

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 24
2018
№ 8

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

DOI 10.17587/issn.1684-6400

УЧРЕДИТЕЛЬ

Издательство "Новые технологии"

СОДЕРЖАНИЕ

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

Божко А. Н. Методы структурного анализа сложных изделий в интегрированных CAD/CAM-системах 499

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Цветков В. Я. Алгоритмы как средство познания 507

Шучалова Ю. С., Ланин В. В. Исследовательский портал для анализа и оценки стиля научных публикаций 515

Лиманова Н. И., Седов М. Н. Алгоритм нечеткого поиска реквизитов физических лиц в базах данных на основе метрики Левенштейна 523

ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ

Александров А. Е. Оценка достоверности результатов контроля металла на основе альтернативного алгоритма 529

Чунг Ле Ба, Холопов Ю. А. Информационная среда цифровых систем управления, использующая для управления периферией концепцию "вынесенной руки" 538

ИНТЕЛЛЕКТУАЛЬНЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

Абрамов А. Г. Панъевропейский научно-образовательный сетевой консорциум GEANT: особенности инфраструктуры, ключевые проекты и сервисы 546

Романов А. А., Волчек Д. Г., Муромцев Д. И. Семантические технологии в массовых открытых онлайн-курсах: экстракция, обогащение и связывание данных электронных учебных курсов для образовательной платформы Open edX 553

Главный редактор:

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

Зам. главного редактора:

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

Редакционный совет:

БЫЧКОВ И. В., акад. РАН, д. т. н.

ЖУРАВЛЕВ Ю. И.,

акад. РАН, д. ф.-м. н., проф.

КУЛЕШОВ А. П.,

акад. РАН, д. т. н., проф.

ПОПКОВ Ю. С.,

акад. РАН, д. т. н., проф.

РУСАКОВ С. Г.,

чл.-корр. РАН, д. т. н., проф.

РЯБОВ Г. Г.,

чл.-корр. РАН, д. т. н., проф.

СОЙФЕР В. А.,

акад. РАН, д. т. н., проф.

СОКОЛОВ И. А.,

акад. РАН, д. т. н., проф.

СУЕТИН Н. В., д. ф.-м. н., проф.

ЧАПЛЫГИН Ю. А.,

акад. РАН, д. т. н., проф.

ШАХНОВ В. А.,

чл.-корр. РАН, д. т. н., проф.

ШОКИН Ю. И.,

акад. РАН, д. т. н., проф.

ЮСУПОВ Р. М.,

чл.-корр. РАН, д. т. н., проф.

Редакционная коллегия:

АВДОШИН С. М., к. т. н., доц.

АНТОНОВ Б. И.

БАРСКИЙ А. Б., д. т. н., проф.

ВАСЕНИН В. А., д. ф.-м. н., проф.

ВАСИЛЬЕВ В. и., д. т. н., проф.

ВИШНЕКОВ А. В., д. т. н., проф.

ДИМИТРИЕНКО Ю. И., д. ф.-м. н., проф.

ДОМРАЧЕВ В. Г., д. т. н., проф.

ЗАБОРОВСКИЙ В. С., д. т. н., проф.

ЗАРУБИН В. С., д. т. н., проф.

КАРПЕНКО А. П., д. ф.-м. н., проф.

КОЛИН К. К., д. т. н., проф.

КУЛАГИН В. П., д. т. н., проф.

КУРЕЙЧИК В. В., д. т. н., проф.

ЛЬВОВИЧ Я. Е., д. т. н., проф.

МАРТЫНОВ В. В., д. т. н., проф.

МИХАЙЛОВ Б. М., д. т. н., проф.

НЕЧАЕВ В. В., к. т. н., проф.

ПОЛЕШУК О. М., д. т. н., проф.

САКСОНОВ Е. А., д. т. н., проф.

СОКОЛОВ Б. В., д. т. н., проф.

ТИМОНИНА Е. Е., д. т. н., проф.

УСКОВ В. Л., к. т. н. (США)

ФОМИЧЕВ В. А., д. т. н., проф.

ШИЛОВ В. В., к. т. н., доц.

Редакция:

БЕЗМЕНОВА М. Ю.

ЛЫСЕНКО А. В.

ЧУГУНОВА А. В.

Информация о журнале доступна по сети Internet по адресу <http://novtex.ru/IT>.

Журнал включен в систему Российского индекса научного цитирования и базу данных RSCI на платформе Web of Science.

