Д., Кулаков Е. Н., Денисова И. Б. Влияние количества конденсатных насосов на надежность работы питательных насосов в схеме энергоблока БРЕСТ-ОД-300 // Материаловедение. Энергетика. 2021. Т. 27. № 1. С. 31—40.
8. Кулаков Е. Н., Попов А. В., Кругликов П. А. Оптимизация параметров системы поддержания температуры воды на входе парогенератора энергоблока с реактором БРЕСТ-ОД-300 // Технологии обеспечения жизненного цикла ядерных энергетических установок. 2021. № 3(25). С. 9—21.
9. *pSeven* [Электронный ресурс]. URL: <https://www.datadvance.net/ru/product/pseven/> (дата обращения: 14.09.2022).

## APPROBATION OF TECHNOLOGY FOR OPTIMIZING THE SETTINGS OF EQUIPMENT REGULATORS BASED ON A DYNAMIC MODEL OF A POWER UNIT

T. K. Zhaboev, A. S. Sachenko, S. V. Sumarokov  
Nuclear Safety Institute of the RAS, Moscow, Russia

A. V. Popov, E. N. Kulakov  
Joint-Stock Company "I. I. Polzunov Scientific and Development Association  
on Research and Design of Power Equipment", St. Petersburg, Russia

*The results of approbation of the calculation complex for solving the problems of optimizing the operating modes of a reactor plant (RU) are presented. Approbation was carried out on the example of solving the problem of optimizing the equipment controller settings based on a dynamic model of the second circuit of a power plant with a BREST-OD-300 reactor. The technology of searching for the settings of the level controller of the mixed heater of feed water for various modes of operation of the power unit was applied.*

*Keywords:* computational complex, multicriteria optimization, NPP, dynamic model, transient process, regulation, pSeven, SimInTech.

---

**Жабоев Тембулат Караканович**, заведующий лабораторией.

E-mail: tembulat@ibrae.ac.ru

**Саченко Алексей Сергеевич**, ведущий инженер.

E-mail: sachenko.a@ibrae.ac.ru

**Сумарокров Сергей Вячеславович**, заведующий отделом.

E-mail: sumarokov@ibrae.ac.ru

**Попов Алексей Валентинович**, инженер.

E-mail: popovav@ckti.ru

**Кулаков Егор Николаевич**, инженер.

E-mail: kulakoven@ckti.ru

*Статья поступила в редакцию 11 января 2023 г.*

УДК 004.9

DOI: 10.52190/2073-2597\_2023\_2\_13

EDN: JHPJIK

## О ПЕРСПЕКТИВНОСТИ ПРИМЕНЕНИЯ DATA MINING В ПРОЕКТИРОВАНИИ

В. А. Прудников, канд. техн. наук

Московский авиационный институт (национальный исследовательский университет),  
Москва, Россия

*Описана методология Data Mining, и проведен анализ применения возможностей OLAP-систем для координации работ по проектированию. Рассмотрены вопросы организации работы информационных систем с архитектурой, где сочетаются реляционные и многомерные хранилища данных. Методология Data Mining понимается в статье как готовое решение для формирования, передачи, хранения и обработки данных, предназначенных для получения информации об изделиях. Затронуты вопросы конфигурирования архитектуры серверных устройств в решениях, работающих совместно с центрами обработки данных.*

**Ключевые слова:** Data Mining, OLAP-системы, реляционные и многомерные БД, многомерные данные, гиперкуб, комплексность, параллельность.

В последнее время всё чаще рассматривается вопрос о практическом применении информационных систем в специализированных центрах предприятий различного профиля, например в диспетчерских службах аэропортов, серверных центрах торговых компаний. "Компьютерная периферия" становится всё более сложным и значимым элементом, позволяющим осуществлять связь между удалёнными пунктами, выполнять постоянный и периодический информационный обмен.

Современные технические решения связаны с созданием автоматизированных систем (АС), работающих на основе стандартизированных правил, предусматривающих унификацию и универсализацию оборудования, используемого в коммуникационных и информационных устройствах. Это обуславливает продвижение этапа массового применения таких устройств, позволяет обрабатывать большие информационные массивы, образуемые в распределённых и сосредоточенных системах, расширяет возможности проведения научных работ группами учёных, находящихся на значительных расстояниях друг от друга, а также может способствовать переходу на качественно другой уровень агрегатного проектирования и улучшению выполнения конструкторских работ, технологических подготовок производств и других инженерных задач.

Использованием вычислений с обработкой больших массивов данных инженеры-прикладники аргументируют улучшение показателей качества создаваемой продукции. Проявляется проблема интеграции работ инженеров — проектных, конструкторских и других специальностей — с деятельностью ИТ-специалистов. Обеспечение инверсионных возможностей АС позволит проводить подробный анализ обрабатываемых данных и целей, для которых они собираются, и использовать наиболее подходящие алгоритмы моделирования процессов обработки данных, полнее применять автоматизиро-

ванные инженерные и подобные им системы. АС — это не единственная причина запаздывания построения решений с использованием вычислительных систем, но их влияние вполне заметно даже при поверхностных оценках. Например, методическое обеспечение АС в каждом случае применения задач моделирования необходимо конкретизировать. Следует учитывать роль и место АС в общественной деятельности для повышения эффективности их использования, снижения вызываемых ими негативных явлений и получения приемлемых решений [1].

**Data Mining в автоматизированных системах**

В отечественной литературе рассматривают вопросы внедрения и практического использования при создании АС методологии Data Mining, уже применённой и стандартизированной в ряде других стран. Методология Data Mining, также называемая "обнаружением знаний в данных", включает технологии нахождения новых, действительных и потенциально полезных знаний в создаваемых базах данных (БД) общего и специализированного применения. Программно-аппаратные решения таких АС создают уже не только для анализа БД в специализированных центрах обработки данных, они появляются в хостингах пересылок — майнинговых фермах и в распределённых БД. Data Mining охватывает нескольких научных областей, главные из которых связаны с системной организацией БД, искусственным интеллектом. Эта методология применена в устройствах, работающих в режиме реального времени, и использует модели, которые формируют, хранят, обрабатывают информацию и создают вспомогательные специально организованные БД в многомерных гиперкубах [2]. Похожий анализ данных применен в системах Big Data. Не углубляясь в проблемы передачи,

хранения и обработки данных, следует признать, что при более чем сорокалетней инициативе процесс внедрения новых систем всё ещё не принёс удовлетворительных объяснений того, как следует правильно организовать архитектуры серверных устройств для их эффективного производственного применения, а также каких методических рекомендаций следует придерживаться проектировщикам таких систем.

Уход от прямых ответов на эти вопросы делает невозможным проведение содержательного анализа, направленного на улучшение процесса коллективного решения задач, появляющихся в ходе серверных сборок, и осуществления обменов между серверами в потоках реального времени.

Замечено, что до сих пор некоторые перспективные и прогрессивные компании, например работающие в области проектно-строительных работ, не используют специализированные программные комплексы, а ограничиваются отдельными пакетами программ, такими, как *AutoCAD*, *NanoCad*, *ArchiCAD*, Компас, КЗ-Коттедж, *Project Smeta CS*, обходя этим трудности работы с многомерными автоматизированными системами и особенностями организации пересылок данных. В Интернет-обсуждениях встречаются упоминания о работах с трёхмерными моделями строительных сооружений, зданий, территориальных комплексов, отдельных подсистем. Их серверное обеспечение ограничивается коммуникационными системами, как правило, применимыми в документообороте. Все такие решения существенно повышают производительность труда отдельного работника. О методах обработки решений, сочетающих совместную работу нескольких систем, включённых в строительные проекты, сообщений мало. Возможно, предприятия не хотят делиться "ноу-хау", но вероятно также, что специалисты не имеют доступа или возможности применять такие системы. В процессы включаются другие группы работников, выполняющие задачи совместного монтажа и отладки разных подсистем и др. без применения АС. Следующий шаг подразумевает повышение производительности работы коллективов за счёт применения улучшенных серверных возможностей, но на практике заметно вторичное запаздывание исполнения работ по причине участия серверных групп в общем и, как правило, удалённом процессе изготовления изделий. Отмечать улучшение работы от совершенствования электронного документооборота руководители компаний явно не спешат. Тормозится внедрение предусмотренных национальным ГОСТом форм для прохождения инженерных документов по рабочим станциям АС и в специализированных программных комплексах, применяемым в проектировании, конструировании и технологической подготовке производств. В чём дело? Вероятно, качество обработки инженерной документации вручную остаётся выше, чем спо-

собны предложить сервисы, предоставляемые коммуникационными системами. Работу бизнес-аналитиков можно рассматривать как часть проектировочной деятельности, в которой целью ставится получение и оценивание новых данных, не заметных на других этапах инженерных работ.

В России сообщения о методологии *Data Minig* появились после внедрения на отечественных предприятиях коммуникационных систем, которые не всегда предусматривались как изначальные, особенно на этапах межсервисных коммуникаций. Из-за этого возникает необходимость перенастройки уже смонтированных АС.

Персональные вычислительные устройства в *Data Minig* не рассматриваются отдельно от информационных систем. Технологии *Data Mining* и им подобные ранее не обсуждали вообще. Между тем их постепенное внедрение в инженерные процессы создает улучшенный инструмент для решения конкретных практических задач сквозного проектирования, конструирования, испытаний, управления. Разделение работ на проектные (теоретические) и производственные (практические) уже многократно апробировалось в разных странах во всём мире. Появившиеся информационные системы вносят и будут вносить некоторые поправки не только в инженерные разработки, но и в серийное изготовление изделий. Такие системы не предусматривают обязательную обратную связь от производственного персонала. Однако в реальных обстоятельствах ее приходится продумывать и предусматривать, что также обеспечит улучшение результатов работ.

### Примеры применения *Data Mining* в информационных системах

Опыт применения информационных систем в электронном обучении демонстрирует возможности использования компьютеров как средств коммуникации, но многомерные информационные массивы при этом не рассматривают, так как для коммуникационных систем в первую очередь проявляются проблемы лингвистического обеспечения [3]. При работе с многоязычным контентом и с повторением основных проектных модулей потребуется применение программы языковых переводчиков. Специализированные программные комплексы, например для строительства, используют значительно меньшие по объему, чем при обучении, текстовые библиотеки, содержащие в основном нормативные документы.

Применение сложных информационных устройств, способных хранить, передавать и обрабатывать большие объёмы данных, заставляет обращать внимание на развитие технологий, подобных *Data Mining*, и учитывать их возможности. *Data Mining* включает в себя и поз-



воляет выполнять совместную и одновременную работу БД, считавшихся ранее отдельными (объектные БД, реляционные БД, хранилища БД, серверы сотовой связи, почтовые серверы и др.). Сложность внедрения технологий *Data Mining* в России заключается в отсутствии отечественного серверного оборудования и методик его организации в системы, для этого обязательно используют международную кооперацию. Логика совместной работы, осуществляемой с помощью электронных устройств, кажется единой и универсальной, распространена во многих странах; разрозненная информация может объединяться в единых БД при условии специального универсального задания индексации измерений.

Недостатком применения *Data Mining* является низкий уровень научной подготовки объектов анализа. Для преодоления этого создают новые версии методического обеспечения. На сегодняшний день оценки полезности и эффективности результатов применения систем, реализующих технологии *Data Mining*, приближительные. Некоторые процессы в производствах или их частях носят одноразовый неповторяющийся характер, и управление проектами при уже сформированных и запущенных в ход действиях без их приостановки на участках исполнения работ не допускаются. По этим причинам не следует, по крайней мере пока, полностью автоматизировать обработку информационных моделей, оставляя принятие решений оператору.

В настоящее время в периодических изданиях вопросы внедрения *Data Mining* обсуждается мало. Присутствуют, главным образом, публикации в Интернете, в которых сообщается о некоторых примерах и возможностях использования этой методологии:

1) *В кредитовании, страховании, связи, банковской деятельности, социальной экономике* [4]. Например, компании Яндекс, "ВымпелКом" сообщают об использовании программного сервиса "Сбор-Аналитика", с помощью которого обрабатывается информация по кредитам, выполненным транзакциям, осуществляется борьба со спамом. О расширении подобной деятельности свидетельствует большое количество вакансий в области аналитики данных, которые компании размещают на сайтах для поиска работы, например на портале *hh.ru*.

2) *В сферах торговли и информационной безопасности* [5]. Например, технологии применяются в сетях гипермаркетов "Лента", *Amazon*. Кроме того, производители могут прогнозировать износ производственных активов и планировать их техническое обслуживание и ремонт. Существуют готовые решения для коммерческих предприятий, например интеллектуальная система поддержки принятия решений *Jet Galatea* от российской компании "Инфосистема Джет".

3) *В производстве*. Предприятия могут осуществлять анализ больших объемов данных,

согласовывать планы поставок с прогнозами спроса, обнаруживать проблемы производств на ранних стадиях, устанавливать формы и нормы оплаты труда и инвестирования и т. п.

4) *В социологии*. МВД России, например, с 2016 г. использует в некоторых регионах систему "Зеус" [6].

5) *В медицине*. В частности, британский стартап *Babylon Heath* собирает всю информацию о здоровье клиентов, их образе жизни и привычках, а затем алгоритм строит гипотезы и предлагает варианты обследования, лечения и даже рекомендует конкретных врачей и клиник [7]. Подобному анализу подвергают симптомы и болезни, наблюдаемые у пациентов. Результаты собирают в информационных системах для дальнейшей обработки. Изучение сочетаний болезней и симптомов, встречающихся наиболее часто, помогают точнее ставить диагноз. [8].

6) *В электронном обучении*. Сообщается о ряде программных комплексов, предложенных для применения в процессах обучения в высших школах [9].

### Обработка многомерных данных в *Data Mining*

Типовые информационные модели интенсивно совершенствуют для различных конфигураций информационных архитектур [10]. В их создании участвуют объединения научных институтов и школ, международные производственные корпорации. Этап понимания работы таких архитектур, осмысление возможностей и назначения занимает длительное время. Для подготовки к работе с соответствующими АС специалистам требуется изучить основные положения компьютерного моделирования, а также методические рекомендации и пояснения от разработчиков и фирм-изготовителей оборудования.

Внедрение новых устройств сделало актуальным выполнение задач принятия решений не только для учёных, создателей серверных устройств, пользователей и наладчиков новой техники, но и для людей, прямо или косвенно вовлечённых в те процессы, где новые устройства были применены. Наиболее понятными оказались серверные устройства, применяемые в сетях гипермаркетов и супермаркетов, а также в сетевом маркетинге. По ним присутствует достаточно подробная описательная литература, поясняющая работу серверных устройств.

Логика формирования и обработки данных, организованных по стандартам *Data Mining*, базируется на создании многомерных цифровых пространств, которые подвергаются обработке по специальным образом сформированным правилам. Например, методами *Data Mining* создают так называемые "предсказательные алгоритмы", по которым осуществляется поиск решений по ассоциативным связям.

Поиск решений по ассоциативным связям является одним из популярных приложений

*Data Mining*. Суть решаемой задачи заключается в определении часто встречающихся наборов объектов в большом множестве их выборок [11]. Эта задача является частным случаем задачи классификации. Первоначально она встречалась при анализе тенденций в поведении покупателей в супермаркетах. Анализу подвергались данные о совершаемых покупках, которые покупатели складывают в тележку/корзину. Вероятно, это и послужило причиной появления второго часто встречающегося названия этой задачи — "анализ рыночных корзин" (*Basket Analysis*). Интерес при анализе данных представляет информация о том, какие товары покупают вместе, в какой последовательности, какие категории потребителей присутствуют в передаваемых данных, какие товары предпочитают, в какие периоды времени происходят события. Информация после обработки позволяет планировать закупку товаров, проведение рекламных кампаний, выкладку товаров и т. д.

В ассоциативном анализе исследуют последовательность происходящих событий. При обнаружении закономерностей в таких последовательностях с некоторой долей вероятности предсказывается появление таких же или подобных событий в будущем, что позволяет выполнять прогнозирование. Задача является разновидностью задачи поиска ассоциативных правил и называется секвенциальным анализом.

Основное отличие секвенциального анализа от поиска ассоциативных правил заключается в установлении отношений порядка без определения причин между обрабатываемыми наборами. Данные отношения могут быть определены разными способами. При анализе последовательности событий, происходящих во времени, объектами таких наборов являются события, а отношениям порядка соответствует хронология их появления. Например, задачи секвенциального анализа решают в телекоммуникационных компаниях при сборе и анализе данных об авариях на различных узлах сетей для оценок загруженности линий [12]. Информация о последовательности совершения аварий помогает в обнаружении неполадок и предупреждениях новых аварий. Если дополнительно присутствует ещё и знание о времени между сбоями, то можно предсказать не только факт его появления, но и время возможного последующего сбоя.

Метод аналитической обработки многомерных данных заключается в следующем. В процессе принятия решений пользователь генерирует гипотезы [13], которые проверяются и затем превращаются в законченные решения. Проверку гипотез осуществляют на основании информации об анализируемой предметной области. Удобным способом представления информации в человеческом понимании является зависимость между некоторыми значениями, возможность их соизмерения. Например, зависимость объемов продаж от региона, времени, категории товара. Другой пример — количество

выздоровливающих пациентов в зависимостях от применяемых средств лечения и возраста. В процессах анализа данных и поиска решений возникает необходимость в построении зависимостей между различными параметрами. Кроме того, число параметров может варьироваться в широких пределах. Технические устройства повышают интенсивность и научную ёмкость работ. Традиционные средства анализа оперируют данными, которые представляются в виде таблиц реляционной, как правило, уже проиндексированной БД, но и они не могут в полной мере удовлетворять требованиям адекватности модели моделируемому процессу или явлению. Поэтому следует отметить, что людям ещё рано полностью доверять искусственному интеллекту, понимая, что он лишь создаёт удобства в повышении скорости рутинной обработки сведений об исследуемом объекте.

#### **OLAP-системы для анализа многомерных данных**

Каким образом можно учитывать упомянутые выше технические устройства в проектных работах, когда состав, внешние формы и конфигурирование работ на начальном этапе могут быть известны лишь примерно. В логике формирования проектных решений по данным, получаемым из систем их обработки, отсутствует описание больших участков работ, посвящённых архитектурам информационных систем, работающим в режимах сбора, хранения и обработки многомерных данных. Между тем столь сложно организованные информационные системы могли бы быть использованы для проведения углублённого изучения вопросов, возникающих при тиражировании изделий, повышении их качественно важных характеристик, формировании типорядов для серийных выпусков продукции.

В произвольных заполнениях БД позволяют выполнить нумерацию ячеек (машинных слов), где хранятся данные. Разные типы данных собираются в разных группах. Для различных видов сравнения таких групп используют специальные отдельные программы, задающие реляционные связи. Правила установки реляционных связей — модельные и необязательно отражают реальные явления. Основоположник реляционной модели баз данных Э. Кодд изучал и рассматривал ее недостатки, указывая на невозможность объединять, просматривать и анализировать данные с точки зрения множественности измерений, то есть вполне понятным для аналитиков способом [14].

Реляционная обработка данных встречается при обработках таблиц или текстов вспомогательной документации на отдельных, иногда автономных серверах. Такая документация может не передаваться в центры обработки данных, так как не содержат информации об изделиях.

В общем случае данными могут быть массивы геометрических моделей, комплекты чертежей или иной документации. Можно ли их передавать в центры обработки данных с использованием исключительно реляционной БД. Скорее всего, многомерные системы этого не предусматривают.

Множественность измерений предполагает конечное представление данных в виде многомерных моделей. По осям измерений в многомерной модели откладывают параметры, относящиеся к анализируемой предметной области. Сами "измерения" могут собираться от отдельных серверных узлов, образуя группы, изначально выделенные для информационных обменов.

Одновременный анализ по нескольким измерениям определяется как многомерный анализ. Измерения могут представляться в виде иерархической структуры и собираться в ее верхних уровнях в классы. Несмотря на критику реляционных БД самим Э. Коддом, они продолжают использоваться. Как отмечает Щавелев Л. И., "реляционные модели БД были, есть и будут наиболее подходящей технологией для хранения данных" [15]. Автор [15] пишет, что необходимость совершенствования существует не в новой технологии БД, а в создании средств анализа, дополняющих функции существующих систем управления БД, достаточно гибких, чтобы предусмотреть и автоматизировать разные виды интеллектуального анализа, присущего OLAP-системам.