Журнал входит в Перечень научных журналов, в которых по рекомендации ВАК РФ должны быть опубликованы научные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

INFORMATION TECHNOLOGIES

INFORMACIONNYYE TEHNOLOGII

Vol. 24
2018
No. 8

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

ISSN 1684-6400

CONTENTS

CAD-SYSTEMS

- Bozhko A. N.** Methods of Structural Analysis of Complex Products in Integrated CAD/CAM-systems 499

INTELLIGENT SYSTEMS AND TECHNOLOGIES

- Tsvetkov V. Ya.** The Algorithms as a Means of Knowledge 507

- Shuchalova Yu. S., Lanin V. V.** Research Portal for Scientific Publication Style Analysis 515

- Limanova N. I., Sedov M. N.** Fuzzy Searching Algorithm of Personal Details on the Basis of Levenshtein Distance 523

APPLICATION INFORMATION SYSTEMS

- Aleksandrov A. E.** Evaluation of the Reliability of Metal Testing Results Based on an Alternative Algorithm 529

- Le Ba Chung, Holopov Y. A.** Information Environment of Digital Control Systems, Using the Concept of "Outstretched Hand" to Control Peripherals 538

INTELLECTUAL TECHNOLOGIES IN EDUCATION

- Abramov A. G.** Pan-European Research and Education Network Consortium GÉANT: Features of Infrastructure, Key Projects and Services 546

- Romanov A. A., Volchek D. G., Mouromtsev D. I.** Towards Semantic MOOC: Extracting, Enriching and Interlinking E-learning Data in Open edX Platform ... 553

Editor-in-Chief:

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

Deputy Editor-in-Chief:

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

Chairman:

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

Editorial Board Members:

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

Editors:

Bezmenova M. Yu.
Lysenko A. V.
Chugunova A. V.

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

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

А. Н. Божко, канд. техн. наук, доц., e-mail: abozhko@inbox.ru,
Московский государственный технический университет им. Н. Э. Баумана

Методы структурного анализа сложных изделий в интегрированных CAD/CAM-системах

Рассматривается проблема структурного анализа сложных изделий в системах автоматизированного проектирования. Предложена гиперграфовая математическая модель механической структуры. Показано, что связанные и геометрически координированные фрагменты изделия описываются стягиваемыми гиперграфами. Данную модель можно использовать для глубокого структурного анализа проекта. Она позволяет обнаружить многие проектные ошибки и потенциально опасные решения на самых ранних стадиях проектирования. Предложены критерии для идентификации перебазирования, некогерентности, потери координации деталей и др. Разработан способ оценки "глубины вложения" деталей, который позволяет оценить сложность разборки и структурную ремонтпригодность изделия.

Ключевые слова: сборка, механическая структура, структурный анализ, гиперграф, автоматизированная система проектирования, перебазирование, геометрическая координация, разборка

Введение

При создании трехмерной геометрической модели изделия оператор CAD-системы не связан жесткими технологическими ограничениями, поэтому для координации деталей он может использовать абстрактные геометрические элементы: точки, линии, оси, биссектрисы, медианы и др. В технологической подготовке производства, на этапах производства и ремонта определенность положения детали или заготовки достигается с помощью баз: конструкторских, технологических, измерительных, ремонтных. Преобразование системы геометрических связей в механические сопряжения, а далее в базы является сложной, многовариантной и неформализованной проектной процедурой. В результате нее формируется так называемая механическая структура изделия. От механической структуры и ее согласованности с другими конструктивно-технологическими свойствами изделия зависят многие свойства технической системы в операциях сборки, разборки и разбиения на сборочные единицы.

В современных CAD-системах отсутствуют средства глубокого анализа механической структуры изделия как отдельного проектного решения. Она формируется во многом автома-

тически, в процессе создания трехмерной геометрической модели изделия и верификации контактов и зазоров между моделями элементов. Свойства механической структуры объективируются на последующих этапах жизненного цикла, связанных с технологической подготовкой производства. Поэтому необходимо разработать методы структурного анализа, которые в процессе проектирования способны оценить качество механической структуры и спрогнозировать поведение изделия на этапах сборки, разборки и ремонта.

В актуальных исследованиях и разработках по автоматизации проектирования основной структурной моделью технических объектов служит граф связей. Этот носитель проектной информации предложен А. Bourjault [1]. В настоящее время граф связей под разными названиями (Liaison diagram, Part mating graph, Connective relation graph, Component mating graph, Connection graph, Liaison graph, Parts liaison graph, Hierarchical relation graph, Skeleton graph, Attributed liaison graph, Product liaison graph, Component graph, Adjacency graph, Weighted undirected connected graph и др.) активно используется для структурного анализа и синтеза проектных решений сложных технических систем [1–11].