### Информационные системы для обработки многомерных данных

Многомерные информационные модели описывают с помощью "гиперкубов". Над гиперкубом могут выполняться срезы, вращения, консолидация и масштабирование. Для обработки данных для многомерных моделей используются специальные программные приложения.

Работа серверных устройств освещается в опубликованных статьях и сети Интернет [16]. Разнообразия серверных решений определяют различные архитектуры серверных сборок. Отмечаются перспективы управления потоками данных, суть которых заключается в формализации задач передачи данных на следующий шаг с возможной организацией различных дополнительных шагов при появлении информации в блоках управления для получения различных результатов. В таких обстоятельствах проявляющиеся проблемы связаны с возникновением альтернатив и поиском правильности дальнейших действий, которые оцениваются по различным ограничениям, таким как скорость, точность передачи данных, правильность определения адресата, системность управляющего воздействия и др.

Сложная программно-аппаратная организация, используемая при выполнении анализа многомерных данных, предполагает большое количество вариантов серверных решений и разнообразие программных комплексов, задействованных для обработки конечных и промежуточных данных. Сбалансированная система сбора и обработки данных позволит не повторять ранее произведенные часто рутинные объемы работ и сконцентрировать усилия на показателях, улучшающих потребительские свойства выпускаемых продуктов.

Информационные группы программно-аппаратных устройств должны работать комплексно и с учетом соседних серверных групп и центров обработки данных, создавая вектор усилий в направлении общих целей. На промежуточных этапах должны выполняться задачи по выявлению качественно важных условий, созданию предложений по их улучшению и анализу внедренных решений, что способствует совершенствованию программно-аппаратных решений для компьютерного моделирования в части иллюстрирования расчетов, математической обработки данных, отладки в программных кодах и тому подобное. Это подразумевает непрерывные параллельные работы по улучшению сервисов. Специалисты серверных служб в настоящее время ориентированы на улучшение в коммуникационной части устройств, создание же программ расчетов по математическим моделям и способам обработки и обновления многомерных хранилищ данных не обсуждается.

Вместе с тем, использование многомерных БД в OLAP-системах имеет следующие преимущества: поиск и выборка данных осуществляются значительно быстрее, чем при многомерном концептуальном подходе для реляционных БД. Для него база многомерных данных денормализована и содержит заранее агрегированные показатели, что обеспечивает оптимальный доступ к запрашиваемым ячейкам и не требует дополнительных преобразований при переходе от множества связанных таблиц к многомерной модели.

Многомерные БД справляются с задачами включения в информационную модель разнообразных встроженных функций (реализуемых в программных приложениях), тогда как объективно существующие ограничения языка SQL делают выполнение этих задач на основе реляционных БД достаточно сложным, а иногда и невозможным.

Многомерные БД чувствительны к изменениям в многомерной модели. При добавлении нового измерения приходится изменять структуру всей БД, что влечет за собой большие затраты времени на подготовку серверной обработки данных. Многомерные системы образуют особую информационную среду со свойствами обработки общего информационного пространства по математическим алгоритмам и разработанным специально для них алгоритмическим

правилам. В результате появляются новые, не достигаемые ранее иными средствами, методы получения результата. Происходит формирование многомерных пространств с установленными связями обработки между его элементами, в которых представление таких пространств применимо, например, через задание подробных картографических сеток всех числовых и других элементов множеств [17].

Появились новые схемы с общим названием "графовые алгоритмы". Особенностью реализаций таких алгоритмов с применением компьютерных систем является их визуализация, позволяющая дополнительно демонстрировать на экранах результаты моделирования процессов прогнозирования (в том числе и срезов многомерных изображений). Модели такого рода обладают специфическими свойствами. В них создаётся эффект демонстративности (показательности) представляемых результатов в удобной для понимания форме [18]. Проблемность понимания работы подобных устройств требует подробных пояснений от квалифицированных специалистов, работающих в областях применения информационных систем. Встречающиеся на практике информационные системы могут не образовывать группы данных в измерениях целиком, а собираться из разных частей для обработки в информационных центрах в соответствии с их назначением.

### Применение *Data Mining* в проектировании

Особенностью информационных средств, использующих *Data Mining*, является формирование специальных правил обработки численных значений данных и адресов, в которых эти данные собираются и хранятся. Разрабатываются специальные компьютерные модели как для формулирования задач, так и для интерпретации решений. Используя методологию *Data Mining*, можно создавать шаблоны чертёжных документов, перемещать в реальном времени элементы по координатным сеткам карт и т. п. Это позволит формировать многомерные изображения и более подробно выполнять их моделирование. Применение программ обработки моделей имеет практический интерес для разных производств. Например, их можно использовать в системах автоматизированного проектирования, в которых вновь создаваемые решения сочетаются с обработкой решений в базах информационной поддержки производства изделий [19]. Это потребует усовершенствования интерфейсов программных комплексов внутренних сетей, создания для расширенных алгоритмов специальных закладок и страниц.

Инженерный персонал предприятий оказывается с необходимостью вовлечен в процессы настройки и конфигурирования информационных систем даже в тех случаях, когда произво-

дители аппаратуры ещё не пришли к окончательным стандартизированным решениям в части автономных работ узловых и лепестковых серверов и сеансов с центрами обработки данных. Такие ситуации известны и в проектно-строительных работах, они рассматриваются как резервируемые системы. Ввиду ограничений, с которыми приходится работать в современных условиях, отмечать широкое применение научно-практических результатов в проектировании преждевременно. Появляется и апробируется новая по своему назначению аппаратура, но отмеченные выше ограничения не позволяют полностью использовать ее функциональные возможности. Развёртывание прогресса по разным научно-техническим направлениям создаёт возможность для обеспечения комплексного развития внедряемого технического оборудования и улучшения качества создаваемой продукции одновременно с выполнением предварительного моделирования. Именно это является предметом споров и изучения, в том числе и специалистов по качеству [20].

### Заключение

Разумно понимать внедрение технологий *Data Mining* как следующий шаг к нормализации производственных процессов и отношений. Не следует считать, что благодаря использованию *Data Mining* автоматизированные системы станут обладать некими уникальными волшебными возможностями. Проявляются проблемы внедрения таких АС, связанные с нехваткой грамотных специалистов и необходимостью освоения дополнительных специальных знаний персоналом, применяющим АС.

Область применения технологий *Data Mining* расширена до использования в производстве. Вероятно, поэтому нет точной информации, какие системы обрабатывают данные, например на базе операционных систем *Mac OS*. Недоступна полная техническая документация на программно-аппаратные решения компании *Microsoft*, а также на системы серверных поддержек широко внедряемых технических устройств на базе операционных систем Андроид. Возможно появление подобных устройств со своей спецификой или принципиально других, работающих на иных способах организации метамоделей. Теория принятия решений неспособна дать полного ответа о преимуществах и недостатках той или иной системы без их подробных технических описаний. Для обсуждаемых в отечественных публикациях алгоритмов работы информационных систем приводят различные пояснения, но информация от компаний-производителей часто появляется с заметным запаздыванием после внедрения самих устройств. Постепенное прояснение вопросов в области программирования, математического обеспечения, о типах и составе



БД раскрывает особенности работы таких систем, но не может считаться уже сегодня предельно ясным и прозрачным для применения в широком масштабе. Требуются заметные усилия и время на их усвоение.

Первым условием для внедрения и распространения отечественных информационных систем следует считать рост и развитие производств электронной промышленности. Усилия, способствующие пониманию возможностей полезного применения информационных систем, следует предпринимать в областях научного сотрудничества. Повышение уровня научной обоснованности проектов проектирования при использовании информационных систем в настоящее время понимают как демонстрирующие первые примеры удачного применения устройств, которые следует поддерживать, например реализацией образовательных программ, международным сотрудничеством. В правильном, удобном и эффективном применении таких систем заинтересованы не только специалисты в области информационного обеспечения, но и специалисты из широкого перечня других областей.

Проектно-строительные работы — хороший пример создания комплексных решений на основе синтеза готовых подсистем в единых решениях. Информационные системы добавляют сложности к обработке свойств получаемых результатов, но позволяют выделять в модельных обработках преимущественные решения, например, по показателям качества. Архитектурное проектирование обязательно включает проектирование строительных конструкций, внутренние и подключаемые системы, архитектурный дизайн и прочее. Сложно представить, какая архитектура информационных систем наиболее полно сможет обеспечить и проектные работы, и этап создания архитектурных решений в процессе исполнения строительства.

Некоторые приведенные в данной работе утверждения, возможно, требуют дополнительной аргументации и вызовут несогласие у специалистов, использующих привычные методы для организации работы серверных устройств. Автор считает, что возможности методологии *Data Mining* представлены в открытых источниках неполно. *Data Mining* не следует понимать узко только как контроль и обеспечение прохождения информационных потоков, пригодность данной методологии для инженерных задач определяют ее возможности обработки, интерпретации и использования информации, в том числе многомерных данных.

### Литература

1. Майнинг ферма: что это такое, как ее собрать и стоит ли это делать в 2023 году [Электронный ресурс]. URL: <https://news.garantex.io/majning-ferma->

[chto-eto-takoe-kak-ee-sobrat-i-stoit-li-eto-delat-v-2023-godu/](https://news.garantex.io/majning-ferma-chto-eto-takoe-kak-ee-sobrat-i-stoit-li-eto-delat-v-2023-godu/) (дата обращения: 21.03.2023).

2. Барсегян А. А., Куприянов М. С., Холод И. И., Тесс М. Д., Елизаров С. И. Анализ данных и процессов: учеб. пособие. Изд. 3, перераб. и доп. — СПб.: БХВ-Петербург, 2009. — 512 с.

3. Sopan Talecar. The role of mathematics in machine learning [Электронный ресурс]. URL: [https://www.researchgate.net/publication/369452709\\_THE\\_ROLE\\_OF\\_MATHEMATICS\\_IN\\_MACHINE\\_LEARNING](https://www.researchgate.net/publication/369452709_THE_ROLE_OF_MATHEMATICS_IN_MACHINE_LEARNING) (дата обращения: 27.03.2023).

4. Наука о данных: мат. Междунар. науч.-практ. конф. Санкт-Петербург, 5—7 февраля 2020 г. — СПб.: Изд-во СПбГЭУ, 2020 [Электронный ресурс]. URL: <https://issras.ru/docs/sbornik05-02-2020.pdf> (дата обращения: 04.04.2023).

5. Студенческая наука для развития информационного общества: сб. материалов VIII Всерос. науч.-техн. конф. Ч. 2. — Ставрополь: Изд-во СКФУ, 2018 [Электронный ресурс]. URL: [https://is.ncfu.ru/files/nauch/VIII\\_2018-05\\_p\\_2.pdf](https://is.ncfu.ru/files/nauch/VIII_2018-05_p_2.pdf) (дата обращения: 04.04.2023).

6. Зеус. Программно-аналитический модуль мониторинга социальных сетей [Электронный ресурс]. URL: [https://www.tadviser.ru/index.php/Продукт:Зеус\\_Программноаналитический\\_модуль\\_мониторинга\\_социальных\\_сетей](https://www.tadviser.ru/index.php/Продукт:Зеус_Программноаналитический_модуль_мониторинга_социальных_сетей) (дата обращения: 04.04.2023).

7. Крутько В. Н., Донцов В. И., Митрохин О. В., Матвеев А. А., Ермакова Н. А., Потемкина Н. С. Искусственный интеллект для здоровьесбережения и развития личностного потенциала (Обзор) [Электронный ресурс]. URL: <http://www.isa.ru/proceedings/images/documents/2020-70-3/86-100.pdf> (дата обращения: 04.04.2023).

8. Михеева Е. В. Информационные технологии в профессиональной деятельности: учеб. пособие для студ. сред. проф. образования. Изд. 7, стер. — М.: Издательский центр "Академия", 2008.

9. Прудников В. А. О выборе автоматизированных компьютеризированных систем для реализации учебных программ в ВУЗах // Информационные технологии в проектировании и производстве. 2016. № 3. С. 73—78.

10. Саймон Г. Науки об искусственном. (The Sciences of the Artificial). Изд. 2 / Пер. с англ. Наппельбаума Э. Л. — М.: Едиториал УРСС, 2004.

11. Паклин Н., Орешков В. Бизнес-аналитика: от данных к знаниям: учеб. пособие. Изд. 2, испр. — СПб: Питер, 2013.

12. Матвейкин В. Г., Дмитриевский Б. С., Ляпин Н. Р. Информационные системы интеллектуального анализа. — М.: Машиностроение, 2008. — 92 с.

13. Халин В. Г. и др. Теория принятия решений. В 2-х т. Т. 1: учеб. и практ. для бакалавриата и магистратуры. — М.: Изд-во Юрайт, 2016. — 250 с.

14. Кузнецов С. Обзор статьи "The Birth of the Relational Model" [Электронный ресурс]. URL: [http://citforum.ru/database/digest/dig\\_1410.shtml](http://citforum.ru/database/digest/dig_1410.shtml) (дата обращения: 04.04.2023).

15. Щавелев Л. И. Способы аналитической обработки данных для поддержки принятия решений 01.08.1998 [Электронный ресурс]. URL: <https://www.osp.ru/news/articles/1998/0901/13031591?ysclid=lg4z13gx2m655393822> (дата обращения: 04.04.2023).

16. Основные виды серверов: назначение и особенности [Электронный ресурс]. URL: [https://galtsystems.com/blog/start/osnovnye\\_vidy\\_serverov\\_](https://galtsystems.com/blog/start/osnovnye_vidy_serverov_)

naznachenie\_i\_osobennosti/ (дата обращения: 04.04.2023).

17. Долгова Ю. К. О программных комплексах по планированию производства. — М.: Экономика, 2015. — 188 с.

18. Что такое гиперкуб? Построение тессеракта [Электронный ресурс]. URL: <https://dzen.ru/a/W24HSoERVQSpYrMM> (дата обращения: 04.04.2023)

19. Норенков И. П. Основы автоматизированного проектирования: учеб. для вузов. Изд. 4, перераб. и доп. — М.: Изд-во МГТУ им. Н. Э. Баумана, 2009. — 430 с.

20. Ларичев О. И., Браун Р. В. Количественный и вербальный анализ решений: сравнительное исследование возможностей и ограничений // Экономика и математические методы. 1998. Т. 34. Вып. 4. С. 97—107.

## ABOUT THE PROSPECTS OF USING DATA MINING IN DESIGN

V. A. Prudnikov

Moscow Aviation Institute (National Research University), Moscow, Russia

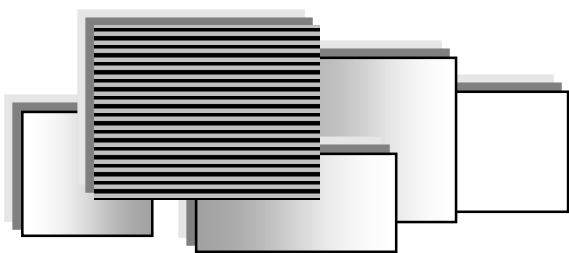
*The article is devoted to the description of the Data Mining methodology and the analysis of the use of the capabilities of OLAP systems for coordinating design work. The issues of organizing the work of information systems with an architecture that combines relational and multidimensional data stores are considered. The Data Mining methodology is understood in the article as a ready-made solution for generating, transmitting, storing and processing data intended to obtain information about products. Issues of configuring the architecture of server devices in solutions that work in conjunction with data centers are touched upon.*

**Keywords:** Data Mining, OLAP systems, relational and multidimensional databases, multidimensional data, hypercube, complexity, parallelism.

---

Прудников Виталий Анатольевич, доцент.  
E-mail: prudnikovs64@mail.ru

Статья поступила в редакцию 21 марта 2023 г.



## ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В РЕШЕНИИ ЗАДАЧ УПРАВЛЕНИЯ

УДК 004

DOI: 10.52190/2073-2597\_2023\_2\_21

EDN: JKWIIF

### ИСПОЛЬЗОВАНИЕ СВЁРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ РЕШЕНИЯ ЗАДАЧ РАСПОЗНАВАНИЯ ОБРАЗОВ В ПРОМЫШЛЕННОСТИ

Р. Р. Жалялов

Московский государственный технический университет имени Н. Э. Баумана  
(национальный исследовательский университет), Москва, Россия

*Приведены примеры использования технологий компьютерного зрения в российских и зарубежных компаниях. Рассмотрены актуальные проблемы промышленности, связанные с применением компьютерного зрения. Описаны различные варианты реализации компьютерного зрения на основе сверточных нейронных сетей. Дан обзор популярных архитектур сверточных нейронных сетей и проведен их сравнительный анализ.*

**Ключевые слова:** компьютерное зрение, *Computer Vision*, сверточные нейронные сети, идентификация изображений, распознавание объектов, сегментация изображений, затухающий градиент, *ImageNet*, *AlexNet*, *VGG*, *Inception*, *ResNet*, *SENet*, *EfficientNet*.

Технологии искусственного интеллекта находят все более широкое применение для решения актуальных вопросов автоматизации производственных процессов, повсеместного внедрения робототехнических систем, усложнения цепочек производственной логистики, повышения стандартов защиты жизни рабочих, ужесточения требований к качеству выпускаемой продукции. Фундаментом множества интеллектуальных систем обработки данных является компьютерное зрение (*Computer Vision*, *CV*).

В основе компьютерного зрения лежит принцип распознавания образов, аналогичный человеческому зрению. Зная отличительные признаки, можно определить, к какому классу отнести тот или иной образ, и использовать полученную информацию в дальнейшем для решения огромного спектра задач. В современных компьютерных системах используется многоступенчатая система извлечения признаков. Сам процесс определения отличительного признака на изображении возможен благодаря операции свертки.

В начале XXI в. в технологиях компьютерного зрения произошел значительный рост, обусловленный развитием вычислительной техники, созданием механизмов хранения больших массивов информации (датасетов) с размечен-

ными данными, появлением новых архитектур нейронных сетей с различными подходами к улучшению точности и эффективности распознавания образов.

Интерес к применению технологий компьютерного зрения в промышленности обусловлен рядом их преимуществ перед другими технологиями, а именно: сокращается время и стоимость производственной логистики; вредный для человека ручной труд автоматизируется там, где это раньше нельзя было сделать; контроль качества продукции осуществляется в реальном времени, что позволяет быстро реагировать на возможные отклонения; реализуется совместное взаимодействие человека и робота, становится возможной навигация и контроль робототехнических систем в пространстве для одновременного мониторинга множества мобильных объектов.

Вместе с тем, развитие и применение компьютерного зрения в промышленном производстве связано с трудностями, обусловленными отсутствием готовых данных в виде размеченных изображений для обучения модели под конкретную задачу. В ходе разработки *CV*-систем наибольшее количество времени разработчик тратит на составление и разметку обширного

датасета, включающего в себя краевые случаи и возможные отклонения, которые могут возникнуть в процессе производства. Необходимо учитывать, что в случае изменения производства, например включения в технологический процесс дополнительных операций, потребуется дообучение нейронной сети на новых данных, описывающих новый процесс.

### Примеры применения технологий компьютерного зрения в промышленности

Компьютерное зрение находит широкое применение в промышленности для автоматизации производственных процессов и повышения качества продукции. Технологию CV используют в промышленности для следующих целей:

- контроля качества продукции на производственной линии, в частности для обнаружения дефектов на поверхности продукта, анализа размеров и формы изделия. Например, технологии компьютерного зрения применяют для проверки качества компонентов автомобилей во время производства [1];
- определения размеров и веса груза, его классификации, а также для ориентации и навигации роботов-грузчиков на складах и в логистических центрах [2];
- определения положения и ориентации деталей на производственной линии [3];
- настройки роботов на точное выполнение сборочных и монтажных операций [3];
- контроля качества сварочных швов, плазменной резки и настройки робота [4, 5];
- мониторинга параметров производственных процессов в режиме реального времени.

В работе [6] приведены примеры из мировой практики применения компьютерного зрения в различных сферах промышленности, в частности упоминает о распознавании штрих-кодов и текстов различных видов, контроле качества выполнения в производстве, контроле уровня заполнения сосуда в фармацевтике. Каплун Д. отмечает, что в России компьютерное зрение внедряется в промышленность для контроля и автоматизации технологических процессов [7]. По мнению автора [7], компьютерное зрение позволяет ускорять производственный процесс, снижать затраты на персонал и оборудование, а также минимизировать влияние человеческого фактора.