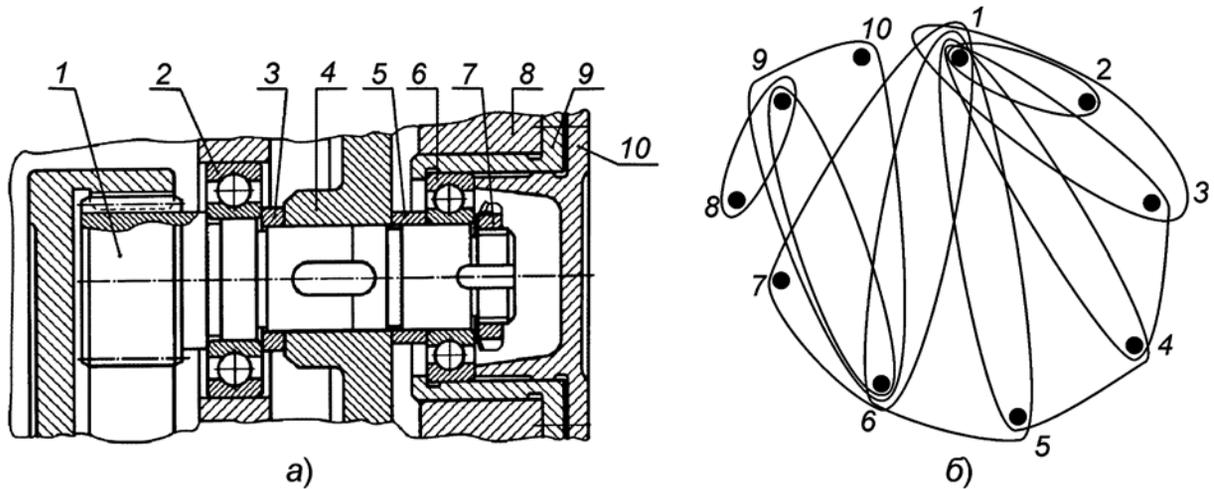


Рис. 1. Конструкция промежуточного вала (а) и гиперграф конструкции H_s (б)

Приведем точное определение графа связей. Пусть $X = \{x_i\}_{i=1}^n$ — множество деталей изделия. В графе связей $G = (X, V)$ множество вершин X описывает детали изделия, а множество ребер V — соединения и сопряжения между ними. Ребро $v = (x, y)$, $v \in V$, соединяет вершины x и y тогда и только тогда, когда между соответствующими деталями существует механическая связь.

В некоторых исследованиях основным носителем структурной информации об изделии служит матрица контактов [2]. Легко показать, что эта матрица является матрицей смежности графа связей, что дает основания считать равносильными графовый и матричный языки описания структурных данных.

Пространственная координация деталей при сборке изделия достигается базированием по конструкторским базам. Легко привести примеры, когда конструкторские базы принадлежат нескольким (двум, трем и т. д.) разным деталям. Поэтому базирование является отношением переменной местности, которое **невозможно корректно описать бинарными математическими средствами, коими являются граф связей и его различные вариации.**

Граф связей не обладает предсказательной силой настоящей математической модели и, по сути дела, является дескриптивной моделью. С его помощью можно решить только самые простые задачи структурного анализа, например, провести анализ связности, поиск мостов, перегруженных вершин и решить некоторые другие задачи [3].

В работах автора [13, 14] предложена новая гиперграфовая модель структуры сложного изделия, которая корректно описывает его свойства в операциях сборки, разборки и разбиения

на сборочные единицы. Эта модель позволяет выполнить глубокий анализ и идентифицировать многие структурные "пороки" и ненадежные проектные решения на самых ранних стадиях технической подготовки производства.

1. Гиперграфовая модель структуры сложных изделий

Изделию поставим в соответствие гиперграф $H = (X, R)$, в котором вершины X описывают детали, а гиперребра R — минимальные геометрически определенные группировки деталей, полученные базированием по конструкторским базам. На рис. 1 изображены конструкция промежуточного вала (а) и гиперграф H_s данной конструкции (б).

Определение 1. Сборочная операция называется секвенциальной (двуручной), если ее можно выполнить с помощью двух исполнительных органов. В противном случае операция называется несеквенциальной (мнгоручной) [2, 11].

На рис. 2 показан простейший пример несеквенциальной (трехручной) сборочной операции.

Процесс сборки изделия можно описать как последовательность стягиваний ребер ги-

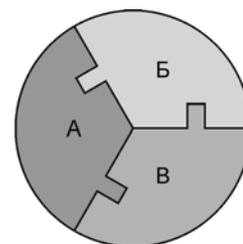


Рис. 2. Пример несеквенциальной сборочной операции

перграфа H . Стягиваются только ребра второй степени, поскольку можно с высокой степенью уверенности полагать, что в инженерной практике не используются несеквенциальные (многоручные) сборочные операции.

Определение 2. Нормальным стягиванием назовем операцию отождествления двух вершин гиперграфа, связанных ребром второй степени, и удаление данного ребра.

Определение 3. Гиперграф, который некоторой последовательностью нормальных стягиваний преобразуется в одновершинный гиперграф без петель и изолированных ребер (точку), назовем s -гиперграфом [13].

Теорема 1. Если гиперграф $H = (X, R)$ является s -гиперграфом, то:

- 1) он связный;
- 2) во множестве ребер R найдется по крайней мере одно ребро второй степени;
- 3) выполняется равенство $|X| = |R| + 1$.

В работе [14] дано формальное доказательство этой теоремы о необходимых условиях стягиваемости. Приведем здесь содержательные аргументы в пользу корректности этого важного для структурного анализа утверждения. Стягивание гиперграфа H в точку свидетельствует о справедливости условий 1 и 2. Рассмотрим подробнее условие 3. Действительно, для детали, которая монтируется первой, не нужны внутренние конструкторские базы, она устанавливается в сборочное приспособление. Для координации каждой следующей детали требуется только один комплект конструкторских баз. Отсюда следует, что число комплектов конструкторских баз на единицу меньше числа деталей изделия. Это простое соотношение можно эффективно использовать для идентификации многих структурных дефектов конструкции.