Активное внедрение технологий компьютерного зрения в России проводят ООО "Яндекс", ООО "Диорам", АО "Петербургский тракторный завод", АО "Объединённая двигателестроительная корпорация".

В компании ООО "Яндекс" разработали и реализовали проекты беспилотного автомобиля и робота-доставщика на основе аппаратных средств, таких как лидары и радары, в сово-

купности с машинным обучением и компьютерным зрением [8].

В ООО "Диорам" использовали алгоритм одновременной локализации и построения карты *SLAM One*. Особенностью данного решения в области компьютерного зрения является возможность топологической навигации в масштабах города без данных от спутниковой связи. Система наложения полученных с камер и обработанных *SLAM One* исходных данных на загруженную ранее карту предоставляет возможность отказа от использования спутников для навигации [9].

В октябре 2021 г. АО "Петербургский тракторный завод" и компания-разработчик систем искусственного интеллекта для сельхозтехники ООО "Когнитив Роботикс" договорились о начале производства пробных промышленных партий беспилотной версии трактора "Кировец" К-7М [7];

В АО "Объединённая двигателестроительная корпорация" с помощью технологии компьютерного зрения контролируется качество лопаток турбины нового авиадвигателя ПД-8. В компании отмечают, что цифровые технологии позволят существенно сократить время производства двигателя [10].

### Задачи компьютерного зрения

Основными задачами в компьютерном зрении являются распознавание объектов, идентификация и сегментация изображений.

*Распознавание объектов* заключается в определении вида и границ объекта на изображении. На рис. 1 показано, как модель YOLO способна с достаточно высокой точностью распознавать заранее заученный набор объектов [11].

*Цель задачи идентификации изображений* — классифицировать изображения по всем отличительным признакам. Выявление таких ключевых областей и последующая классификация изображений возможны при помощи сверточных нейронных сетей.

*Сегментация изображений* — это процесс разбиения изображения на несколько сегментов или регионов, каждый из которых представляет собой группу пикселей с общими свойствами, такими как цвет, текстура, яркость или форма. Целью сегментации изображений является выделение объектов на изображении, различных структур и фоновых регионов, что облегчает дальнейшую их обработку и анализ. Так, например, архитектура *U-net* [13] представляет собой классический инструмент для сегментации, который можно адаптировать (как правило, меняя входную размерность), для конкретных целей. Результат работы этой модели продемонстрирован на рис. 2.



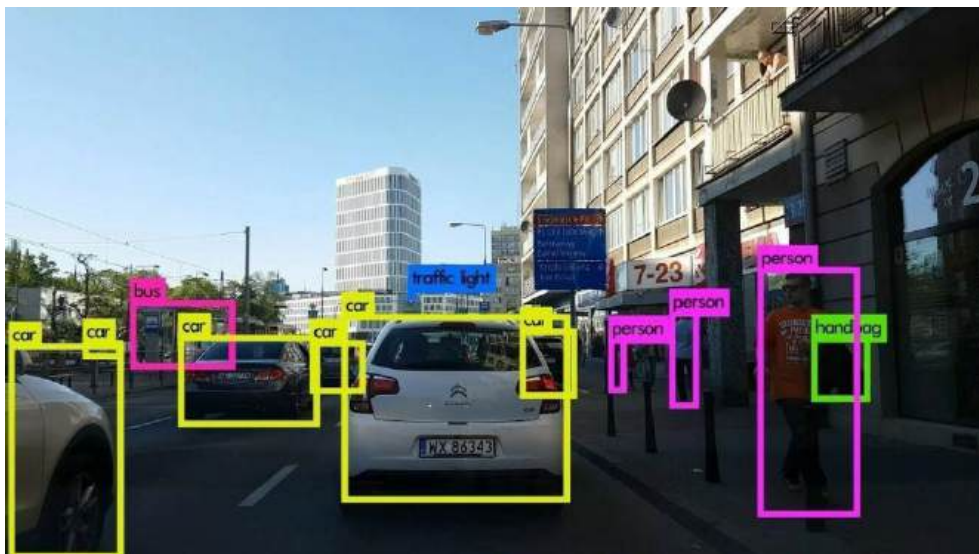


Рис. 1. Выделение классов объектов на фото улицы [12]



Рис. 2. Типичный пример сегментации изображений [14]

На основе распознавания объектов, идентификации и сегментации изображений на практике уже реализованы различные методы и модели для таких задач, как:

- выделение каркаса объекта (например скелета человека) и определение положения этого каркаса на изображении;
- распознавание текстов (рукописный текст, символы на дорожном знаке, номер машины, адрес дома или подъезда и т. д.);
- генерация объектов (создание объектов, похожих на заранее известный набор объектов, но при этом не совпадающих ни с одним объектом из этого набора). Например, в модели *Midjourney-v4* возможно задавать параметры для генерации самых различных объектов (люди, машины, символы, что угодно, что можно

описать текстом или изображением), как при помощи текстовых запросов, так и загруженных изображений [15];

- анализ видео (предсказание положения объекта на последующих кадрах на основе информации от предыдущих).

Основные задачи компьютерного зрения позволяют решать сверточные нейронные сети, о которых будет сказано ниже.

### Обзор архитектур свёрточных нейронных сетей

В основе большинства алгоритмов компьютерного зрения лежит распознавание объектов, с которым отлично справляются архитектуры на основе сверточных слоев. Принцип работы многослойной сверточной нейронной сети за-

ключается в извлечении паттернов — наборов отличительных признаков того или иного объекта, свойственных определенному классу анализируемого изображения и передаче их в последующий сверточный слой.

Глубина сверточной нейронной сети характеризуется числом слоев. Чем больше слоев, тем более глубокой является сеть. Увеличение количества слоев нейронной сети дает возможность обучить ее различать более сложные паттерны, что, в свою очередь, позволяет на выходе выдавать наиболее достоверный результат. Например, определить, изображен ли квадрат или треугольник, можно с помощью простой неглубокой сверточной нейронной сети, в то время как решить задачу отделения на изображении человека от промышленного манипулятора будет возможно только при использовании значительного количества слоев и фильтров. Таким образом, повышение качества работы нейронной сети связано с увеличением ее глубины, что неизбежно приводит к проблеме затухающего градиента, так как градиенты на первичных слоях оказываются настолько маленькими, что дальнейшее обучение сети становится невозможным.

Концепт архитектуры искусственных нейронных сетей предложил Ян Лекун в 1988 г. [16]. Созданные на основе свертки первые нейронные сети не могли продемонстрировать свой потенциал в распознавании образов из-за отсутствия качественных больших наборов изображений, а также из-за ограниченных вычислительных мощностей компьютеров в то время.

В 2009 г. был официально представлен датасет *ImageNet* [17]. База данных изображений *ImageNet* — одна из самых внушительных, включающая в себя 14197122 размеченных изображений [18]. Эта база данных пользуется особой популярностью для задач классификации, так как каждое загруженное изображение отражает только один объект и сопровождается

односложным точным описанием (например "робот", "самолет" и т. д.)

Бурный рост развития архитектур нейронных сетей начался в 2012 г., когда сеть *AlexNet*, используя базу данных изображений *ImageNet*, выиграла соревнование *ImageNet LSVRC-2012* и показала потенциал свёрточных нейронных сетей. Вслед за *AlexNet* появились архитектуры *VGG*, *Inception*, *ResNet*, *DenseNet*, *SENet*, *EfficientNet*.

Рассмотрим особенности упомянутых архитектур.

### Архитектура AlexNet

Помимо использования базы данных *ImageNet*, успех *AlexNet* обусловлен усовершенствованием графических процессоров, а также особенностями самой архитектуры сети (рис. 3).

Архитектура *AlexNet* состоит из 8 обучаемых слоёв: 5 свёрточных слоев (на рис. 3 обозначены как *CONV*) и 3 полносвязных слоя (на рис. 3 обозначены как *FC*), а также сеть содержит 3 слоя *max pooling* с фильтром размера  $3 \times 3$  и с шагом 2 (на рис. 3 обозначены как *Overlapping Max POOL*  $3 \times 3$ ,  $\text{stride}=2$ ). На вход подается цветное изображение размера  $227 \times 227$ . К нему применяется сначала свёртка размера  $11 \times 11$ , затем  $5 \times 5$ , а затем три свёртки размера  $3 \times 3$ .

Важной особенностью сети *AlexNet* по сравнению с предшествующими ей архитектурами является использование функции активации *ReLU* вместо сигмоиды. Использование *ReLU* избавило сеть от вычислительно-трудоемких операций с экспонентой. Кроме того, *ReLU* позволило избежать затухания градиентов во время обучения сети. Если выход сигмоиды близок к 0, то градиент будет близок к нулю, что затруднит обучение нейронной сети. В *ReLU* в области положительных значений производная всегда равна 1, так что проблемы затухания градиентов не возникает.

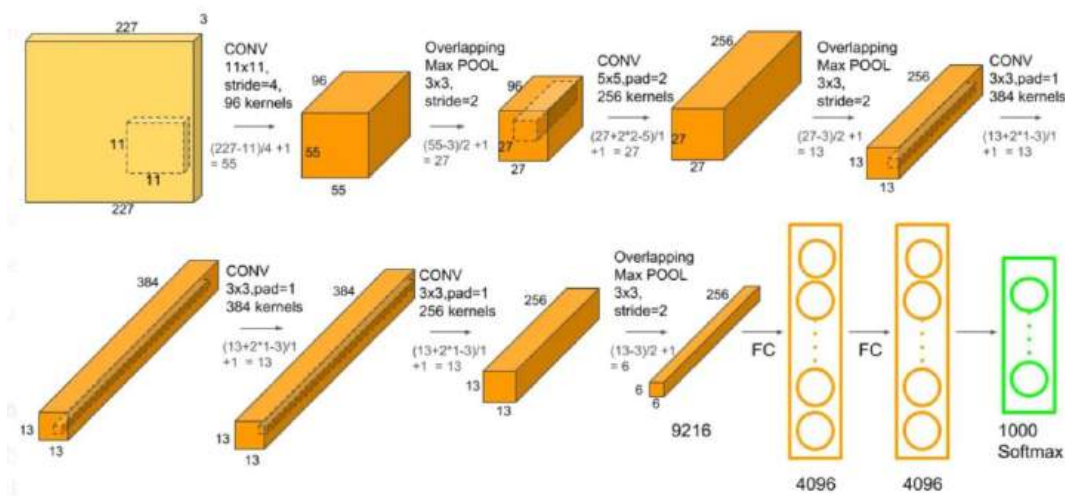


Рис. 3. Архитектура AlexNet [19]

### Архитектура VGG

Следующей архитектурой, повлиявшей на развитие свёрточных нейронных сетей, стала VGG, впервые представленная в 2014 г. командой разработчиков *Visual Geometry Group*. Две ключевые идеи архитектуры VGG — это повторяющаяся структура блоков и использование фильтров небольших размеров (рис. 4).

Система модулей архитектуры VGG дает возможность создавать более сложные и глубокие сети за счет использования повторяющихся блоков, каждый из которых складывается из ряда сверточных слоев с функциями активации *ReLU* (на рис. 4 они обозначены как блоки с черными ребрами *convolution + ReLU*). За сверточными слоями и функциями активации следует слой *max pooling*  $2 \times 2$  с шагом 2 (на рис. 4 обозначены как блоки с красными ребрами). Обозначенные выше слои образуют повторяющуюся структуру, называемую блоком. Модель версии *VGG-16* представляет архитектуру из пяти указанных блоков, за которыми следуют три полносвязных слоя с функциями активации *ReLU* (на рисунке 4 блоки с голубыми ребрами *fully connected*). Разработчики сети VGG использовали только свёртки  $3 \times 3$ , а не  $11 \times 11$  и  $5 \times 5$ , как это было в *AlexNet*. Таким образом, в каж-

дом слое оказалось меньше весов, а сеть стала глубже, благодаря чему сократилось количество обучаемых параметров при сохранении рецептивного поля (поля восприятия сети *receptive field* [21]).

### Архитектура Inception

В 2014 г. команда исследователей компании *Google* представила несколько иной подход к увеличению количества слоев, т. е. глубины сети, разработав архитектуру *Inception*. Данная архитектура подразумевает наличие блоков, одновременно применяющих к входному изображению различные по размеру свертки с фильтрами. Преимущество этого подхода заключается в более точной обработке объектов разных размеров на изображении, что стало возможным за счет особых модулей, которые авторы назвали *Inception*-модулями. Они объединяют несколько параллельных сверточных операций с различными размерами фильтров и с последующей конкатенацией результатов вдоль оси количества каналов. Чтобы при конкатенации размеры четырех выходов по высоте и ширине совпадали, используется паддинг.

Структура *Inception-v1* (первая версия сети *GoogLeNet*) состоит из 9 блоков (рис. 5).

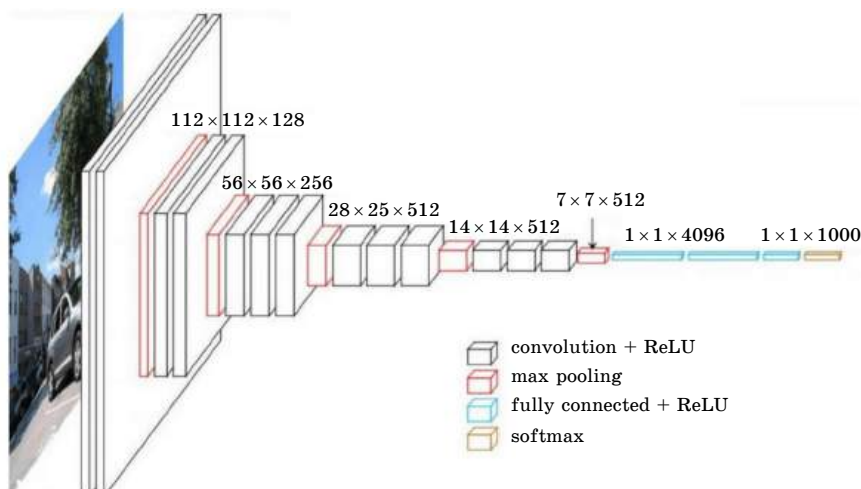


Рис. 4. Архитектура VGG-16 [20]

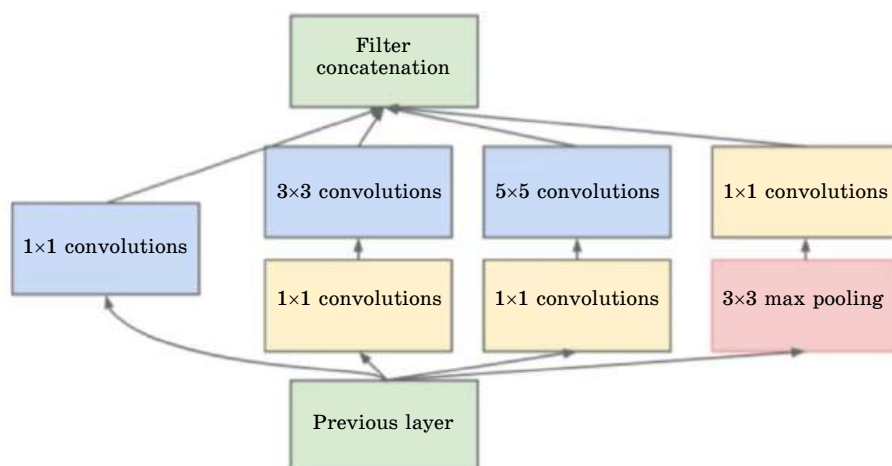


Рис. 5. Структура Inception [22]



Изображение одновременно поступает на четыре параллельных слоя. В крайнем левом (первом) слое применяется свертка  $1 \times 1$ . Во втором и третьем слоях применяются свертки  $1 \times 1$ , а также решающие задачи контроля количества каналов, после чего к выходным матрицам применяются свертки  $3 \times 3$  и  $5 \times 5$  соответственно. В последнем слое к изображению сначала применяется слой *max pooling*  $3 \times 3$  с шагом 1, и затем все слои соединяются (конкатенируются) в единую матрицу с помощью паддинга для каждого выхода в целях контроля размерности по высоте и ширине.

В *Inception* проблема затухающего градиента решается с помощью внедрения на начальном этапе и в середине сети двух вспомогательных выходных слоев — за счет их градиентов (то есть их использования в обратном распространении ошибки для обновления весов на начальных слоях) происходит качественное улучшение возможностей обучения сети.

### Архитектура ResNet

В 2015 г. появилась архитектура *ResNet*, разработчики которой предложили свое решение проблемы затухающих градиентов. Новшеством *ResNet* стали остаточные соединения (*residual connections*), которые передают информацию с предыдущих слоёв в последующие, пропуская слои между ними.

На рис. 6 представлен принцип работы данных соединений, где карта признаков обозначена матрицей  $X$ .

Несколько сверточных слоев (*conv layer* на рис. 6) преобразуют её в новую карту признаков  $F(X)$ . Без остаточных соединений при обучении градиентным спуском эти слои учитываются в

производной функции ошибки по весам как множитель  $F'(X) < 1$ , уменьшающий градиент. Остаточные соединения позволяют прибавлять карту признаков  $X$  к выходу этих слоёв  $F(X)$ . Тогда в производной функции ошибки по весам эти слои будут учитываться как множитель  $(F(X) + X)' = F'(X) + 1$ , большой единицы, уже не уменьшающий градиент.

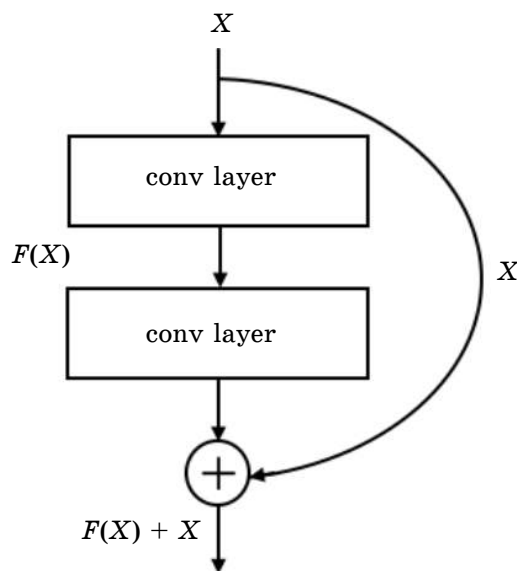


Рис. 6. Принцип работы остаточных соединений [23]

Архитектура *ResNet-34* (вариант реализации архитектуры с 34 слоями) может иметь разную глубину (от 34 слоев до 1000 и более). Такая глубина может быть достигнута за счёт применения нескольких остаточных соединений, которые решают проблему затухающего градиента в глубоких сетях. Архитектура *ResNet-34* представлена на рис. 7.

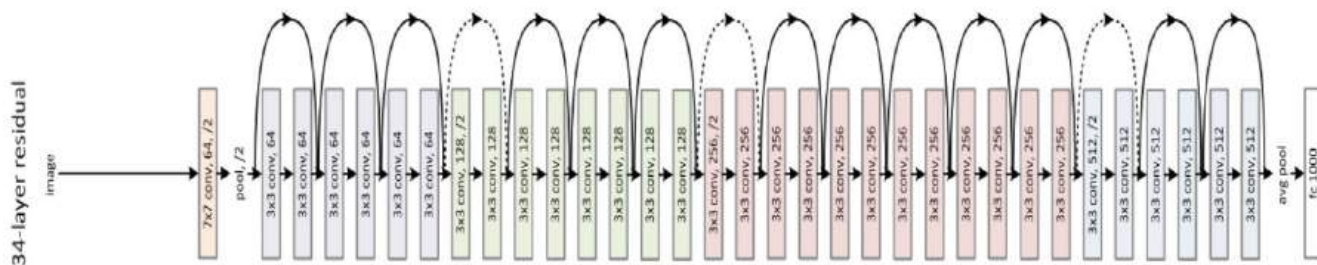


Рис. 7. Архитектура ResNet-34 [24]

### Архитектура SENet

В 2017 г. авторы архитектуры *SENet* предложили учитывать взаимодействие каналов выходных матриц и спроектировали *SE* блок (*squeeze&excitation*). Этот блок добавляет параметры (веса) каждому каналу свёрточного слоя, т. е. перевзвешивает каналы относительно друг друга [25]. Структура *SE* блоков изображена на рис. 8.

Для любого заданного преобразования ( $F_{tr}$ ) входных данных  $X$  (размера  $C' \times W' \times H'$ ) на тензор признаков  $U$  (размера  $C \times W \times H$ ), можно построить соответствующий *SE* блок. Тензор признаков  $U$  сначала проходит через операцию сжатия ( $F_{sq}$ ), которая создает дескриптор канала путем объединения карт признаков по их пространственным размерам ( $H \times W$ ). Функция этого дескриптора заключается в глобальном перераспределении весов ответов функций по каналам, что позволяет получать информацию из глобального поля восприятия сети (*receptive field*), которая будет использована всеми слоями сети. За агрегированием следует операция возбуждения ( $F_{ex}$ ), которая представляет саморегулирующийся механизм, принимающий входные данные и выдающий набор перераспределенных весов для каждого канала. Эти веса применяются к тензору признаков  $U(F_{scale})$ , и получаются выходные данные, которые могут быть переданы в последующие слои.

Добавление *SE* блока не является сложной вычислительной задачей, поскольку добавляется всего лишь один полносвязный блок с одним внутренним слоем. Таким образом, его добавле-

ние к любой архитектуре почти всегда улучшит качество сети, при этом не сильно увеличив количество обучаемых параметров. Кроме того, *SE* блок позволяет добавить дополнительную нелинейность.

### Архитектура EfficientNet

Чтобы улучшить работу нейронной сети, можно увеличивать ее глубину, ширину или разрешение входного изображения, но такая возможность зависит от доступных вычислительных ресурсов. В 2019 г. разработчики архитектуры *EfficientNet* предложили метод компаунд-шкалирования (*compound scaling*), который учитывает взаимосвязь между этими параметрами и позволяет одновременно настраивать оптимальную глубину, ширину и разрешение с помощью параметра  $\phi$  [26].

Варьируя параметр  $\phi$ , можно менять архитектуру сети *EfficientNet* от самой маленькой и легкой версии — *EfficientNet-B0* до тяжелой и наиболее точной — *EfficientNet-B7*. Оптимальным является выбор архитектуры на основе максимального качества (точности) при минимальном количестве выполняемых операций *FLOPS* (*Floating-point Operations Per Second*).

На рисунке 10 представлены 5 вариантов архитектур сети *EfficientNet*: стандартный вариант (a), вариант с увеличенной шириной сети (b), вариант с увеличенной глубиной сети (c), вариант с увеличенным разрешением сети (d), оптимальный вариант сети после применения метода компаунд-шкалирования (e).

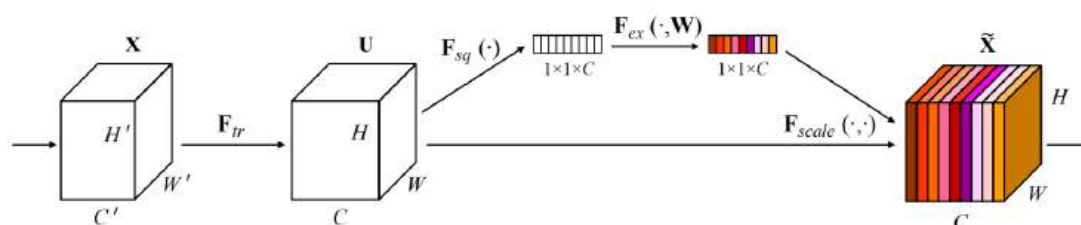


Рис. 8. Архитектура блока *squeeze&excitation*. Цитируется по [25]

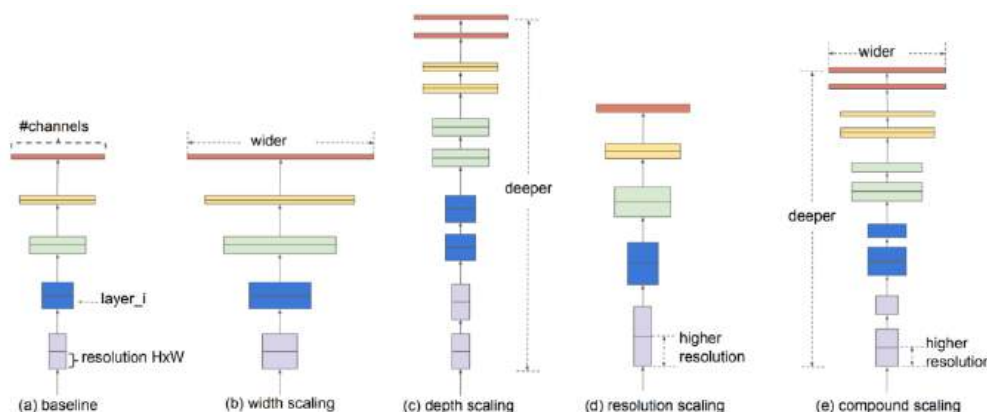


Рис. 10. Результативность различных вариантов архитектур *EfficientNet*. Цитируется по [27]

В сравнении с другими свёрточными сетями, *EfficientNet* качественно выигрывает как в точности, так и в быстродействии. В частности, *EfficientNet-B7* обеспечивает наивысшую точность в 84,3 % в категории топ-1 изображений, но при этом в 8,4 раза меньше и в 6,1 раза быстрее, чем следующая за ним по результативности *GPipe* (сверточная нейронная сеть, основанная на принципе гигантской сверточной сети [28]). *EfficientNet-B1* в 7,6 раза меньше и в 5,7 раза быстрее, чем *ResNet-152* [26].

Обучение архитектуры *EfficientNet* могло занимать довольно много времени, поэтому в 2021 г. разработчики решили эту проблему, оптимизировав время обучения сети (уменьшая размер сети и незначительно теряя в точности), и представили новую версию архитектуры *EfficientNetV2*. На начало 2023 г. данная архитектура является наиболее эффективной, занимает лидирующие позиции по точности на некоторых датасетах в задаче идентификации картинок [29].

Рассмотренные архитектуры являются работками с открытым исходным кодом, поэтому у разработчиков есть возможность не только использовать готовые реализации этих архитектур в библиотеках *PyTorch*, *TensorFlow* и других, но и изменять исходную архитектуру под свои задачи.

В последнее время в компьютерном зрении становятся популярны архитектуры-трансформеры [30]. Они показывают высочайшую точность, но имеют свои минусы, такие как огромный набор качественно размеченных входных данных для эффективного обучения модели и значительные вычислительные ресурсы. Трансформеры требуют выполнения большого количества операций с высокой сложностью, таких как множественные внутренние произведения и преобразования Фурье.

### Сравнительный анализ сверточных нейронных сетей

В таблице автор данной статьи предлагает сравнительную характеристику некоторых архитектур сверточных нейронных сетей. Архитектуры-трансформеры реализуют подход, основанный на создании визуальных токенов, поэтому они далее не рассматриваются. Оценить быстродействие приведенных архитектур можно только относительно друг друга, так как время обучения модели и вывода конечной информации зависит от вычислительной мощности оборудования, на котором будут проводить измерения. Приблизительно оценить быстродействие можно по количеству обучаемых параметров в рассматриваемой архитектуре (чем меньше, тем быстрее).

Основные особенности, преимущества и недостатки различных архитектур сверточных нейронных сетей

Архитектура	Особенности	Преимущества	Недостатки
<i>AlexNet</i>	Глубокая сверточная нейронная сеть (количество слоев 12). Использование сверток 11×11, 5×5, 3×3. Использование функции активации <i>ReLU</i> . 62,3 млн обучаемых параметров. Использование слоев <i>max pooling</i> и <i>dropout</i>	Лучшая точность на конкурсе в 2012 г. Относительно небольшое количество параметров (весов). Отсутствие вычислительно-трудоемких операций с экспонентой благодаря использованию <i>ReLU</i>	Медленнее современных архитектур. Проблема затухающего градиента присутствует
<i>VGG</i>	Использование слоев <i>max pooling</i> и функции активации <i>ReLU</i> . Идея повторяющихся блоков. Глубокие сети (до 19 слоев). 138 млн обучаемых параметров	Высокая точность на <i>ImageNet</i> . Меньшее количество параметров в каждом слое в сравнении с <i>AlexNet</i> . Простота и понятность архитектуры	Очень высокие вычислительные затраты. Проблема затухающего градиента присутствует
<i>Inception</i>	Использование блока, состоящего из нескольких параллельных сверточных слоев различных размеров. 5 млн обучаемых параметров. 27 слоев (для версии v.1). 23 млн обучаемых параметров. 48 слоев (для версии v.3)	Высокая точность на <i>ImageNet</i> . Уменьшение вычислительной сложности. Проблема затухающего градиента при помощи дополнительных выходных слоев решена	Незначительное усложнение архитектуры. Значительные вычислительные затраты
<i>ResNet</i>	Использование <i>skip-connections</i> для более эффективного прохождения градиента по обратной связи в глубоких сетях. Различные варианты кол-ва слоев (от 18 до 152). Количество обучаемых параметров варьируется от 13,6 млн ( <i>ResNet-34</i> ) до 63,5 млн ( <i>ResNet-152</i> )	Высокая точность на <i>ImageNet</i> . Построение глубоких сетей без затухания градиента. Быстродействие зависит от глубины рассматриваемой версии архитектуры	Для глубоких вариантов <i>ResNet</i> существует проблема увеличенного количества параметров (весов)
<i>SENet</i>	Использование <i>SE</i> блока, позволяющего перевзвешивать каналы относительно друг друга и выделять наиболее значимые признаки. 28,1 млн обучаемых параметров. 19 слоев	Высокая точность на <i>ImageNet</i> . Улучшение качества сети за счёт добавления <i>SE</i> -блока. <i>SE</i> -блок может внести дополнительную нелинейность	Большое количество обучаемых параметров. Проблема затухающего градиента присутствует
<i>EfficientNet</i>	Оптимизация трех параметров (глубины, ширины сети и размера изображения) с помощью метода шкалирования. Количество слоев варьируется в процессе подбора оптимальных параметров. Количество обучаемых параметров варьируется от 5 млн ( <i>EfficientNet-B0</i> ) до 66 млн ( <i>EfficientNet-B7</i> )	Высочайшая точность на <i>ImageNet</i> при меньшем количестве параметров, чем у других архитектур сверточных сетей. Проблема затухающего градиента решена при помощи выборочных соединений и <i>SE</i> -блоков	Является менее изученной, чем другие архитектуры. По точности уступает трансформерам. Старшие (глубокие и широкие) модели вычислительно-сложные



Исходя из проведенного анализа, можно сделать вывод, что для последующего применения компьютерного зрения в производственных системах необходимо ориентироваться на особенности приведенных сетей, а также учитывать доступные вычислительные мощности и требования к точности системы.

Сравнение архитектур сверточных нейронных сетей проводили с учетом следующих важнейших параметров: точность модели на датасете *ImageNet* в задаче классификации; вычислительная сложность модели, которая напрямую зависит от глубины, ширины, функций активаций и других архитектурных особенностей выбранной нейронной сети; быстродействие нейронной сети, которое напрямую связано с вычислительной сложностью модели, так как чем сложнее модель, тем большее количество преобразований и вычислений будет применено к входному изображению для получения конечного результата.

Разработчикам систем компьютерного зрения следует понимать, что не существует классификации архитектур сверточных нейронных сетей по критерию конкретных решаемых производственных задач. В процессе выбора архитектуры следует опираться на изложенные особенности и параметры, определять приоритеты и желаемые результаты, которые подходят в рамках допустимой точности и располагаемого вычислительного оборудования. Например, если необходима наибольшая точность и вопросы быстродействия не критичны, то стоит попробовать использовать модификацию нейронной сети *EfficientNet-B7*. Однако она будет неэффективна для задачи обработки видеопотока в настоящем времени.

Если возможно пожертвовать точностью модели в угоду быстродействию, то можно выбрать *EfficientNet-B1* или *AlexNet*, быстродействие которых позволит обрабатывать видео с большим количеством кадров в секунду.

При выборе в пользу той или иной архитектуры сверточных нейронных сетей следует также учитывать ряд критериев и ограничений, таких как:

- наличие достаточного объема качественно размеченных данных в случае дообучения модели;
- будет ли использована нейронная сеть под конкретную задачу или ее обучение необходимо проводить с нуля;
- допустимые временные затраты на обучение сверточной нейронной сети и на подбор оптимальных гиперпараметров модели в соответствии с выполняемой задачей;
- наличие современного вычислительного оборудования для обучения и использования новейших моделей сверточных нейронных сетей.

## Заключение

Основными целями применения систем компьютерного зрения в промышленности как в российских, так и в иностранных компаниях является решение задач оптимизации производственных процессов, таких как контроль точности выполнения технологического процесса, позиционирование манипуляторов, инструментов обработки и т. п. Эти задачи можно эффективно выполнять при помощи программно-аппаратной реализации технологий CV.

В основе компьютерного зрения лежит сверточная нейронная сеть, распознающая различные образы. Создатели архитектур сверточных нейронных сетей внесли значительный вклад в развитие технологий CV. В каждой архитектуре реализуется свой особый подход для устранения проблемы затухающего градиента, повышения точности и уменьшения вычислительной сложности обработки информации. Наблюдается тенденция создания комбинированных систем компьютерного зрения, использующих лучшие практики в архитектурах сверточных нейронных сетей, что позволяет добиваться заданных показателей точности и быстродействия без затухания градиента.

Сравнительный анализ популярных архитектур сверточных нейронных сетей показал, что на вопрос, какую архитектуру предпочтительно использовать в промышленном производстве, нет однозначного ответа. Разработчик системы CV может выбрать архитектуру с учетом имеющихся на производстве вычислительных мощностей и требований к конечной точности модели. Для наиболее точных архитектур, которые, как правило, являются очень глубокими, необходимо с нуля собирать огромное количество данных для качественного обучения или дообучения системы компьютерного зрения.

Таким образом, в процессе разработки систем CV необходимо учитывать наличие: достаточного объема данных и времени для обучения нейронной сети (обучение наиболее глубоких моделей даже на самом передовом вычислительном оборудовании может занимать несколько дней); специалистов, способных поддерживать систему CV, дополнять и расширять функционал нейронной сети, дообучать ее на данных, описывающих новый технологический процесс; средств для поддержки вычислительных и сенсорных аппаратных частей системы компьютерного зрения.

Перспективным решением в построении моделей компьютерного зрения являются трансформеры, которые демонстрируют большую точность, чем любая из рассмотренных в данной статье архитектур сверточных нейронных сетей, но требуют специализированного мощного вычислительного оборудования и исключительно огромного набора размеченных изображений для обучения.

## Литература

1. How Are Robots Used in Car Manufacturing? [Электронный ресурс]. URL: <https://www.roboticscareer.org/news-and-events/news/23035> (дата обращения: 08.04.2023).
2. Let's talk logistics and warehouse automation: how mobile robots improve supply chain efficiencies & solve labor challenges [Электронный ресурс]. URL: <https://www.mobile-industrial-robots.com/the-mir-blog/let-s-talk-logistics-and-warehouse-automation-how-mobile-robots-improve-supply-chain-efficiencies-solve-labor-challenges/> (дата обращения: 08.04.2023).
3. Using 3D Vision for Assembly Robots in Automotive Manufacturing Automation [Электронный ресурс]. URL: <https://www.mech-mind.com/blog/3d-vision-assembly-robots-automotive-manufacturing-automation.html> (дата обращения: 08.04.2023).
4. Роботы-сварщики на смену человеку [Электронный ресурс]. URL: <https://vektorus.ru/blog/robotizirovannaya-svarka.html> (дата обращения: 08.04.2023).
5. Плазменная резка [Электронный ресурс]. URL: <https://www.robotomatic.ru/robots/plazmennaya-rezka> (дата обращения: 08.04.2023).
6. Sathiyamoorthy S. Industrial Application of Machine Vision [Электронный ресурс]. URL: [https://www.researchgate.net/publication/271509417 INDUSTRIAL\\_APPLICATION\\_OF\\_MACHINE\\_VISION](https://www.researchgate.net/publication/271509417 INDUSTRIAL_APPLICATION_OF_MACHINE_VISION) (дата обращения: 08.04.2023).
7. Савченко А. В. Глаз-алмаз смотрит в цель [Электронный ресурс]. URL: <https://nnov.hse.ru/ma/vision/news/559218902.html> (дата обращения: 08.04.2023).
8. Захаров В. С. Технологии компьютерного зрения на российском и мировом рынках и их перспективы // Вестник Таганрогского института управления и экономики. 2022. № 1(35). С. 114—115.
9. Гончаренко Д. О. Развитие технологий компьютерного зрения в России на примере компании Dioram: мат. IX Ежегодной Всерос. науч.-практ. конф. "Цифровизация общества: состояние, проблемы, перспективы". Москва, 07 июня 2022 года. Т. 1. — М: ФГБОУ ВО "РЭУ им. Г. В. Плеханова", 2022. С. 186—190.
10. Калмацкий М. В России бурными темпами растет рынок компьютерного зрения [Электронный ресурс]. URL: <https://rg.ru/2022/01/19/v-rossii-burnymi-tempami-rastet-rynok-kompiuternogo-zreniia.html> (дата обращения: 06.04.2023).
11. Gandhi R. R-CNN, Fast R-CNN, Faster R-CNN & YOLO — Object Detection Algorithms [Электронный ресурс]. URL: <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e> (дата обращения: 08.04.2023).
12. Костерин В. В. Как выполнить обнаружение объектов YOLO с помощью OpenCV и PyTorch в Python [Электронный ресурс]. URL: <https://waksoft.susu.ru/2021/05/19/kak-vypolnit-obnaruzhenie-obektov-yolo-s-pomoshhyu-opencv-i-pytorch-v-python/> (дата обращения: 08.04.2023).
13. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation [Электронный ресурс]. URL: <https://arxiv.org/abs/1505.04597/> (дата обращения: 08.04.2023).
14. Kirillov A. et al. Panoptic feature pyramid networks: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019. P. 6399—6408.
15. MidJourney Documentation [Электронный ресурс]. URL: <https://docs.midjourney.com/docs/models> (дата обращения: 08.04.2023).
16. LeCun Y., Boser B., Denker J. S., Henderson D., Howard R. E., Hubbard W., Jackel L. D. Backpropagation Applied to Handwritten Zip Code Recognition // Neural Computation. 1989. V. 1(4). P. 541—551.
17. AlexNet and ImageNet: The Birth of Deep Learning [Электронный ресурс]. URL: <https://www.pinecone.io/learn/imagenet/#:~:text=When%20the%20paper%20detailing%20ImageNet,up%20into%20evermore%20general%20categories> (дата обращения: 04.04.2023).
18. ImageNet [Электронный ресурс]. URL: <https://www.image-net.org/> (дата обращения: 04.04.2023).
19. Pujara A. Concept of AlexNet Convolutional Neural Network [Электронный ресурс]. URL: <https://medium.com/analytics-vidhya/concept-of-alexnet-convolutional-neural-network-6e73b4f9ee30> (дата обращения: 06.04.2023).
20. Boesch G. VGG (Very Deep Convolutional Networks) [Электронный ресурс]. URL: <https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/> (дата обращения: 06.04.2023).
21. Balov B. Глубокое обучение: разбираемся со свертками [Электронный ресурс]. URL: <https://medium.com/@balovbohdan/глубокое-обучение-разбираемся-со-свертками-6e47bfc27792/> (дата обращения: 06.04.2023).
22. Raj B. A Simple Guide to the Versions of the Inception Network [Электронный ресурс]. URL: <https://towardsdatascience.com/a-simple-guide-to-the-versions-of-the-inception-network-7fc52b863202> (дата обращения: 06.04.2023).
23. Liu Xiaoyu, Diyu Yang, Aly El Gamal. Deep neural network architectures for modulation classification [Электронный ресурс]. URL: [arXiv:1712.00443](https://arxiv.org/abs/1712.00443) (дата обращения: 06.04.2023).
24. Литвинов С. ResNet (34, 50, 101) [Электронный ресурс]. URL: <https://neurohive.io/ru/vidy-nejrosetej/resnet-34-50-101/> (дата обращения: 06.04.2023).
25. Hu Jie, Li Shen, Gang Sun. Squeeze-and-Excitation Networks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2017): 7132–7141 [Электронный ресурс]. URL: [arXiv:1709.01507](https://arxiv.org/abs/1709.01507) (дата обращения: 06.04.2023).
26. Tan Mingxing, Quoc V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. (2019) [Электронный ресурс]. URL: [arXiv:1905.11946](https://arxiv.org/abs/1905.11946) (дата обращения: 06.04.2023).
27. EfficientNet (paperswithcode.com) [Электронный ресурс]. URL: <https://paperswithcode.com/method/efficientnet> (дата обращения: 08.04.2023).
28. Huang Yanping, Yonglong Cheng, Dehao Chen, HyoukJoong Lee, Jiquan Ngiam, Quoc V. Le, Chen Z. GPipe: Efficient Training of Giant Neural Networks using Pipeline Parallelism [Электронный ресурс]. URL: [arXiv:1811.06965](https://arxiv.org/abs/1811.06965) (дата обращения: 08.04.2023).
29. Papers with Code: Image Classification [Электронный ресурс]. URL: <https://paperswithcode.com/task/image-classification> (дата обращения: 08.04.2023).
30. Vision Transformers [Электронный ресурс]. URL: <https://blog.xperience.ai/vision-transformers/> (дата обращения: 08.04.2023).