2. Перебазирование

Пусть для гиперграфа $H = (X, R)$ выполняется неравенство $|X| < |R| + 1$. Это соотношение свидетельствует о перебазировании технической системы. Перебазированием в инженерной практике называется ситуация избыточной координированности деталей. При сборке изделия данный феномен проявляется в том, что деталь приходится устанавливать, определяя ее положение по нескольким полным конструкторским базам. Для деталей, которые ведут себя как абсолютно твердые тела, монтаж с перебазированием невозможен, а деформируемые де-

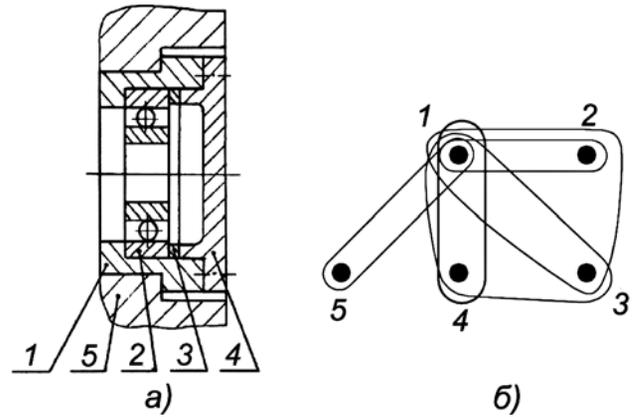


Рис. 3. Конструкция с перебазированием (а) и ее гиперграф H_1 (б)

тали просто не нуждаются в избыточных базах. Устранение перебазирования влечет за собой нарушение технологических инструкций и производственных расписаний. Оно может быть причиной критической потери точности и использования экстраординарных слесарных или пригоночных операций. В некоторых случаях для ликвидации перебазирования приходится вносить изменения в конструкцию изделия [15].

На рис. 3 показаны простая конструкция с перебазированием и ее гиперграф $H_1 = (X_1, R_1)$. Легко проверить, что $|X_1| = 5$, $|R_1| = 5$, т. е. $|X| < |R| + 1$. Согласно чертежу положение крышки 4 задают два полных комплекта конструкторских баз. Эта деталь может быть установлена на стакан 1, а также на стакан 1 вместе с шайбой 3.

Неравенство $|X| < |R| + 1$ дает легко вычисляемый критерий перебазирования для изделия и любого его фрагмента, который обладает свойством независимой сборки. Данный критерий можно использовать уже на самых ранних этапах проектирования, сразу после генерации трехмерной геометрической модели изделия.

Для проекта сложной технической системы, которая может состоять из нескольких тысяч деталей и комплектующих, важно не только установить факт перебазирования, но и локализовать конструктивный фрагмент, содержащий избыточные базы. Для этого можно использовать следующее утверждение, которое приведем здесь без доказательства [13].

Утверждение 1. Пусть для гиперграфа $H = (X, R)$ выполняется неравенство $|X| < |R| + 1$. Для локализации фрагмента конструкции с перебазированием требуется найти минимальный связный порожденный подграф $A = (X_A, R_A)$, для которого справедливо неравенство $|X_A| < |R_A| + 1$.

3. Несеквенциальные и некоординированные конструктивные фрагменты

Пусть для гиперграфа $H = (X, R)$, описывающего механическую структуру некоторого изделия, выполняется неравенство $|X| > |R| + 1$. Будем считать, что гиперграф является связным. Связность представляет собой фундаментальное свойство любого проекта, а ее нарушение — это грубая проектная ошибка, которая весьма редко встречается на практике.

Операция нормального стягивания обладает двумя очевидными свойствами:

- 1) сохраняет связность носителя;
- 2) уменьшает и число вершин и число ребер гиперграфа ровно на единицу.

Выполним все возможные нормальные стягивания в H . В результате получится связный нестягиваемый гиперграф, состоящий из ребер третьей и выше степеней. Вхождение в гиперграф нестягиваемых ребер высокой размерности свидетельствует о наличии в изделии несеквенциальных или некоординированных конструктивных фрагментов.

Пример простейшей несеквенциальной конструкции приведен на рис. 2. Некоординированность — это неполная пространственная определенность взаимного положения деталей, вызванная наличием нелегальных степеней свободы. Некоординированность, как правило, является следствием инженерных ошибок. Для иллюстрации этого свойства воспользуемся примерами двух простейших конструкций, которые изображены на рис. 4, а и рис. 4, в. Их гиперграфы показаны на рис. 4, б и рис. 4, г, соответственно. Полагаем, что горизонтальная степень свободы, которой располагают деталь 2 на рис. 4, а и детали 2, 3, 4 на рис. 4, в, — это проектные ошибки. Данные конструкции не обладают достаточной пространственной координацией, и для обоих гиперграфов выполняется неравенство $|X| > |R| + 1$.