---

## USING CONVOLUTIONAL NEURAL NETWORKS TO SOLVE IMAGE RECOGNITION PROBLEMS IN INDUSTRY

R. R. Zhalyalov

Bauman Moscow State Technical University, Moscow, Russia

*The article provides examples of the use of computer vision technologies in Russian and foreign companies. The actual problems of industry related to the use of computer vision are considered. Various variants of computer vision implementation based on convolutional neural networks are described. An overview of popular convolutional neural network architectures is given and their comparative analysis is carried out.*

**Keywords:** computer vision, convolutional neural networks, object recognition, identification, classification, vanishing gradient, ImageNet, AlexNet, VGG, Inception, ResNet, SENet, EfficientNet

---

**Жалялов Руслан Ринатович**, магистрант.  
E-mail: zhalyalov.rr@mail.ru

*Статья поступила в редакцию 16 мая 2023 г.*

УДК 004.054

DOI: 10.52190/2073-2597\_2023\_2\_32

EDN: JLHNJX

## ПРИМЕНЕНИЕ ПРОГРАММНЫХ СРЕДСТВ ДЛЯ АВТОМАТИЗАЦИИ ТЕСТИРОВАНИЯ БЕЗОПАСНОСТИ ОБЪЕКТОВ КОНТЕЙНЕРИЗАЦИИ

П. Д. Белов

ООО «Локус Софт Текнолоджис», Москва, Россия

А. О. Махмадзиёев

ООО «РусБИТех-Астра», Москва, Россия

А. А. Павленко

ООО «СРМ Солюшнс», Москва, Россия

К. В. Пителинский, канд. техн. наук

Московский политехнический университет, Москва, Россия

*Обсуждаются ограничения традиционных подходов к тестированию безопасности объектов контейнеризации и преимущества использования средств автоматизации тестирования. Рассматривается применение программной среды Kubernetes для автоматизированного тестирования объектов контейнеризации. Приводятся практические примеры использования инструментов Kube-bench, Kube-hunter, Kubesploit, Trivy для поиска уязвимостей. Подчеркивается положительное влияние автоматизации тестирования на обеспечение качества контейнерных приложений.*

**Ключевые слова:** контейнеризация, автоматизированное тестирование, безопасная разработка, информационная безопасность, надежность ПО.

Контейнеризация изменила индустрию программного обеспечения, обеспечив беспрецедентные уровни гибкости, масштабируемости и эффективности развертывания приложений [1]. Она позволяет создавать изолированные контейнеры, которые инкапсулируют приложения и их зависимости, упрощая процессы разработки, развертывания и управления программным обеспечением. Однако с ростом сложности и динамичности контейнеризации становится необходимым всестороннее тестирование объектов контейнеризации для обеспечения надежности, стабильности и безопасности соответствующих объектов инфраструктуры.

Существует ряд способов тестирования объектов контейнеризации — функциональное тестирование, тестирование производительности, нагрузочное тестирование, тестирование на совместимость, тестирование безопасности.

Функциональное тестирование — вид тестирования, направленный на проверку соответствия объекта контейнеризации (приложения или сервиса) ожидаемым функциональным требованиям. Такой вид включает в себя тестирование поведения ввода/вывода и пользовательских и программных интерфейсов.

Тестирование производительности направлено на оценку производительности и масштабируемости контейнеров, подов (*pod* — мельчайший объект структуры, который может состоять из одного или нескольких контейнеров) или кластеров при различных рабочих нагрузках. Оно позволяет выявить потенциально узкие места, ограничения ресурсов и оптимизировать систему.

Нагрузочное тестирование позволяет понять поведение контейнеров, подов или кластеров при предполагаемых производственных нагрузках, а также помогает определить пороговые значения производительности контейнеров.

Тестирование на совместимость — вид тестирования, направленный на оценку совместимости между объектами контейнеризации и средой их выполнения, библиотеками и зависимостями. С его помощью выявляют любые конфликты или несовместимости, которые могут возникнуть.

Тестирование безопасности направлено на оценку уровня безопасности контейнеров и контейнерных приложений. Такой вид тестирования включает в себя сканирование уязвимостей, тестирование на проникновение, проверку контроля доступа и методов укрепления контейнеров.

Авторы данной работы акцентируют внимание на тестировании безопасности, так как его роль для развития отечественной отрасли информационной безопасности очень важна в настоящее время.

Тестирование безопасности объектов контейнеризации представляет собой сложную задачу. Динамическая природа сред, где контейнеры могут предоставляться, перемещаться и удаляться по требованию, создает трудности для ручных процессов тестирования. Более того, сложные взаимосвязи между отдельными компонентами сред, а также их взаимодействие с внешними системами затрудняют обеспечение тщательного тестового покрытия и выявление

потенциальных проблем. Для преодоления этих трудностей ключевую роль играет использование программного обеспечения (ПО) при тестировании объектов контейнеризации. Благодаря автоматизации обеспечивается системный подход и повторяемость процесса тестирования, что позволяет улучшать покрытие тестами и ускорять цикл обратной связи. В результате организации могут оптимизировать процессы тестирования объектов контейнеризации и добиться более надежных и эффективных результатов.

#### Автоматизация тестирования в среде *Kubernetes*: примеры применения

Из-за санкций, которым подверглись российские компании, возникла потребность обеспечения эффективной оркестровки (координации и масштабирования в условиях кластера, автоматизации развёртывания и др.) контейнерных приложений. Под действием секторальных санкций, наложенных на РФ, с ее рынка программного обеспечения уходят традиционные поставщики программных и аппаратных решений, что требует поиска альтернатив. Одним из альтернативных решений является *Kubernetes*, которое поддерживает технологии аппаратной виртуализации и базовые технологии контейнеризации.

*Kubernetes* — среда оркестровки контейнеров с открытым исходным кодом, разработанная Google и развиваемая компаниями Amazon, Microsoft, IBM и другими [2]. Данная среда активно применяется на Западе. Согласно исследованию, проведённому *Positive Technologies*, а также приведенной ими тепловой карте, в России технику поиска информации в контейнерах применяли менее, чем в 20 % организаций среди 30 исследованных заказчиков, представивших 53 проекта [3]. Однако на фоне активного продвижения облачных сервисов от VK, Yandex и МТС использование *Kubernetes* на российском рынке получает все большее распространение.

Ключевые функции среды *Kubernetes*:

- автоматизированное развёртывание и масштабирование контейнеров;
- самовосстановление контейнеров и узлов;
- балансировка нагрузки и обнаружение сервисов.

*Kubernetes* позиционируется как безопасный инструмент, и, следовательно, он регулярно обновляется без необходимости кардинально изменять уже выстроенную инфраструктуру. Однако, как и любая новая технология, стремительно охватывающая рынок информационных услуг, этот инструмент имеет недо-

статки и уязвимости. По данным *Red Hat*, за 2022 г. сотни опрошенных инженеров хотя бы раз сталкивались с инцидентами информационной безопасности в своих кластерах *Kubernetes*, а многим из них и вовсе приходилось замедлять или останавливать работу кластера [4]. Поэтому возникает целесообразность проведения тестирования безопасности инфраструктур, где применяется *Kubernetes*.

Авторы под объектами контейнеризации понимают контейнеры и абстракции, рассматриваемые в процессе использования *Kubernetes* — кластеры, ноды, сервисы и поды. Кластер — это непосредственно компьютер или множество взаимосвязанных компьютеров, на которых установлен *Kubernetes* (чаще всего физический сервер или виртуальный стенд). Нода (*node*) — узел, отвечающий за запуск контейнерных приложений. В одном кластере может быть минимум один "главный" узел (*master node*), а дополнительные узлы (дополнительные компьютеры, на которые устанавливают *Kubernetes*), подключаемые к главному узлу извне, становятся "рабочими" (*worker node*). Сервис — абстракция, определяющая набор подов и политику доступа к ним.

Для автоматизированного тестирования *Kubernetes* применяют инструменты *Kube-bench*, *Kube-hunter*, *Kubesploit*, *Trivy*, каждый из которых имеет как свою область воздействия на инфраструктуру (конфигурации, кластер в целом, сервисы), так и свой тип взаимодействия со структурой (сканирование конфигураций, сканирование подов и их зависимостей, автоматическую проверку наличия уязвимостей и попыток их эксплуатации). В данной работе подробно описаны возможности каждого инструмента, для демонстрации которых в качестве объекта тестирования выбран виртуальный стенд с установленной платформой *Deckhouse*, разработанной отечественной компанией АО "Флант" и входящей в реестр российского ПО. *Deckhouse* позволяет автоматически развёртывать кластеры *Kubernetes* с необходимыми дополнительными средствами мониторинга в любой инфраструктуре. Эта платформа включает в себя контроллер для пользовательского трафика, аутентификацию на базе любых внешних провайдеров и веб-панель для администрирования [5].

*Kube-bench* — автоматизированный сканнер конфигураций *Kubernetes*, разработанный компанией *Aqua Security* и представленный в виде отдельного пода, интегрируемого в инфраструктуру и записывающего информацию о прохождении проверок в своих логах [6]. На рис. 1 представлен пример выходных данных в виде логов подключаемого пода *Kube-bench*.

```

[INFO] 1 Master Node Security Configuration
[INFO] 1.1 API Server
[FAIL] 1.1.1 Ensure that the --allow-privileged argument is set to false (Scored)
[FAIL] 1.1.2 Ensure that the --anonymous-auth argument is set to false (Scored)
[PASS] 1.1.3 Ensure that the --basic-auth-file argument is not set (Scored)
[PASS] 1.1.4 Ensure that the --insecure-allow-any-token argument is not set (Scored)
[FAIL] 1.1.5 Ensure that the --kubelet-https argument is set to true (Scored)
[PASS] 1.1.6 Ensure that the --insecure-bind-address argument is not set (Scored)
[PASS] 1.1.7 Ensure that the --insecure-port argument is set to 0 (Scored)
[PASS] 1.1.8 Ensure that the --secure-port argument is not set to 0 (Scored)
[FAIL] 1.1.9 Ensure that the --profiling argument is set to false (Scored)
[FAIL] 1.1.10 Ensure that the --repair-malformed-updates argument is set to false (Scored)
[PASS] 1.1.11 Ensure that the admission control policy is not set to AlwaysAdmit (Scored)
[FAIL] 1.1.12 Ensure that the admission control policy is set to AlwaysPullImages (Scored)
[FAIL] 1.1.13 Ensure that the admission control policy is set to DenyEscalatingExec (Scored)
[FAIL] 1.1.14 Ensure that the admission control policy is set to SecurityContextDeny (Scored)
[PASS] 1.1.15 Ensure that the admission control policy is set to NamespaceLifecycle (Scored)
[FAIL] 1.1.16 Ensure that the --audit-log-path argument is set as appropriate (Scored)
[FAIL] 1.1.17 Ensure that the --audit-log-maxage argument is set to 30 or as appropriate (Scored)
[FAIL] 1.1.18 Ensure that the --audit-log-maxbackup argument is set to 10 or as appropriate (Scored)
[FAIL] 1.1.19 Ensure that the --audit-log-maxsize argument is set to 100 or as appropriate (Scored)
[PASS] 1.1.20 Ensure that the --authorization-mode argument is not set to AlwaysAllow (Scored)
[PASS] 1.1.21 Ensure that the --token-auth-file parameter is not set (Scored)
[FAIL] 1.1.22 Ensure that the --kubelet-certificate-authority argument is set as appropriate (Scored)

```

Рис. 1. Пример вывода логов пода Kube-bench [7]

Авторы данной статьи протестировали при помощи *Kube-bench* кластер *Kubernetes* на стенде с установленной платформой *Deckhouse*. Результаты тестирования впечатляют — из 125 проверок лишь 3 остались не пройденными (рис. 2).

```

== Summary policies ==
0 checks PASS
0 checks FAIL
30 checks WARN
0 checks INFO

== Summary total ==
83 checks PASS
3 checks FAIL
39 checks WARN
0 checks INFO

```

Рис. 2. Вывод общих результатов проверок с помощью Kube-bench

*Kube-hunter* — автоматизированный сканнер кластеров *Kubernetes*, также разработанный компанией *Aqua Security*, но представленный уже в виде консольной утилиты, для выполнения которой указывается атакуемый кластер. Вывод уязвимостей производится в виде таблицы непосредственно на консоли после завершения работы утилиты [6, 8]. Пример выходных данных представлен на рис. 3.

При тестировании авторами данной статьи кластера *Kubernetes* с помощью *Kube-hunter* на стенде с *Deckhouse* были обнаружены лишь две потенциальные уязвимости, связанные непосредственно с установленным кластером, причем уровень критичности одной из них очевидно не выше, чем "низкий". На рис. 4 представлен фрагмент выходных данных, соответствующий проверке виртуального стенда с *Deckhouse*.

Vulnerabilities						
For further information about a vulnerability, search its ID in: <a href="https://avd.aquasec.com/">https://avd.aquasec.com/</a>						
ID	LOCATION	CATEGORY	VULNERABILITY	DESCRIPTION	EVIDENCE	
KHV034	192.168.199.211:2379	Unauthenticated Access	Etcd is accessible using insecure connection (HTTP)	Etcd is accessible using HTTP (without authorization and authentication), it would allow a potential attacker to gain access to the etcd	b'{"etcdserver":"3.4.13","etcdcluster":"3.4.0"}'	
KHV033	192.168.199.211:2379	Information Disclosure	Etcd Remote version disclosure	Remote version disclosure might give an attacker a valuable data to attack a cluster	b'{"etcdserver":"3.4.13","etcdcluster":"3.4.0"}'	
KHV002	192.168.199.211:6443	Information Disclosure	K8s Version Disclosure	The kubernetes version could be obtained from the /version endpoint	v1.19.10	

Рис. 3. Пример вывода уязвимостей, выявленных Kube-hunter [8]



Vulnerabilities						
For further information about a vulnerability, search its ID in: <a href="https://avd.aquasec.com/">https://avd.aquasec.com/</a>						
ID	LOCATION	MITRE CATEGORY	VULNERABILITY	DESCRIPTION	EVIDENCE	
None	Local to Pod (kube-hunter-vtlf)	Lateral Movement // ARP poisoning and IP spoofing	CAP_NET_RAW Enabled	CAP_NET_RAW is enabled by default for pods. If an attacker manages to compromise a pod, they could potentially take advantage of this capability to perform network attacks on other pods running on the same node		
KHV002	10.222.0.1:443	Initial Access // Exposed sensitive interfaces	K8s Version Disclosure	The kubernetes version could be obtained from the /version endpoint	v1.23.16	
KHV005	10.222.0.1:443	Discovery // Access the K8S API Server	Access to API using service account token	The API Server port is accessible. Depending on your RBAC settings this could expose access to or control of your cluster.	b'{"kind":"APIVersion", "versions":["v1"], "serverAddressByClientCIDRs":[{"clientCIDR":"0.0.0.0/0", "s...	

Рис. 4. Уязвимости, выявленные Kube-hunter в инфраструктуре с платформой Deckhouse

*Kubesploit* — автоматизированное средство эксплуатации уязвимостей кластеров *Kubernetes*, разработанное компанией *CyberArk* [9]. Принцип использования этого средства аналогичен широко известному универсальному средству эксплуатации уязвимостей *Metasploit*, встроенному в дистрибутив *Kali OS Linux*. Кроме того, в *Kubesploit* для выполнения тестов используют точно такой же набор команд и синтаксис, как в *Metasploit* [10]. На рис. 5 указан пример перечня уязвимостей, выявляемых *Kubesploit*.

Наиболее интересным, с точки зрения авторов статьи, оказался модуль *clusterCVEScan*,

реализующий сканирование кластера на наличие известных уязвимостей из открытого реестра. Важно отметить, что для работы этого модуля необходимо иметь доступ к внутренней сети, где расположен кластер, а также знать адрес и порт непосредственно кластера *Kubernetes* (порт определяется автоматически в процессе построения инфраструктуры, включающей в себя *Kubernetes*).

Процесс выявления порта кластера *Kubernetes* на платформе *Deckhouse* при помощи обращения к командной строке изображен на рис. 6.

```
[*] Results for 8c241139-4dc4-496b-9a5e-95f1a388797a job QaiqYQIMIXS at 2020-11-12T10:36:28Z
[*] Scanning Kubernetes cluster: https://ks.amazonaws.com/version
[*] Current cluster version: v1.14.9-eks-4
[*] ID: CVE-2019-11254
[*] Description: The Kubernetes API Server component in versions 1.1-1.14, and versions prior to 1.15.10, 1.16.7 and 1.17.3 allows an authorized user who sends malicious YAML payloads to cause the kube-apiserver to consume excessive CPU cycles while parsing YAML.
[*] Fixed versions: 1.15.10, 1.16.7, 1.17.3
[*] ID: CVE-2020-8558
[*] Description: The kubelet and kube-proxy were found to contain security issue which allows adjacent hosts to reach TCP and UDP services bound to 127.0.0.1 running on the node or in the node's network namespace. Such a service is generally thought to be reachable only by other processes on the same host, but due to this defect, could be reachable by other hosts on the same LAN as the node, or by containers running on the same node as the service.
[*] Fixed versions: 1.16.11, 1.17.7, 1.18.4, 1.16.2
[*] ID: CVE-2020-8559
[*] Description: The Kubernetes kube-apiserver is vulnerable to an unvalidated redirect on proxied upgrade requests that could allow an attacker to escalate privileges from a node compromise to a full cluster compromise.
[*] Fixed versions: 1.16.13, 1.17.9, 1.18.6
[*] Done
```

Рис. 5. Пример вывода уязвимостей, выявленных *Kubesploit* [10]

```
root@deckhouse:/home/deckhouse# kubectl cluster-info
Kubernetes control plane is running at https://127.0.0.1:6445
```

Рис. 6. Выявленный порт кластера *Kubernetes*



Однако при попытке сканирования кластер запретил соединение, из-за чего проведение сканирования стало невозможным (рис. 7).

*Trivy* — сканер уязвимостей с открытым исходным кодом, предназначенный для сканирования образов контейнеров или виртуальных машин, кластеров *Kubernetes*, удаленных репозиторий *Git* и элементов файловой системы на наличие уязвимостей, разработанный компанией *Aqua Security* и доступный под лицензией *Apache 2.0* [11]. Одним из ключевых преимуществ *Trivy* является простота использования. Его можно интегрировать в существующие конвейеры непрерывной интеграции и разработки и использовать для автоматизации процесса сканирования уязвимостей. Этот инструмент поддерживает несколько выходных форматов,

включая *JSON*, *YAML* и *Junit XML*, что облегчает интеграцию с другими инструментами. Еще одним преимуществом *Trivy* является его быстродействие. Он может сканировать образы контейнеров за считанные секунды, что делает его идеальным инструментом для быстрого сканирования больших хранилищ образов. Процесс сканирования оптимизирован для экономии ресурсов, что делает его пригодным для использования в средах с ограниченными ресурсами. Используется база данных уязвимостей, пополняемая из разных источников (*NVD*, *Red Hat Security Data* и *Debian Security Tracker*).

На рис. 8 приведен фрагмент перечня уязвимостей кластера *Kubernetes* на стенде с платформой *Deckhouse*, выявленных с помощью *Trivy*.

```
kubesploit» use module linux/go/clusterCVEScan
kubesploit(module)[K8sClusterCVEScan]» set url https://
[+] New authenticated agent checkin for ed3e7eed-95b9-4b3f-96e7-5ab4721dac6b from [REDACTED]:51304 at
2023-04-17T12:55:42Z
kubesploit(module)[K8sClusterCVEScan]» set url https://[REDACTED]:6445
kubesploit(module)[K8sClusterCVEScan]»
[+] url set to https://[REDACTED]:6445
kubesploit(module)[K8sClusterCVEScan]» run
kubesploit(module)[K8sClusterCVEScan]»
[-] Created job FGaSutgomT for agent ed3e7eed-95b9-4b3f-96e7-5ab4721dac6b at 2023-04-17T12:55:59Z

[+] Results for ed3e7eed-95b9-4b3f-96e7-5ab4721dac6b job FGaSutgomT at 2023-04-17T12:56:10Z

[*] Scanning Kubernetes cluster: https://[REDACTED]:6445/version
Failed with error: %s Get "https://[REDACTED]:6445/version": dial tcp [REDACTED]:6445: connect: connection refused
[*] Done
```

Рис. 7. Попытка атаки на кластер *Kubernetes*, выявленная *Kubesploit*

wordpress:4.8-apache (debian 8.9)

=====

Total: 1079 (UNKNOWN: 54, LOW: 160, MEDIUM: 440, HIGH: 336, CRITICAL: 89)

Library	Vulnerability	Severity	Installed Version	Fixed Version	Title	
apache2	CVE-2018-1312	CRITICAL	2.4.10-10+deb8u11	2.4.10-10+deb8u12	httpd: Weak Digest auth nonce generation in mod_auth_digest <a href="https://avd.aquasec.com/nvd/cve-2018-1312">https://avd.aquasec.com/nvd/cve-2018-1312</a>	
	CVE-2017-15710				httpd: Out of bounds write in mod_authnz_ldap when using too small Accept-Language... <a href="https://avd.aquasec.com/nvd/cve-2017-15710">https://avd.aquasec.com/nvd/cve-2017-15710</a>	
	CVE-2017-15715				httpd: <FilesMatch> bypass with a trailing newline in the file name <a href="https://avd.aquasec.com/nvd/cve-2017-15715">https://avd.aquasec.com/nvd/cve-2017-15715</a>	
	CVE-2018-1303				httpd: Out of bounds read in mod_cache_socache can allow a remote attacker... <a href="https://avd.aquasec.com/nvd/cve-2018-1303">https://avd.aquasec.com/nvd/cve-2018-1303</a>	
	CVE-2018-17199				2.4.10-10+deb8u13	httpd: mod_session_cookie does not respect expiry time <a href="https://avd.aquasec.com/nvd/cve-2018-17199">https://avd.aquasec.com/nvd/cve-2018-17199</a>
	CVE-2019-0217				2.4.10-10+deb8u14	httpd: mod_auth_digest: access control bypass due to race condition <a href="https://avd.aquasec.com/nvd/cve-2019-0217">https://avd.aquasec.com/nvd/cve-2019-0217</a>
	CVE-2018-1283	MEDIUM		2.4.10-10+deb8u12	httpd: Improper handling of headers in mod_session can allow a remote user... <a href="https://avd.aquasec.com/nvd/cve-2018-1283">https://avd.aquasec.com/nvd/cve-2018-1283</a>	
	CVE-2018-1301				httpd: Out of bounds access after failure in reading the HTTP request... <a href="https://avd.aquasec.com/nvd/cve-2018-1301">https://avd.aquasec.com/nvd/cve-2018-1301</a>	
	CVE-2019-0220			2.4.10-10+deb8u14	httpd: URL normalization inconsistency <a href="https://avd.aquasec.com/nvd/cve-2019-0220">https://avd.aquasec.com/nvd/cve-2019-0220</a>	
	CVE-2019-10092			2.4.10-10+deb8u15	httpd: limited cross-site scripting in mod_proxy error page <a href="https://avd.aquasec.com/nvd/cve-2019-10092">https://avd.aquasec.com/nvd/cve-2019-10092</a>	
	CVE-2019-10098				httpd: mod_rewrite potential open redirect <a href="https://avd.aquasec.com/nvd/cve-2019-10098">https://avd.aquasec.com/nvd/cve-2019-10098</a>	

Рис. 8. Часть перечня уязвимостей, выявленных *Trivy*

Виртуальный стенд для тестирования безопасности объектов контейнеризации может быть также развернут при помощи платформы *Testo*, разработанной компанией *Testo Lang* и активно используемой отечественными специалистами по контролю качества ПО [12]. С помощью этой платформы можно автоматически развернуть отдельный виртуальный стенд, используемый как имитация компьютера злоумышленника или тестировщика, где эмулируется работа ПО, функционал которых позволяет производить воздействие на инфраструктуру извне (в случае текущего исследования — *Kube-hunter* и *Kubesploit*).

#### Сравнительный анализ результатов использования инструментов автоматизированного тестирования

Исходя из ряда практических соображений, таких как стоимость владения инструментами автоматизированного тестирования, их функциональных характеристик, обученности персонала и др., лицу, принимающему решения (ЛПР) часто приходится осуществлять выбор одного из нескольких возможных инструментальных средств. При этом очевидно, что одновременное использование многих средств тестирования не оправдано по целому ряду процедурных, экономических и организационных причин: увеличивается объем ресурсов, выделяемых на владение и обслуживание ПО; требуется инфраструктурная интеграция разнородных платформ тестирования и разработки ПО; растут финансовые затраты, в том числе на обучение персонала.

Для проведения сравнительного анализа инструментов автоматизации тестирования авторы данной статьи предлагают применить метод анализа иерархий Томаса Саати [13, 14]. Метод Т. Саати применим к различным предметным

областям и имеет преимущество перед другими методами экспертных оценок. Процесс автоматизированного тестирования объектов контейнеризации весьма сложен и допускает лишь субъективные оценки по предложенным ЛПР метрикам качества. При применении метода анализа иерархий можно ограничиться оценками, полученными от одного эксперта, а не от группы специалистов, как, например, в методе Дельфи [15].

Рассмотрим пример использования метода Т. Саати для выбора наиболее предпочтительного инструмента автоматизированного тестирования в среде *Kubernetes*. Для сравнения были выбраны критерии (рис. 9). По мнению авторов данной статьи, ключевыми при проведении автоматизированного тестирования являются следующие четыре критерия:

- покрытие уязвимостей — количество как типов уязвимостей, так и видов в каждом типе;
- объекты анализа — количество типов объектов инфраструктуры, подвергнутых анализу;
- полнота документации — количество полезных примеров и их наглядность, точность рекомендаций по применению;
- опыт ЛПР — число случаев использования инструмента в рамках собственных испытаний.

Авторы сравнивают элементы, относящиеся к одному уровню иерархии. Для построения матриц парных сравнений использована линейная фундаментальная шкала предпочтений (табл. 1), позволяющая количественно оценить предпочтения ЛПР (в данном примере ЛПР является системным аналитиком). Шкала имеет равный шаг оценок, одинаково важных для ЛПР, в отличие, например, от многоинтервальной шкалы Харрингтона. При построении матрицы парных сравнений достаточно задать элементы над главной диагональю матрицы, прочие элементы из-за обратной симметричности матрицы вычисляются по формуле  $a_{ij} = 1/a_{ji}$ .

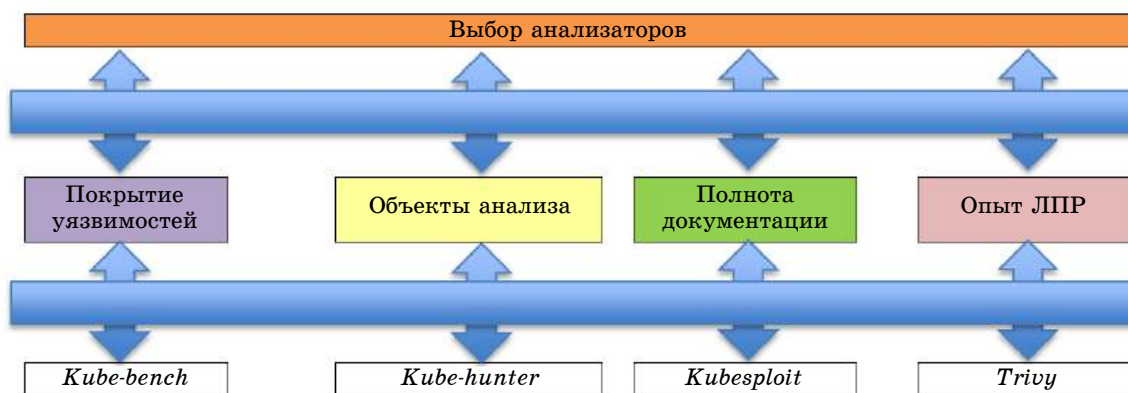


Рис. 9. Иерархия критериев по методу Саати

Таблица 1

## Фундаментальная шкала предпочтений (шкала относительной важности)

$a_{ij}$	Смысл
1	Равная важность сравниваемых элементов иерархии. Для элемента более высокого уровня оба сравниваемых элемента имеют одинаковую значимость
3	Слабое преобладание $i$ -го элемента иерархии над $j$ -м. Прошлый опыт и оценка говорят о чуть большей значимости одного элемента по сравнению с другим
5	Сильное (существенное) преобладание $i$ -го элемента. Прошлый опыт и оценка дают более высокую значимость одного элемента по сравнению с другим
7	Большое преобладание $i$ -го элемента. Очень большая значимость элемента явно выявилась в прошлом
9	Очень большое преобладание $i$ -го элемента, то есть максимально возможное различие между двумя элементами
2, 4, 6, 8	Промежуточные степени преобладания. Значения входят в интервал между приведенными ранее показателями значимости

Сначала была заполнена матрица сравнения критериев, рассчитаны веса критериев и проверена допустимость выставленных оценок по критериям ЛПР.

В ячейки матрицы сравнения критериев (табл. 2) лицо, принимающее решение, проставляет количественные оценки в соответствии с фундаментальной шкалой предпочтений (см. табл. 1).

Затем была составлена матрица весов критериев (табл. 3) и определен вектор весов критериев, то есть рассчитаны средние значения по каждой строке матрицы.

Для оценки допустимости выставленных баллов нужно рассчитать индекс согласованности (ИС) и отношение согласованности (ОС) по формулам:

$$\text{ИС} = \frac{\lambda_{\max} - n}{n - 1}; \quad \text{ОС} = \frac{\text{ИО}}{\text{ПСС}},$$

где  $\lambda_{\max}$  — максимальное собственное значение матрицы  $n$  с баллами;  
ПСС — показатель случайной согласованности, определяемый по табл. 4.

При значении ОС, равным или меньшим 10 %, оценки, сделанные ЛПР, являются допустимыми. Если значение ОС превышает 10 %, нарушается логика предметной области, оценки в матрице (табл. 2) подлежат пересмотру.

В целях вычисления значения  $\lambda_{\max}$  необходимо составить собственные числа для матрицы попарного сравнения критериев (табл. 5).

Таблица 2

## Матрица сравнения критериев

Критерии	Покрытие уязвимостей	Объекты анализа	Полнота документации	Опыт ЛПР
Покрытие уязвимостей	1	3	2	8
Объекты анализа	1/3	1	2	5
Полнота документации	1/2	1/2	1	7
Опыт ЛПР	1/8	1/5	1/7	1

Таблица 3

## Матрица весов критериев

Критерии	Покрытие уязвимостей	Объекты анализа	Полнота документации	Опыт ЛПР	Среднее значение
Покрытие уязвимостей	0,51064	0,63830	0,38889	0,38095	0,47969
Объекты анализа	0,17021	0,21277	0,38889	0,23810	0,25249
Полнота документации	0,25532	0,10638	0,19444	0,33333	0,22237
Опыт ЛПР	0,06383	0,04255	0,02778	0,04762	0,04544

Таблица 4

## Значения ПСС

Размер матрицы	1	2	3	4	5	6	7	8	9	10
ПСС	0	0	0,58	0,9	1,12	1,24	1,32	1,41	1,45	1,49

Таблица 5

## Матрица попарного сравнения критериев

$1 - \lambda$	3	2	8
0,33333	$1 - \lambda$	2	5
0,5	0,5	$1 - \lambda$	7
0,125	0,2	0,14285	$1 - \lambda$

Характеристический полином вышеприведенной матрицы:

$$\lambda^4 - 4\lambda^3 - 3,22023\lambda - 0,45 = 0$$

Его решения, полученные с помощью [16]:

$$\lambda_1 = 0,136678066054879;$$

$$\lambda_2 = 4,18958110132991;$$

$$\lambda_3 = -0,157482248981884 - 0,873601659311384 i;$$

$$\lambda_4 = -0,157482248981884 + 0,873601659311384 i;$$

$$\lambda_{\max} \approx 4,18958.$$

$$\begin{aligned} \text{Тогда ИС} &= \frac{4,18958 - 4}{4 - 1} \approx 0,06319, \text{ а ОС} = \\ &= \frac{0,06319}{0,9} \approx 0,07222 \approx 7,02 \%, \text{ т. е. допусти-} \end{aligned}$$

мость выставленных ЛПР оценок подтверждена.

Аналогичные сравнительные расчеты проводятся и между альтернативами по каждому из критериев. В силу идентичности процесса выставления оценок приведем ниже расчет только для весов и значений  $\lambda_{\max}$ , ИС и ОС (табл. 6—9).

В результате работы с матрицами (табл. 6—9) определяется общая матрица с весами альтернатив по каждому критерию (табл. 10), в которой собраны ранее рассчитанные средние значения.

Таблица 6

Расчет весов по критерию "покрытие уязвимостей"

	<i>Kube-bench</i>	<i>Kube-hunter</i>	<i>Kubesploit</i>	<i>Trivy</i>	Среднее значение
<i>Kube-bench</i>	0,10000	0,12500	0,04255	0,12371	0,09782
<i>Kube-hunter</i>	0,10000	0,12500	0,06383	0,15464	0,11087
<i>Kubesploit</i>	0,30000	0,25000	0,12766	0,10309	0,19519
<i>Trivy</i>	0,50000	0,50000	0,76596	0,61856	0,59613

$$\lambda_{\max} \approx 4,19659, \text{ ИС} \approx 0,06553, \text{ ОС} \approx 7,28 \%.$$

Таблица 7

Расчет весов по критерию "объекты анализа"

	<i>Kube-bench</i>	<i>Kube-hunter</i>	<i>Kubesploit</i>	<i>Trivy</i>	Среднее значение
<i>Kube-bench</i>	0,11111	0,10345	0,04348	0,14483	0,10072
<i>Kube-hunter</i>	0,22222	0,20690	0,26087	0,19310	0,22077
<i>Kubesploit</i>	0,22222	0,06897	0,08696	0,08276	0,11523
<i>Trivy</i>	0,44444	0,62069	0,60870	0,57931	0,56329

$$\lambda_{\max} \approx 4,26871, \text{ ИС} \approx 0,08957, \text{ ОС} \approx 9,95 \%.$$

Таблица 8

Расчет весов по критерию "полнота документации"

	<i>Kube-bench</i>	<i>Kube-hunter</i>	<i>Kubesploit</i>	<i>Trivy</i>	Среднее значение
<i>Kube-bench</i>	0,13636	0,16129	0,18750	0,12556	0,15268
<i>Kube-hunter</i>	0,13636	0,16129	0,31250	0,15695	0,19178
<i>Kubesploit</i>	0,04545	0,03226	0,06250	0,08969	0,05747
<i>Trivy</i>	0,68182	0,64516	0,43750	0,62780	0,59807

$$\lambda_{\max} \approx 4,12668, \text{ ИС} \approx 0,04223, \text{ ОС} \approx 4,69 \%.$$

Таблица 9

Расчет весов по критерию "опыт ЛПР"

	<i>Kube-bench</i>	<i>Kube-hunter</i>	<i>Kubesploit</i>	<i>Trivy</i>	Среднее значение
<i>Kube-bench</i>	0,23810	0,19048	0,31250	0,25000	0,24777
<i>Kube-hunter</i>	0,23810	0,19048	0,25000	0,16667	0,21131
<i>Kubesploit</i>	0,04762	0,04762	0,06250	0,08333	0,06027
<i>Trivy</i>	0,47619	0,57143	0,37500	0,50000	0,48065

$$\lambda_{\max} \approx 4,06015, \text{ ИС} \approx 0,02005, \text{ ОС} \approx 2,23 \%.$$

Таблица 10

Матрица весов альтернатив по критериям сравнения

	Покрытие уязвимостей	Объекты анализа	Полнота документации	Опыт ЛПП
<i>Kube-bench</i>	0,09782	0,15936	0,15268	0,24777
<i>Kube-hunter</i>	0,11087	0,16963	0,19178	0,21131
<i>Kubesploit</i>	0,19519	0,12225	0,05747	0,06027
<i>Trivy</i>	0,59613	0,54876	0,59807	0,48065

Такую матрицу надо умножить на вектор весов критериев (см. столбец "Среднее значение" в табл. 3), чтобы получить вектор весов альтернатив с точки зрения лица, принимающего решение (табл. 11).

Таблица 11

Вектор весов альтернатив ЛПП

Альтернатива	Вес*
<i>Kube-bench</i>	0,13237
<i>Kube-hunter</i>	0,14826
<i>Kubesploit</i>	0,14002
<i>Trivy</i>	0,57935

ЛПП выбирает инструмент тестирования по наибольшему значению в полученном векторе Вес\* (табл. 11), которое характеризует наилучший для ЛПП результат в пространстве проектных решений. В рассмотренном примере наибольший вес имеет инструмент *Trivy* (0,57935), поэтому для авторов данной статьи его использование для автоматизации тестирования в среде *Kubernetes* будет предпочтительным.

### Заключение

Использование инструментов автоматизированного тестирования объектов технологий виртуализации и контейнеризации необходимо для обеспечения надежности, производительности и безопасности этих сложных сред. Динамическая природа и сложные взаимозависимости виртуализированных и контейнерных систем требуют комплексного и систематизированного подхода к тестированию, который и обеспечивает рассмотренное ПО для автоматизации данного бизнес-процесса.

Рассмотрены различные инструменты автоматизации, которые помогают тестировать объекты контейнеризации в среде *Kubernetes*, такие как *Kubesploit*, *Kube-bench*, *Kube-hunter* и *Trivy*.

В результате исследования, авторы пришли к выводу, что такие решения, как *Kube-bench*, *Kube-hunter* и *Trivy*, разработанные компанией *Aqua Security*, имеют перспективны применения в российской практике, поскольку в процессе тестирования виртуального стенда с установленной платформой *Deckhouse*, разработанной

отечественной компанией и входящей в реестр российского ПО, они отличились результативностью применения.

При проведении сравнительного анализа инструментов автоматизации тестирования объектов контейнеризации с использованием метода анализа иерархий наиболее предпочтительным с точки зрения системного аналитика оказался инструмент *Trivy*.

### Литература

1. Устинов С. Основы контейнеризации (обзор *Docker* и *Podman*) [Электронный ресурс]. URL: <https://habr.com/ru/post/659049/> (дата обращения: 23.05.2023).
2. Хренников А. Основы *Kubernetes* [Электронный ресурс]. URL: <https://habr.com/ru/post/258443/> (дата обращения: 23.05.2023).
3. Positive Technologies. Итоги пентестов — 2022 [Электронный ресурс]. URL: <https://www.ptsecurity.com/ru-ru/research/analytics/results-of-pentests-2021-2022/#id15> (дата обращения: 23.05.2023).
4. Red Hat. State of Kubernetes security report 2022 [Электронный ресурс]. URL: <https://www.redhat.com/en/resources/state-kubernetes-security-report> (дата обращения: 23.05.2023).
5. Шурупов Д. Представляем *Kubernetes*-платформу *Deckhouse*. Теперь в Open Source и для всех [Электронный ресурс]. URL: <https://habr.com/ru/companies/flant/articles/569840/> (дата обращения: 23.05.2023).
6. Обзор инструментов для оценки безопасности кластера *Kubernetes*: *kube-bench* и *kube-hunter* [Электронный ресурс]. URL: <https://habr.com/ru/company/flant/blog/572328/> (дата обращения: 23.05.2023).
7. Checks whether *Kubernetes* is deployed according to security best practices as defined in the CIS *Kubernetes Benchmark* [Электронный ресурс]. URL: <https://github.com/aquasecurity/kube-bench> (дата обращения: 23.05.2023).
8. Hunt for security weaknesses in *Kubernetes* clusters [Электронный ресурс]. URL: <https://github.com/aquasecurity/kube-hunter> (дата обращения: 23.05.2023).
9. Gerzi Eviatar. *Kubesploit*: A New Offensive Tool for Testing Containerized Environments [Электронный ресурс]. URL: <https://www.cyberark.com/resources/threat-research-blog/kubesploit-a-new-offensive-tool-for-testing-containerized-environments> (дата обращения: 23.05.2023).
10. *Kubesploit* is a cross-platform post-exploitation HTTP/2 Command & Control server and agent written in Golang, focused on containerized environments [Электронный ресурс]. URL: <https://github.com/cyberark/kubesploit> (дата обращения: 23.05.2023).