4. Принцип "конструкторского близкодействия"

Анализ большого массива реальных конструкций показал, что координация минимальных геометрически определенных группировок деталей чаще всего выполняется за счет своих внутренних соединений и сопряжений, без использования внешних конструктивных элементов (деталей или комплектующих). Данное правило назовем "конструкторским близко-

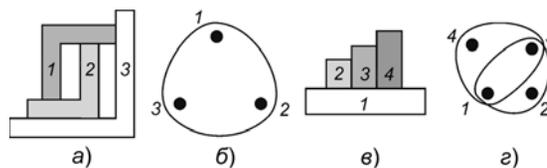


Рис. 4. Некоординированные конструкции (а, в) и их гиперграфы (б, г)

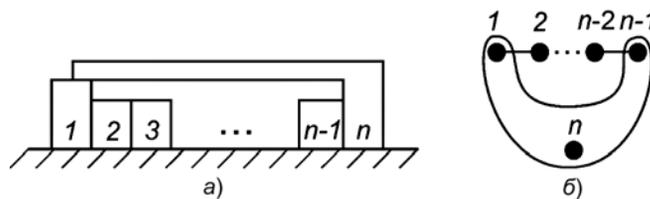


Рис. 5. Пример, в котором нарушен принцип "конструкторского близкодействия" (а), гиперграф данной конструкции (б)

действием". Его удобно выразить в точных графовых терминах.

Утверждение 2. Вершины, которые инцидентны ребру $r \in R$ гиперграфа $H = (X, R)$, образуют клику в графе связей $G = (X, V)$.

Проиллюстрируем данное утверждение на примере промежуточного вала, показанного на рис. 1, а. Легко проверить, что для каждого ребра гиперграфа данной конструкции (см. рис. 1, б) выполняется принцип "конструкторского близкодействия", т. е. все ребра гиперграфа представляют собой полные подграфы графа связей.

Конструктивные фрагменты, реализованные с нарушением данного принципа, — это потенциально опасные проектные решения. Они требуют особого внимания проектировщика. Для примера воспользуемся простой конструкцией, показанной на рис. 5.

Позиционирование детали n в горизонтальном направлении достигается опосредованно, через длинную цепочку, состоящую из $n - 1$ деталей. При сборке этой конструкции накопление погрешностей габаритных размеров, принадлежащих деталям с номерами от 1 до $n - 1$, может привести к потере связи между деталью 1 и левой опорой детали n и, следовательно, к неустойчивости последней.

Обобщая приведенный пример, можно утверждать, что базирование детали на две, три и т. д. детали, которые не имеют сопряжений друг с другом, а координируются опосредованно, через цепочку промежуточных объектов, является потенциально уязвимым проектным решением. Такие конструктивные фрагменты порождают длинные размерные цепи, что нарушает фундаментальный принцип технической подготовки производства, требующий

решать любую размерную задачу самым экономным способом (принцип кратчайшей размерной цепи [15]).

5. Оценка расчленяемости и собираемости

Разбиение изделия на отдельные сборочные единицы (СЕ), которые собираются независимо, — это важнейшее проектное решение, зависящее, в первую очередь, от структуры механических связей изделия. Достаточно сказать, что выбранная декомпозиция на СЕ влияет на многие экономические свойства производственной системы: стоимость, длительность производственного цикла и др. В некоторых случаях разбиение изделия и независимая сборка частей диктуется техническими причинами, например технологической однородностью сборочных операций. Понятно, что соединения, которые получаются с применением больших усилий или высоких температур, плохо совместимы с соединениями прецизионной точности.

Плохую расчленяемость можно уверенно считать структурным недостатком проекта. Такие изделия отличаются низкой адаптируемостью к сборке в различных производственных условиях и более высокими требованиями к качеству технологической оснастки.

Гиперграф $H = (X, R)$ позволяет идентифицировать плохую расчленяемость как самого изделия, так и любых составных частей. На рис. 6 изображены две структуры, которые не допускают разбиения на сборочные единицы.

Поскольку стягивание гиперграфа начинается со стягивания ребра второй степени, то необходимым условием плохой расчленяемости является "дефицит" таких ребер в s -гиперграфе, который описывает механическую структуру изделия. Так, "гиперцепь" (рис. 6, а) и "гиперзвезда" (рис. 6, б) включают в себя только одно ребро второй степени.

Еще одно важное конструктивное качество, которое зависит от структуры механических

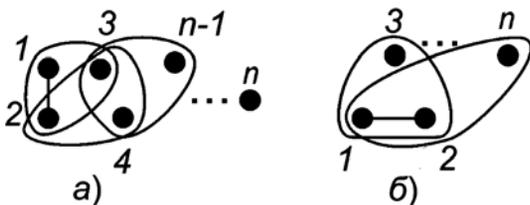


Рис. 6. Примеры структур, не допускающих разбиение на сборочные единицы:

а — "гиперцепь"; б — "гиперзвезда"

связей изделия, — собираемость. Это свойство означает способность технической системы собираться с помощью разных последовательностей. Если изделие имеет высокую собираемость, то ограничения по базированию не являются жесткими. Это значит, что последующий выбор рациональной последовательности сборки по технологическим или экономическим критериям будет располагать более богатым набором альтернатив.