11. Сканер уязвимостей Docker-контейнеров Trivy [Электронный ресурс]. URL: <https://cisoclub.ru/skaner-uyazvimostej-docker-kontejnerov-trivy/> (дата обращения: 23.05.2023).

12. Язык тестовых сценариев Testo Lang: простая автоматизация сложных тестов [Электронный ресурс]. URL: [https://habr.com/ru/company/testo\\_lang/blog/521682/](https://habr.com/ru/company/testo_lang/blog/521682/) (дата обращения: 02.04.2023).

13. Шевченко Д. В. Метод анализа иерархий [Электронный ресурс]. URL: <https://www.hse.ru/mirror/pubs/share/204810376> (дата обращения: 23.05.2023).

14. Образовательный портал ТГУ. Лекция 1. Метод анализа иерархий (метод Т. Л. Саати) [Электронный ресурс]. URL: [https://edu.tltsu.ru/sites/sites\\_](https://edu.tltsu.ru/sites/sites_)

content/site216/html/media67140/lec1\_is-2\_2020%20(1).pdf (дата обращения: 23.05.2023).

15. Михайлова В. М., Кузнецова О. А., Петрова А. В. Применение метода "Дельфи" как инструмента прогнозирования развития рынка [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/primeneniye-metoda-delfi-kak-instrumenta-prognozirovaniya-razvitiya-rynka> (дата обращения: 23.05.2023).

16. Калькулятор матриц онлайн [Электронный ресурс]. URL: <https://coinsite.ru/kalkulyator-matrits-onlajn/#eigenvectors%28%7B%7B1,3,1,1%2F2,5%7D,%7B3333%2F10000,1,1%2F4,1429%2F10000,2%7D,%7B1,4,1,1,6%7D,%7B2,7,1,1,8%7D,%7B1%2F5,1%2F2,833%2F5000,1%2F8,1%7D%7D%29> (дата обращения: 23.05.2023).

## USE OF TOOLS FOR AUTOMATING THE SECURITY TESTING OF CONTAINERIZATION OBJECTS

P. D. Belov

LLC "Locus Soft Technologies", Moscow, Russia

A. O. Makhmadzieev

LLC "RusBITech-Astra", Moscow, Russia

A. A. Pavlenko

LLC "CRM Solutions", Moscow, Russia

K. V. Pitelinskiy

Moscow Polytechnical University, Moscow, Russia

*The limitations of traditional approaches to testing the security of containerization objects and the advantages of using automation tools are discussed. The use of the Kubernetes software environment for automated testing of containerization objects is considered. Practical examples of using the Kube-bench, Kube-hunter, Kubesploit, Trivy tools to find vulnerabilities are given. The positive impact of test automation on quality assurance of containerized applications is emphasized.*

**Keywords:** containerization, automated testing, secure development, information security, software reliability.

Белов Павел Денисович, тестировщик.

E-mail: [pavel.belov00@mail.ru](mailto:pavel.belov00@mail.ru)

Махмадзиев Али Олимович, инженер.

E-mail: [amakhmadzieev@astralinux.ru](mailto:amakhmadzieev@astralinux.ru)

Павленко Александр Александрович, разработчик.

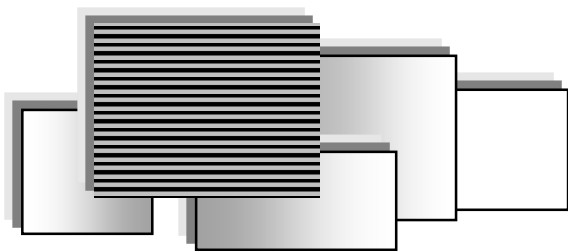
E-mail: [alex.pavlenk0o@gmail.com](mailto:alex.pavlenk0o@gmail.com)

Пителинский Кирилл Владимирович, доцент, МВА,

доцент кафедры "Информационная безопасность".

E-mail: [yekadath@gmail.com](mailto:yekadath@gmail.com)

Статья поступила в редакцию 26 мая 2023 г.



## ПРИКЛАДНЫЕ ВОПРОСЫ И ЗАДАЧИ ПРИМЕНЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

УДК 658.513

DOI: 10.52190/2073-2597\_2023\_2\_42

EDN: JUNEII

### ОПЫТ ПРИМЕНЕНИЯ МАШИНОЧИТАЕМЫХ ТРЕБОВАНИЙ В ПРОЕКТНОМ НАПРАВЛЕНИИ «ПРОРЫВ»

О. В. Жданова

Акционерное общество «Прорыв», Москва, Россия

О. Н. Столяров

Акционерное общество «Атомэнергопроект», Москва, Россия

*Представлен опыт разработки и применения машиночитаемых требований в реализуемом Государственной корпорацией по атомной энергии «Росатом» проекте «Прорыв». Дается пояснение о том, какую эволюцию прошли машиночитаемые требования, какие информационные технологии использовались для разработки и проверки машиночитаемых требований в последние десятилетия. В статье представлен опыт применения одного из методов в проектном направлении «Прорыв», приведены примеры требований, затрагиваются аспекты, затрудняющие распространение машиночитаемых требований как специального способа формулирования ограничений и их проверки, отражена ситуация с развитием отечественной нормативной базы в части перехода на машиночитаемые форматы.*

**Ключевые слова:** система управления требованиями, машиночитаемые требования, алгоритм проверки требований, автоматизированная проверка требований, язык записи требований.

Современный уровень развития информационных технологий существенно расширил круг решаемых задач. Одновременно с этим возникла потребность в таком хранении и представлении информации, которое позволит максимально использовать потенциал этих технологий. В частности, аналитика больших данных, искусственный интеллект, дополненная и виртуальная реальность, моделирование и симуляторы, трёхмерная печать требуют создания информационной модели объекта - очень точной многоаспектной формы его описания. Понятие «информационная модель объекта» закреплено законодательно. В 2019 г. в Градостроительный кодекс Российской Федерации введено понятие «информационная модель объекта капитального строительства». ГОСТ Р 57700.37-2021 «Компьютерные модели и моделирование. Цифровые двойники изделий. Общие положения» определяет эволюцию «информационной модели», вводя понятие «цифровой двойник». Имея настолько подробное математическое описание

предметов, позволяющее с высокой точностью имитировать их поведение, логично сделать следующий шаг развития — автоматизированную проверку выполнения нормативных требований, предъявляемых к объектам.

#### Методы автоматизированной проверки требований. История вопроса

Идея автоматизированной проверки требований не является новой. В 1960 г. была опубликована статья с идеями на эту тему [1], с тех пор по мере развития информационных технологий и языков программирования предпринималось большое число попыток решить вопрос разработки машиночитаемых документов и создания системы автоматизированной проверки информационных моделей, однако ранее применялась иная терминология.

В качестве ретроспективы развития мысли приведем основные методы автоматизированной



проверки требований: метод «прямого» кодирования; метод применения специализированных высокоуровневых языков программирования; метод, построенный на применении логических языков; метод применения расширяемого языка разметки; метод применения семантического подхода, основанного на технологии *Semantic Web*.

#### Метод «прямого» кодирования

Смысл этого метода понятен из его названия: для каждого требования разрабатывается программный код, который выполняет проверку требования. Для этого метода может быть выбран любой язык программирования. Недостатков у метода достаточно много, но наиболее важными являются большое разнообразие форматов САПР-данных и интероперабельность на уровне терминов. Названия объектов и предметов проверки должны совпадать в нормативно-технической и в электронной документации, также должна быть обеспечена наглядность того, что текст требования соответствует программному коду, который, к тому же, может быть скомпилирован. Для решения этих недостатков была предпринята попытка разработки языка более высокого уровня.

#### Метод применения специализированных высокоуровневых языков программирования

Одним из специализированных высокоуровневых языков программирования является *BERA* (*Building Environment Rule and Analysis*) [2]. Язык *BERA* пытается работать с информационными моделями зданий интуитивно понятным способом, чтобы создавать и анализировать правила, которые могут применяться для проверки конфигурации строительных объектов, компоновки зданий даже на ранних стадиях проектирования. Первоначальная реализация *BERA* включала общедоступную схему *IFC*. Схема *IFC*

определена в международном и российском стандартах [3], однако язык *BERA* не стал стандартом по ряду объективных причин. Развиваемая независимо от методов проверки требований схема *IFC* стала использоваться и в других альтернативных методах.

#### Метод, построенный на применении логических языков

В качестве примера можно привести *Semantic natural language processing-based automated compliance checking (SNACC)* [4].

По существу, использование метода сводится к трем шагам (рис. 1).

На первом шаге с помощью программного обеспечения класса *NLP* происходит автоматическое извлечение нормативных требований из строительных норм и правил, а также других типов строительных нормативных документов, и преобразование требований в логические пункты (логические правила).

На втором шаге происходит извлечение и преобразование проектной информации. С помощью языка запросов автоматически извлекается проектная информация из модели *IFC* и происходит преобразование информации в факты, высказанные на языке логики. Информация о конструкции автоматически согласуется с нормативной информацией в силу правил семантического преобразования, закодированных в логических высказываниях.

На третьем шаге выполняется автоматический ризонинг, то есть машина логического вывода проверяет логические факты на соответствие логическими правилами и генерирует отчеты о проверке соответствия.

При кажущейся простоте метода преобразовать с помощью *NLP* текст нормативных документов оказалось чрезвычайно сложной задачей. Однако идея с разметкой документов нашла свое дальнейшее развитие.

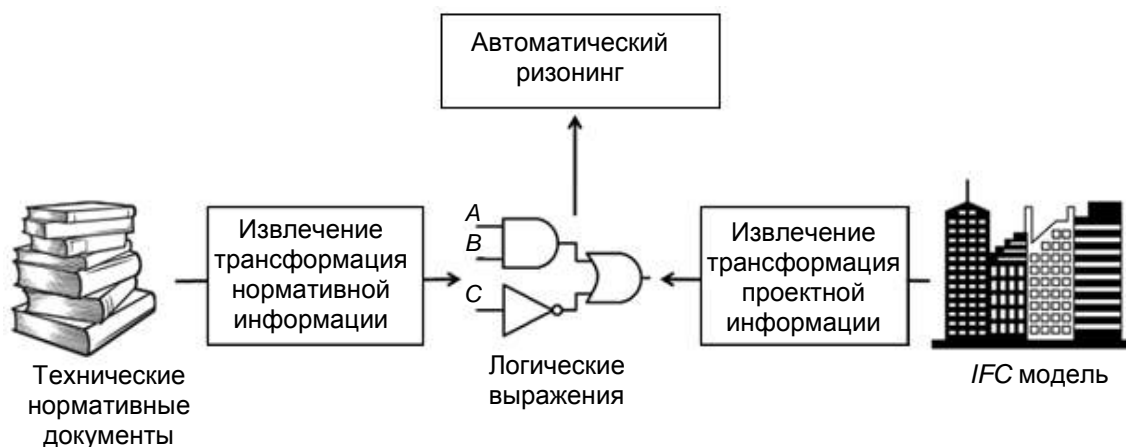


Рис. 1 Пример использования метода логических языков

### Метод применения расширяемого языка разметки

Примером расширяемого языка разметки является язык *RASE*. *RASE* — это семантическая концепция преобразования нормативных документов в единое четко определенное правило, которое может быть реализовано в программном обеспечении для проверки моделей на основе *IFC*. Методология *RASE* основана на ряде правил [5]. В языке разметки *RASE* используются следующие четыре типа *RASE*-меток - «Требование» «Применяется», «Выбор» и «Исключение», которые расставляются в тексте нормативных документов в виде *XML*-тэгов. Для реализации метода осуществляются те же три шага, что и для метода, построенного на применении логических языков.

Отличие *RASE* от *SNACC* в том, что метки представляет не компьютерная программа класса *NLP*, а эксперт. Еще одной особенностью метода применения расширяемого языка разметки является то, что по итогам разметки все требования помещаются в таблицу и дополняются атрибутами.

### Метод применения семантического подхода, основанного на технологии *Semantic Web*

Этот метод включает в себя идеи предыдущих методов, но его реализация основана на группе стандартов *W3C* с общим названием *Semantic Web*. Особенность технологии *Semantic Web* в том, что она использует язык дескрипционной логики и адаптированные для логической обработки способы хранения и извлечения информации, а также способы разметки *HTML* страниц и множество других вспомогательных инструментов [6].

Суть рассматриваемого метода и его главное отличие от метода, построенного на применении логических языков, в том, что первоначально формируется онтология предметной области, которая является и терминологическим словарем для разработки правил и метаданными для проектной информации. Кроме того, в отличие от *IFC*, онтология разрабатывается не ИТ-сообществом, а экспертами в предметной области - инженерами. Далее вся подлежащая проверке проектная информация извлекается из проектных баз данных и преобразовывается в высказывания на языке логики, но вместо *IFC* используется онтология. Автоматическое извлечение нормативных требований из строительных норм и правил предыдущего метода заменяется на переформулирование экспертами требований на язык логики (то, что в этом методе называется правилами). Так как вся информация хранится единым стандартным способом, то с помощью механизма логического вывода можно получать необходимые заключения о выполнении нормативных требований.

Существует несколько способов реализации технологии *Semantic Web*, некоторые представлены в работах [7, 8]. Еще один способ, примененный в проектом направлении «Прорыв», представлен ниже.

В 2020 г. аналогичные исследования в области существующих наработок, связанных с автоматизацией проверки информационной модели и в общем с подходами и методами представления требований научно-технической документации (НТД) в машиночитаемом формате, проводила организация НИЦ ЦПС. Командой НИЦ ЦПС было рассмотрено более 100 различных статей и других информационных источников. В итоге НИЦ ЦПС разработал свой способ формализации требований НТД, основанный на технологиях, которые, по его мнению, «уже сегодня можно использовать для решения конкретных практических задач» [9]. Среди них *XML*, *XQuery*, *XPath*, *SQL*, *MathML*, *SVG*. «Однако в будущем, по мнению экспертов НИЦ ЦПС, формализация требований будет двигаться в направлении семантических технологий *WEB*» [9].

### Опыт применения машиночитаемых требований в проектом направлении «Прорыв»

Проанализировав представленный выше опыт применения методов автоматизированной проверки требований, в проектом направлении «Прорыв» [10] по ряду причин сделали выбор в пользу метода применения семантического подхода, основанного на технологии *Semantic Web*. Во-первых, стратегически более правильно опираться на действующие стандарты *W3C*, чем на конкретные решения. Во-вторых, этот метод самый доступный для освоения и применения. В-третьих, у метода применения семантического подхода есть доступные ИТ-решения с открытым исходным кодом (*OpenSource*). Принимая во внимание нескончаемый поток санкций в отношении России, собственное решение правильнее строить на *OpenSource* разработках. Выбор в пользу программных решений с открытым исходным кодом позволит избежать зависимости от конкретного поставщика программ и, при необходимости, осуществлять доработку собственными силами. Сегодня, когда ряд компаний с российскими «корнями» сменили российскую юрисдикцию на зарубежную, это решение представляется для проекта единственным правильным.

Работа с машиночитаемыми требованиями в проектом направлении «Прорыв» проходит в четыре этапа:

- 1) Создание онтологии предметной области.
- 2) Создание семантической базы данных *Triple Store* [11], в качестве модели данных которой выступает онтология предметной области.
- 3) Выполнение разметки документа с требованиями на отдельные требования с уникальным

кодом. Каждое отдельное требование было переформулировано на языке *SHACL* [12] с сохранением исходного текста с требования.

4) Выполнение по команде пользователя автоматизированного ризонинга для получения логических выводов о выполнении требований.

Рассмотрим подробнее, как и с помощью каких инструментов были реализованы машиночитаемые требования и их автоматизированная проверка.

Как было отмечено выше, онтология является моделью данных, соответственно, каждый класс онтологии имеет экземпляры из состава хранимых данных. Требование на языке *SHACL* называются «Формой», так как является экземпляром класса *sh:Shape*. Для каждой «Формы» задается проверяемый класс онтологии или список классов. Указание на то, какие классы онто-

логии подлежат проверке в «Форме», задает отношение *sh:targetClass*. Примеры требований к экземплярам класса «Насос» и «Клапан», изложенных на языке *SHACL*, приведены на рис. 2 и 3.

В данной работе мы не будем подробно рассматривать синтаксис языка *SHACL*. Обратим внимание на то, то машиночитаемое требование является очень коротким, лаконичным логическим утверждением и не является фрагментом исполняемого кода какого-либо языка программирования. Отношение *sh:name* содержит уникальный код требования, а отношение *sh:description* содержит исходный текст требования из документа источника. Такая форма записи требования позволяет легко проверить соответствие утверждений, сделанных на естественном и математическом языках.

```

onto:PumpShape
  a sh:NodeShape ;
  sh:targetClass onto:Pump ;
  sh:property [
    sh:name "DQR.PMP.001" ;
    sh:description "Материалы для сборочных единиц и деталей трубопроводов, входящих
                    в состав насоса, применяют в соответствии с ГОСТ 20" ;
    sh:path onto:Material ;
    sh:minCount 1 ;
    sh:nodeKind sh:Literal ;
    sh:in ("30XMA" "18X3MB" "20X2MA" "20X3MBФ" "03X17H14M3" "08X18H10T" "08X18H12T"
          "12X18H10T" "12X18H12T" "10X17H13M2T" "10X17H13M3T" "08X17H15M3T") ;
  ] ;
.

```

Рис. 2. Пример машиночитаемого требования для насосов

```

onto: ValveShape
  a sh:NodeShape ;
  sh:targetClass onto:Valve ;
  sh:property [
    sh:name "DQR.VLV.004" ;
    sh:description "Изделие должно соответствовать 3-му классу безопасности по ОПБ-88/97" ;
    sh:path onto:SafetyClass ;
    sh:minCount 1 ;
    sh:nodeKind sh:Literal ;
    sh:in ("3") ;
  ] ;
.

```

Рис. 3. Пример машиночитаемого требования для клапанов

В машиночитаемом виде можно представить любые виды ограничений, накладываемые на атрибуты (характеристики) объектов. Например, все виды условий равенства/неравенства, проверка на наличие значения из определённого списка, проверка типов данных и даже проверки зависимости одних значений от других. Можно также проверять отсутствие нужных характеристик (некоторые примеры приведены на рис. 2 и 3) и формировать ограничения на мощность отношений, например на количество тех или иных атрибутов или других связанных объектов. В качестве примера на рис. 3 приведено отношение *sh:minCount*.

Кроме того, можно проверять любые тексты на соответствие шаблону строки, ее длине или другие виды операций со строками (рис. 4).

Еще один вид проверок касается отношений между объектами. Так как все отношения между объектами выражены семантическими предикатами и отражают инженерный взгляд на предметную область, то любые отношения между объектами и их данными, предусмотренные в онтологии, могут быть использованы при создании машиночитаемого требования.

На рис. 5 приведен пример проверки наличия организации—разработчика у документа через проверку существования отношения *onto:madeBy*.

Ризонинг, выполняемый программой в качестве вывода, показывает выявленные нарушения логики, однако этот вывод можно заменить пояснительным текстом, размещённым после отношения *sh:message*, как показано на рис. 4.

Для автоматизированной проверки машиночитаемых требований необходимо иметь соответствующее программное обеспечение. Ниже представлен список инструментов с открытым исходным кодом, поддерживающих проверку логических утверждений, сделанных на языке *SHACL*:

– *dotNetRDF SHACL* — онлайн-сервис-валидатор *SHACL*, написанный на платформе *.NET Framework*;

– *pySHACL* — библиотека валидаторов *SHACL* с открытым исходным кодом для использования в командной строке, написанная на *Python*;

– *SHaclEX* — *Scala*-реализация *SHACL* и *ShEx*;

– *SHACL Playground* — это бесплатный сервис проверки *SHACL*, реализованный на *JavaScript*.

Эти инструменты можно использовать самостоятельно или включить в состав своих ИТ-систем. В частности, для обеспечения регулярной проверки выполнения машиночитаемых требований в проектом направлении «Прорыв» был разработан собственный модуль на языке *Python*, включающий библиотеку *pySHACL*.

```

onto:ReqIDDocumentShape a sh:NodeShape ;
sh:targetClass onto:Document ;
sh:property [
  sh:name "DQR.018.001" ;
  sh:description "Суффикс в коде Ссылка должен соответствовать типу фрагмента текста и иметь значение L" ;
  sh:path rdfs:label ;
  sh:pattern "^.+L$" ;
  sh:flag "i" ;
  sh:message "Код не соответствует типу фрагмента текста." ;
] ;

```

Рис. 4. Пример машиночитаемого требования для документов

```

onto:madeByShape a sh:NodeShape ;
sh:targetObjectsOf onto:madeBy ;
sh:property [
  sh:path rdf:type ;
  sh:name "DQR.009.001" ;
  sh:description "У каждого документа должен быть указан его разработчик из числа Организаций";
  sh:nodeKind sh:IRI ;

```

Рис. 5. Пример машиночитаемого требования для проверки отношения между объектами

Этот модуль является частью информационной системы управления требованиями, используемой в проекте «Прорыв» для контроля проектных требований. В модуле реализованы три экранные формы:

- форма для редактирования требований (форма доступна только пользователям с определёнными правами);
- форма для просмотра и выбора требований и запуска их автоматизированной проверки (форма доступна любому пользователю модуля);
- форма вывода результатов проверки. Результаты проверки можно сохранить в виде файла и отправить заинтересованным сторонам.

Применяемое в проекте направление «Прорыв» решение является частью информационной системы управления требованиями, введенной в промышленную эксплуатацию.

### Заключение

Опыт разработки и использования в проекте направления «Прорыв» машиночитаемых требований с помощью *Semantic Web* показал доступность и развитость необходимых технологий, возможность их применения в рамках организации. Однако необходимый экономический эффект от применения машиночитаемых требований можно получить только после того, как данный подход выйдет за границы организаций и будет использован в ходе приемки работ, в процессе выполнения экспертиз и прохождения различных разрешительных процедур, лицензирования. Именно сокращение сроков проверки соответствия проектной документации нормам и правилам существенно скажется на всей экономике страны. Трудоемкость экспертиз составляет многие человеко-месяцы, что делает процедуру лицензирования очень длительной и дорогостоящей для всех задействованных сторон. Автоматизированная проверка требований не только сократит это время до машино-часов, но и сделает ее результаты более объективными.

Опыт проекта «Прорыв» по разметке нормативной документации на отдельные требования для создания лицензионной базы в информационной системе управления требованиями показал низкую пригодность большинства нормативных документов к автоматизации в силу особенностей естественного языка и применения оценочных категорий, неоднозначности формулировок, отсутствие единой согласованной терминологии и единой структуры документа.

Аспектами, влияющими на перевод требований в машиночитаемый вид, являются законодательные ограничения, предписывающие сдавать определенные документы на бумажном носителе с «живой» подписью и печатью, а также отсутствие законодательно определенных форм электронных документов, информационных моделей

и используемых для передачи и обработки данных информационных технологий (протоколы, программное обеспечение). При этом следует отметить, что опыт приемки данных от организаций и населения в цифровом виде уже достаточно большой. Например, Федеральная налоговая служба Российской Федерации принимает отчетность в цифровом виде; работа с обращениями граждан ведется через портал *gosuslugi.ru* и др.

И, наконец, еще один барьер — это необходимость участия человека при проверке определенных норм.

Однако несмотря на перечисленные аспекты, переход на машиночитаемые требования стоит в планах российских и зарубежных разработчиков стандартов и законов. В качестве примера можно привести создание «Стратегической консультативной группы по машиночитаемым стандартам» (*ISO Strategic Advisory Group on Machine Readable standards*) в Международной организации по стандартизации (ИСО), в задачи которой входит разработка единого понятийного аппарата и рекомендаций по внедрению машиночитаемых стандартов для технических комитетов по стандартизации.

В России в настоящее время идет работа по правовому оформлению возможности проведения автоматизированной проверки проектной документации на соответствие нормативным документам для строительной отрасли. Законодательное оформление этого вопроса ожидается в ближайшее время согласно Стратегии развития [13]. В соответствии с [13] предусмотрены следующие мероприятия: устранение излишних противоречивых дублирующих нормативных технических требований в строительстве, перевод нормативных технических документов в машиночитаемый и машинопонимаемый форматы.

В заключение хочется отметить, что будущее формируется уже сейчас, поэтому проведение организациями научно-исследовательских и опытно-конструкторских работ, разработка пилотных проектов, информирование широкого круга лиц о своих достижениях по этому вопросу было бы очень своевременно и позволило бы выбрать наилучшее из опробованных решений, а законодательное оформление перехода на машиночитаемые стандарты позволит зафиксировать единые подходы, сократить сроки рассмотрения проектной документации и получения лицензий на строительство объектов.

---

*Работа выполнена при поддержке Российской Федерации, от имени которой выступает Госкорпорация «Росатом», в рамках Государственного контракта №. Н.40.24 1.19.21.1037 от 28.02.2022 с дополнительным соглашением от 20.06.2022 № 1.*

**Литература**

1. Chasalow Mr. The Element of Predictability in Judicial Decision Making/ California State Bar Journal, Nov.-Dec. 1960.
2. Jin Kook Lee. Building Environment Rule and Analysis (BERA) Language. URL: [https://smartech.gatech.edu/bitstream/handle/1853/39482/Lee\\_Jin-Kook\\_201105\\_PhD.pdf](https://smartech.gatech.edu/bitstream/handle/1853/39482/Lee_Jin-Kook_201105_PhD.pdf) [Электронный ресурс] (Дата обращения: 12.05.2023).
3. ГОСТ Р 10.0.02-2019/ИСО 16739-1:2018 «Система стандартов информационного моделирования зданий и сооружений. Отраслевые базовые классы (IFC) для обмена и управления данными об объектах строительства».
4. Jiansong Zhang. A logic-based representation and tree-based visualization method for building regulatory requirements/ Visualization in Engineering. 16.03.2017.
5. Hjelseth E., Nisbet N. (2011). Capturing normative constraints by use of the semantic mark-up RASE methodology// Proceedings of CIB W78-W102 Conference, 1—10, 2011.
6. Web standards. URL: <https://www.w3.org/standards/semanticweb/> [Электронный ресурс] (Дата обращения: 12.05.2023).
7. Beach T.H., Rezgui Y.R., Li H., Kasim T. A Rule-based Semantic Approach for Automated Regulatory Compliance in the Construction Sector. URL: [https://publications.aston.ac.uk/id/eprint/39414/1/Rule\\_based\\_semantic\\_approach.pdf](https://publications.aston.ac.uk/id/eprint/39414/1/Rule_based_semantic_approach.pdf) [Электронный ресурс] (Дата обращения: 12.05.2023).
8. Project openLAB: Interoperate – Machine Readable Handbook V440. URL: <https://buildingpoint-scandinavia.com/blog/project-interoperate-v440> [Электронный ресурс] (Дата обращения: 10.05.2023).
9. Волкодав В. Цифровой нормативно-технический документ в строительстве. Применение в рамках автоматизации процессов проверки ИМ ОКС // Материалы онлайн-конференции «Машиночитаемые стандарты: перспективы применения в промышленности» 25.02.2021. С.41—46. URL: [MAISHSTANDARTY\\_отчет\\_буклет.pdf](https://maishstandarty.ru/maishstandarty_otchet_buklet.pdf) (rgtr.ru) [Электронный ресурс] (Дата обращения: 12.05.2023).
10. Проект «Прорыв». URL: <https://proryv2020.ru/o-proekte/> [Электронный ресурс] (Дата обращения: 12.05.2023).
11. LargeTripleStores. URL: <https://www.w3.org/wiki/LargeTripleStores> [Электронный ресурс] (Дата обращения: 12.05.2023).
12. Shapes Constraint Language (SHACL). W3C Recommendation 20 July 2017. URL: <https://www.w3.org/TR/shacl/> [Электронный ресурс] (Дата обращения: 12.05.2023).
13. Стратегия развития строительной отрасли и жилищно-коммунального хозяйства Российской Федерации на период до 2023 года с прогнозом до 2025 года. Утв. Распоряжением Правительства Российской Федерации от 31.10.2022 от № 3268-р.

## THE EXPERIENCE OF MACHINE-READABLE REQUIREMENTS APPLICATION IN PRORYV PROJECT

O. V. Zhdanova

"Proryv" Joint Stock Company, Moscow Russia

O. N. Stolyarov

"Atomenergoproekt" Joint Stock Company, Moscow Russia

*The article presents the experience of Proryv Project implemented by the State Atomic Energy Corporation Rosatom in developing and applying machine-readable requirements in requirement management. The authors provide an explanation of the evaluation of machine-readable requirements, what information technologies have been used to develop and verify machine-readable requirements in recent decades. The article presents in details the experience of applying one of the methods in Proryv Project and provides examples of requirements. Aspects that make difficult the further distribution of machine-readable requirements as a special way of identifying restrictions and their verification are covered in the article. Current situation with the development of Russian regulatory framework in this field is touched as well.*

**Keywords:** requirements management system, machine-readable requirements, requirements verification algorithm, automated requirement verification, requirements recording language.

**Жданова Оксана Валерьевна**, начальник отдела развития информационной модели и системы управления требованиями ЗЯТЦ.  
E-mail: [zhdov@proryv2020.ru](mailto:zhdov@proryv2020.ru)  
**Столяров Олег Николаевич**, начальник управления научно-технологического обеспечения проектирования АЭС.  
E-mail: [Stolyarov\\_ON@aep.ru](mailto:Stolyarov_ON@aep.ru)

*Статья поступила в редакцию 6 июня 2023 г.*

Информационные технологии в проектировании и производстве: Науч.-техн. журнал /  
ФГУП «НТЦ оборонного комплекса «Компас», 2023. № 2 (190). С. 1—52.

Компьютерная верстка: *Н. В. Ильина, К. В. Трыкина*

☎ редакции (495) 491-43-17  
E-mail: [izdanie@ntckompas.ru](mailto:izdanie@ntckompas.ru)  
<http://ntckompas.ru>

Подписано в печать 25.06.2023.  
Формат 60×84 1/8. Бумага офсетная.  
Усл. печ. л. 6,0. Уч.-изд. л. 6,2.  
Заказ 2018. Тираж 250 экз.  
Адрес редакции: 125424, Москва,  
Волоколамское ш., 77,  
ФГУП «НТЦ оборонного комплекса «Компас».  
Отпечатано: 101000, Москва,  
Лубянский проезд, д. 15, стр. 4, помещ. IX,  
ком. 15, 16  
ООО «Спиди-Принт.ру»  
Индекс 79378.



# Правила для авторов по оформлению рукописей статей, предлагаемых к публикации в журналах

Направляя рукопись статьи в редакцию журнала, авторы передают редколлегии и издателю журнала безвозмездное неисключительное право опубликовать ее на русском языке в качестве статьи в печатной и электронной версиях журнала в сети Интернет. При этом за авторами сохраняются их интеллектуальные права на рукопись статьи (в том числе "авторское право"). В связи с этим и с учетом Четвертой части (Раздел VII) Гражданского кодекса РФ авторами должно быть представлено в редакцию письмо в следующей форме:

## Лицензионный договор о передаче права на публикацию (издательский лицензионный договор)

Мы, нижеподписавшиеся, авторы рукописи .....,  
предоставляем редколлегии журнала и издателю ФГУП «НТЦ оборонного комплекса «Компас»  
.....  
(название журнала)

безвозмездную простую (неисключительную) лицензию на публикацию рукописи статьи как в печатной, так и в электронной версиях журнала.

Мы подтверждаем, что данная публикация не нарушает интеллектуальных прав других лиц или организаций.

Подписи авторов: ..... (ф.и.о., ученая степень, дата)

Статья должна быть подписана всеми авторами. В случае нескольких авторов указывается фамилия автора, ответственного за переписку с редакцией. Рукопись статьи направляется на адрес редакции журнала: 125424, Москва, Волоколамское ш., д. 77, ФГУП «НТЦ оборонного комплекса «Компас», Редакция журнала. Дополнительная информация может быть получена в редакции при обращении по телефону: 8 (495) 491-43-17 или по E-mail: izdanie@ntckompas.ru.

Каждая статья рецензируется в соответствии с решением редколлегии журнала. Рецензенты выбирают из числа ученых и специалистов, компетентных в вопросах, рассматриваемых в статье, и имеющих собственные публикации в данном направлении. Решение о порядке публикации статьи принимается редколлгией с учетом мнения рецензентов. В случае разногласий среди членов редколлегии окончательное решение принимает главный редактор. При полном отклонении рукописи статьи от публикации редакция журнала направляет авторам мотивированный отказ. По этому факту редакция не вступает в полемику с авторами.

### Комплектование статьи (обзора)

Статья должна быть представлена в редакцию в следующем комплекте:

- сопроводительное письмо, подписанное руководством организации;
- экспертное заключение о возможности опубликования в открытой печати и распространения в Российской Федерации и других государствах. При отсутствии в организации экспертной комиссии вме-

сто экспертного заключения может быть представлено соответствующее заявление на имя главного редактора журнала от имени руководства организации, где работают авторы. В экспертном заключении или заявлении в обязательном порядке должна быть отражена возможность открытой публикации и передачи публикуемых материалов за границу.

- лицензионный договор по приведенному образцу;

- рукопись статьи — объем статьи (без рисунков) не должен превышать 10 страниц формата А4 при 1,5 межстрочных интервалах, а объем статьи обзорного характера — 20 страниц. Рекомендуемая гарнитура — New Times Roman. Размер шрифта — 12;

- Материал статьи представляется в редакцию в печатном виде (на бумажном носителе) и в электронном варианте с текстом в формате Word на электронную почту.

### Оформление статьи:

- статья начинается с указания УДК;
- название статьи набирается строчными буквами (кроме начальной прописной) полужирным шрифтом, размер шрифта 14, для остального текста используется простой шрифт размером 12, причем рекомендуемая гарнитура шрифта — Times New Roman;
- после названия — список авторов, инициалы авторов предшествуют их фамилиям;

- далее представляется аннотация статьи (10—15 строк с раскрытием цели работы и её основных результатов);

- далее приводится список ключевых слов для данной статьи (не более десяти);

- страницы текста нумеруются без пропусков и добавлений литерных обозначений (типа 1а, 2б и т. п.), причем в сквозную нумерацию должны быть включены все элементы статьи;

- внизу первой страницы текста помещается отдельный абзац (с полужирным шрифтом), содержащий контактную информацию об авторе (или авторах) в следующем виде: фамилия, имя, отчество, должность, ученая степень, почтовый адрес предприятия, телефон, E-mail;

- основной текст статьи должен начинаться с четкой постановкой цели и задач работы, сопровождаемой аргументами в пользу ее выполнения на фоне существующего состояния затронутой в статье проблемы. Дальнейший текст статьи также должен иметь смысловые рубрикаторы (разделы и подразделы) без их нумерации. Заканчиваться статья должна отдельным разделом «Заключение» с перечислением основных результатов, следующих из них выводов и, по возможности, предложений по развитию исследований и использованию их результатов.

- после основного текста — список использованных источников "Литература" (не менее 5 источников); Список использованной литературы должен соответствовать всем ссылкам на внешние источники в тексте статьи. Ссылки оформляются в квадратных скобках, например, [1—6], [7, 8]. Внутренние ссылки, т. е. ссылки на формулы, рисунки и таблицы статьи оформляются с использованием круглых скобок, например, формула (1), уравнение (4), (рис. 3), (табл. 2). Любые ссылки в подписях к рисункам и в самих рисунках не рекомендуются;

- далее размещается подробная англоязычная информация о статье: название статьи, фамилия и инициалы авторов (английская транслитерация), предприятие, аннотация, ключевые слова (Keywords).

### **Оформление рисунков:**

- рисунки и графики вставляются непосредственно в нужном месте в статье и в желаемом масштабе.

- рядом с осями графиков указываются отображаемые физические величины только в символьной (буквенной) форме, а через запятую — размерность величины по-русски (прямым шрифтом). Различные кривые на графиках рекомендуется нумеровать, даже если они характеризуются отдельным цветом или типом линии. Графики представляются только на

белом фоне. Вспомогательные сетки на площади графика не допускаются;

- подписи под соответствующими рисунками (полужирный курсивный) представляются в нужных местах текста. Каждая подпись должна быть по возможности лаконичной, но емкой по содержанию.

### **Оформление формул:**

- простые формулы вводить в текст в формате используемого текстового редактора, более сложные формулы — с использованием редактора формул MathType;

- стандартные математические обозначения (например,  $\max$ ,  $\log$ ,  $\sin$ ,  $\exp$  и т. д.) должны быть набраны прямо. То же относится к цифрам и числам;

- для символьного обозначения не векторных физических (технических) величин использовать только латинский и греческий алфавиты, при этом в тексте для греческих букв использовать прямой шрифт, для латинских букв — наклонный шрифт (курсив);

- векторы и матрицы обозначать полужирным прямым шрифтом;

- для нижних и верхних индексов применять арабские цифры, латинские или греческие буквы. Если индекс представляет собой сокращенную форму русского слова — характеристики, то допустимо использовать в его обозначении русские буквы (прямой шрифт), например  $U_{\text{вх}}$ ,  $I_{\text{вых}}$ ,  $v_{\text{гр}}$  и т. п.

- размерность физических величин обозначается всегда только по-русски прямым шрифтом.

### **Оформление таблиц:**

- содержание таблицы не должно дублировать данные, приводимые на графиках или в тексте;

- графы должны иметь название без сокращения отдельных слов.

Рисунки, формулы и таблицы должны иметь свою отдельную сквозную нумерацию. Если на конкретную формулу нет дополнительных (возвратных) ссылок в тексте или она в единственном числе, то нумерация ее не нужна. Единственные таблица и/или рисунок также не нумеруются.

При публикации в журнале каждая статья (в контактной информации) сопровождается сноской со знаком охраны авторского права ©, поставленным перед фамилией автора (фамилиями авторов) и годом издания.

Авторы (или автор) каждой статьи после выхода журнала в свет имеют право на получение от редакции электронной версии статьи в PDF-формате (редактор Adobe Acrobat).

**БЛАНК-ЗАКАЗ НА ПОДПИСКУ 2023 г.**  
**на издания ФГУП «НТЦ оборонного комплекса «Компас»**

Наименование издания	Индекс издания (количество выпусков в год)	Цена 1 экз., руб.	Кол-во выпусков в год	Общая сумма, руб.
Оборонный комплекс — научно-техническому прогрессу России	4	1600,00		
Конструкции из композиционных материалов	4	1750,00		
Экология промышленного производства	4	1600,00		
Информационные технологии в проектировании и производстве	4	1800,00		
Вопросы защиты информации	4	1800,00		
В цену включены: НДС — 10 % и стоимость почтовой доставки.				

*Поставка журналов подписчикам через издательство осуществляется почтовыми бандеролями с приложением всех необходимых бухгалтерских документов.*

**Наши реквизиты:**

Полное наименование организации: \_\_\_\_\_

Сокращенное наименование организации: \_\_\_\_\_

ИНН/КПП \_\_\_\_\_

ОКПО \_\_\_\_\_

Расчётный счёт № \_\_\_\_\_ в \_\_\_\_\_

к/с \_\_\_\_\_ БИК \_\_\_\_\_

Юридический адрес: \_\_\_\_\_

Почтовый адрес: \_\_\_\_\_

Контактное лицо \_\_\_\_\_ тел. \_\_\_\_\_

E-mail: \_\_\_\_\_

*(Для оформления счёта и бухгалтерских документов просьба заполнить все строчки).*

**Справочно:**

Заполненный бланк-заказ просьба отправить по факсу: 8(495) 491-44-80 или

E-mail: secretariat@ntckompas.ru

Более подробную информацию об изданиях и подписке можно получить по телефону:

8 (495) 491-43-17

E-mail: ivleva@ntckompas.ru

Адрес редакции: 125424, Москва, Волоколамское шоссе, д.77.

ФГУП «НТЦ оборонного комплекса «Компас».