Поясним сказанное примерами, изображенными на рис. 6. Структура "гиперцепь" допускает всего лишь две последовательности сборки, которые не противоречат условиям базирования. Это последовательности $1, 2, 3, \dots, n$ и $2, 1, 3, \dots, n$. Структура "гиперзвезда" обладает гораздо более высокой собираемостью. Ее можно собрать с помощью $2(n-2)!$ различных последовательностей. Данная структура имеет намного более высокую адаптируемость к различным технологическим системам и производственным условиям.

Очевидно, что s -гиперграф отличается высокой собираемостью, если он включает много висячих вершин, и этим свойством обладают его подграфы, полученные удалением таких вершин. Максимальной собираемостью среди всех s -гиперграфов обладает граф, который является деревом с диаметром 2. Число различных последовательностей, с помощью которых можно собрать данную структуру, равно $2(n-1)!$.

6. Структурная оценка сложности разборки

Многие виды технического обслуживания технической системы связаны с ее полной или неполной разборкой. Механическая структура изделия обязана обеспечивать простой демонтаж составных частей, которые необходимо часто извлекать для ремонта, контроля и иных регламентных работ.

Процесс разборки должен быть организован таким образом, чтобы извлекаемый фрагмент и оставшая часть конструкции сохраняли связность и внутреннюю геометрическую координацию. Приведем формальное описание этого условия.

Утверждение 3. Механические структуры частей разбираемого изделия должны описываться s -гиперграфами.

Сложность разборки можно оценить по минимальному числу деталей, которые нужно удалить из состава изделия для гарантированного доступа к искомому конструктивному элементу.

ту — детали или СЕ. Пусть, например, требуется демонтировать подшипник *b* из промежуточного вала (см. рис. 1, *a*). Гарантированный доступ к подшипнику дает снятие крышки 10 и гайки 7. Однако извлечение подшипника в этой ситуации нарушит связи и координацию конструктивного фрагмента, состоящего из деталей 8 и 9. Поэтому подшипник можно демонтировать после удаления этих деталей или целиком в сборе с деталями 8 и 9.

Введем формализацию, которая позволяет оценить "глубину вложения" деталей и получить количественные характеристики сложности сборки.

Определение 4. Пара $\{A, B\}$, $A, B \subseteq X$, называется разрезанием гиперграфа $H = (X, R)$, если $A \cup B = X$, $A \cap B = \emptyset$, и подграфы $[A]$, $[B]$, порожденные в H множествами вершин A , B , являются s -гиперграфами.

Данное определение постулирует свойство внутренней независимой координации обеих частей разрезания. Иными словами, корректным считается такая декомпозиция изделия, в которой обе части являются связными и геометрически определенными конструктивными фрагментами. Данное свойство должно выполняться для большинства операций жизненного цикла изделия: сборки, разборки, ремонта, разбиения на сборочные единицы, технического контроля и др.

Теорема 2. Разрезание $\{A, B\}$ s -гиперграфа $H = (X, R)$ достигается удалением только одного ребра $r \in R$.

Доказательство. Пусть $[A] = (A, R_A)$ и $[B] = (B, R_B)$ — подграфы, порожденные в H элементами разрезания $\{A, B\}$. Легко видеть, что гиперграф $[A \cup B]$, который порождается объединенным множеством вершин A и B , совпадает с $H = (X, R)$. Для $[A]$ и $[B]$ как s -гиперграфов выполняются соотношения $|A| = |R_A| + 1$ и $|B| = |R_B| + 1$. Сложим данные равенства и получим $|A| + |B| = |R_A| + |R_B| + 2$. Поскольку множества A и B образуют разбиение множества X , то справедливо соотношение $|A| + |B| = |X|$, поэтому $|X| = |R_A| + |R_B| + 2$.

По условиям теоремы H является s -гиперграфом, поэтому для него выполняется линейное соотношение $|X| = |R| + 1$. Из последних двух равенств следует, что $|R| + 1 = |R_A| + |R_B| + 2$ или $|R| = |R_A| + |R_B| + 1$. Число ребер в гиперграфе H на единицу больше общего числа ребер в гиперграфах $[A]$ и $[B]$. Это значит, что в H их соединяет только одно ребро. Теорема доказана.

Данная теорема дает простой способ генерации всевозможных разрезаний s -гиперграфа на

подграфы, которые описывают связные и координированные фрагменты конструкции. Поскольку каждая часть гиперграфа, полученная разрезанием, представляет собой s -гиперграф, то процедуру можно продолжить в глубину, вплоть до элементарных разрезаний, состоящих из отдельных вершин. Результаты рекурсивной процедуры разрезаний представим в виде И—ИЛИ-графа, в котором корневая вершина представляет исходный s -гиперграф, внутренние вершины соответствуют s -подграфам, полученным с помощью разрезаний, листья описывают вершины гиперграфа, а связи представляют сами разрезания.

В качестве примера воспользуемся конструкцией промежуточного вала (см. рис. 1, *a*). На рис. 7 показан И—ИЛИ-граф разрезаний гиперграфа H_s (см. рис. 1, *b*). На рис. 7 исходный гиперграф и его подграфы изображены черными кружками, а связи И—ИЛИ-графа — квадратами белого цвета.

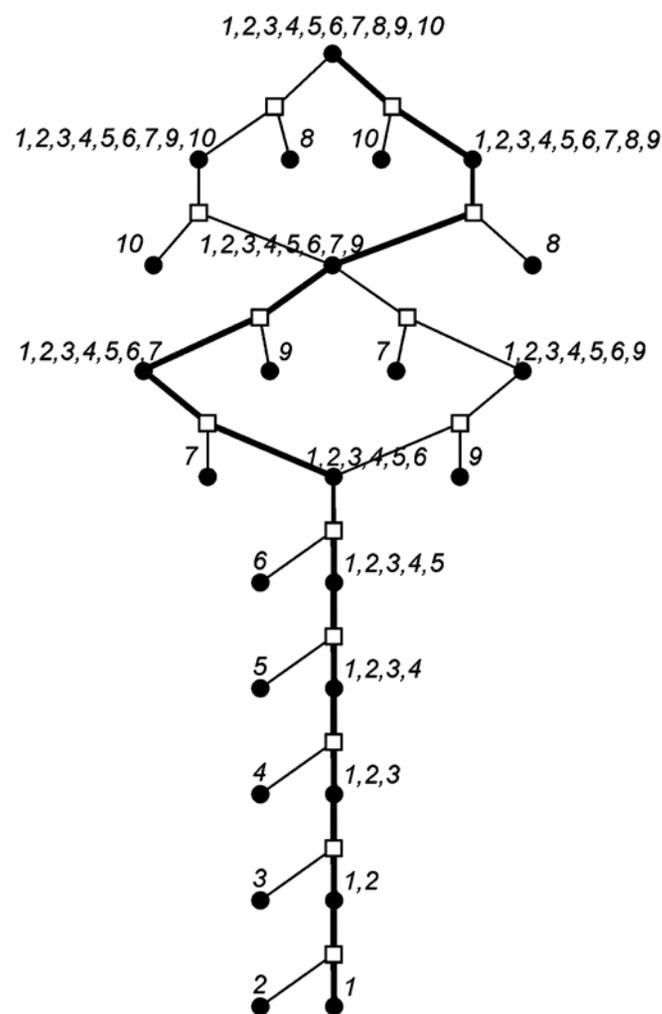


Рис. 7. И — ИЛИ-граф разрезаний гиперграфа H_s

Обозначим $T(H)$ И—ИЛИ-граф разрезов, построенный по s -гиперграфу $H = (X, R)$. Приведем несколько очевидных следствий, которые вытекают из процедуры построения этого объекта. Вершины дерева описывают связные и координированные подмножества деталей, которые обладают свойством независимой сборки (s -подграфы). Такими подмножествами являются СЕ, временные СЕ, узлы, состояния изделия в процессе сборки/разборки и любые конструктивные фрагменты, для которых требуется выполнить вспомогательные технологические операции (контроль, испытания, окраску, маркировку и пр.).

Каждое разрезание можно рассматривать как математическое описание операций разборки или секвенциальной сборки, что зависит от направления перемещения по вершинам и связкам графа $T(H)$. Любой путь, соединяющий висячую вершину графа $T(H)$ с корневой, представляет собой последовательность сборки изделия, каждая операция которой является секвенциальной и когерентной. Максимальные пути описывают последовательности сборки без разбиения на СЕ (линейные), не максимальные пути — с разбиением на СЕ (нелинейные). Путь, соединяющий корень графа $T(H)$ с висячей вершиной, представляет собой последовательность разборки: линейную или нелинейную. Например, выделенный путь на рис. 7 описывает линейную последовательность сборки 1, 2, 3, 4, 5, 6, 7, 9, 8, 10 или линейную последовательность разборки 10, 8, 9, 7, 6, 5, 4, 3, 2, 1.

Отметим некоторые очевидные свойства графа $T(H)$. Он имеет единственный наибольший элемент, который соответствует исходному гиперграфу H . Висячими вершинами в $T(H)$ служат вершины гиперграфа H .

Пусть A — некоторая вершина $T(H)$, которая описывает s -подграф A , $A \subseteq H$. Будем рассматривать $T(H)$ как обычный граф, не учитывая смысловые различия между вершинами и связками. Максимальную цепь в $T(H)$, которая соединяет вершину H с A , обозначим как (H, A) -цепь. Глубину вложения детали x будем оценивать длиной $l(x)$ кратчайшей (H, x) -цепи в графе $T(H)$. Так, глубина вложения деталей 8 и 10 равна двум, а детали 1 — восемнадцати (см. рис. 1, рис. 7).

Параметр $l(x)$ дает важную информацию о позиции, которую занимает деталь в структуре сложной технической системы. Для элементов изделия, требующих частого демонтажа, данный показатель должен принимать небольшие значения.

Выводы

1. Механическая структура сложной технической системы формируется на ранних этапах проектирования и оказывает существенное влияние на поведение системы в процессе сборки, разборки и разбиения на сборочные единицы.

2. Современные системы автоматизированного проектирования не располагают развитыми инструментами структурного анализа и идентификации структурных ошибок.

3. В актуальных исследованиях и разработках основной структурной моделью изделия служит граф связей, который не позволяет корректно описать отношение базирования и геометрическую координацию деталей в изделии.

4. Адекватной структурной моделью сложной технической системы является гиперграф. Он позволяет выполнить глубокий структурный анализ и идентифицировать многие типы распространенных структурных дефектов и ошибок на ранних этапах проектирования, сразу после формирования трехмерной геометрической модели изделия.

5. В работе предложены способы экспресс-анализа важных структурных характеристик: структурная избыточность, перебазирование, некогерентность, некоординированность, нарушение "конструкторского близкодействия". Разработан способ оценки сложности разборки изделия и "глубины вложения" детали в структуру изделия.

Список литературы

1. Bourjault A. Methodology of Assembly Automation: A New Approach // Robotics and Factories of the Future '87. 1988. P. 37—45. DOI: 10.1007/978-3-642-73890-6_6.
2. Cao Y., Kou X., Cao S. A sub-assembly identification algorithm for assembly sequence planning // International Industrial Informatics and Computer Engineering Conference. 2015. DOI: 10.2991/iicec-15.2015.127.
3. Ghandi S., Masehian El. Review and taxonomies of assembly and disassembly path planning problems and approaches // Computer-Aided Design. 2015. Vol. 67—68. P. 58—86. DOI: 10.1016/j.cad.2015.05.001.
4. Whitney D. E. Mechanical Assemblies: Their Design, Manufacture, and Role in Product Development. New York: Oxford University Press, 2004. 518 p.
5. Vignano R., Gomez G. Automatic assembly sequence exploration without precedence definition // International Journal on Interactive Design and Manufacturing. 2013. Vol. 7, Iss. 2. P. 79—89. DOI: 10.1007/s12008-012-0165-9.
6. Mathew A. T., Rao C. S. P. A novel method of using API to generate liaison relationships from an assembly // Software Engineering & Applications / 2010, N. 3. P. 167—175. DOI: 10.4236/jsea.2010.32021.

7. **Bahubalendruni M., Biswal B., Deepak B.** Computer aided assembly attributes retrieval methods for automated assembly sequence generation // *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, 2017. Vol. 11, N. 4. URL: scholar.waset.org/1999.8/10006876 (Дата обращения: 21.08.2017).

8. **Delchambre A.** Computer-aided Assembly Planning, 2012. 276 p. DOI: 10.1007/978-94-011-2322-8.

9. **Pan Ch.** Integrating CAD files and automatic assembly sequence planning. PhD thesis. Iowa State University, 2005. 107 p.

10. **Homem de Mello L., Sanderson A.** Planning repair sequences using the AND/OR graph representation of assembly plans // *Robotics and Automation. Proceedings 1988 IEEE International Conference*, 1988. Vol. 3. P. 1861—1862. DOI: 10.1109/ROBOT.1988.12341.

11. **Hermansson T., Bohlin R., Carlson J. S., Söderberg R.** Automatic assembly path planning for wiring harness installations //

Journal of Manufacturing Systems, 2013. Vol. 32 (3). P. 417—422. DOI: 10.1016/j.jmsy.2013.04.006.

12. **Hsu Y., Tai P., Wang M.** A knowledge-based engineering system for assembly sequence planning // *The International Journal of Advanced Manufacturing Technology*, 2011. Vol. 55, Iss. 5. P. 763—782. DOI: 10.1007/s00170-010-3093-5.

13. **Божко А. Н.** Моделирование механических связей изделия. Условия стягиваемости // *Наука и образование. МГТУ им. Н. Э. Баумана. Электрон. журн.* 2011. № 5. URL: <http://technomag.neicon.ru/doc/182518.html> (дата обращения 05.06.2016).

14. **Божко А. Н.** Моделирование позиционных связей в механических системах // *Информационные технологии*, 2012. № 10. С. 27—33.

15. **Базров Б. М., Таратынов О. В., Клепиков В. В.** *Технология сборки машин / Под общей ред. Б. М. Базрова. М.: Спектр, 2011. 368 с.*

A. N. Bozhko, Assistant Professor, e-mail: abozhko@inbox.ru,
Bauman Moscow State Technical University, Moscow, 105005, Russian Federation

Methods of Structural Analysis of Complex Products in Integrated CAD/CAM-systems

The problem of structural analysis of complex products in computer-aided design systems is considered. A hypergraphic mathematical model of the mechanical structure is proposed. It is shown that the connected and geometrically coordinated fragments of the product are described by contractible hypergraphs. This model can be used for deep structural analysis of the project. It allows you to detect many design errors and potentially dangerous solutions in the earliest stages of projecting. The criteria for identification of the overbasing, incoherence, loss of coordination, etc. are proposed. A method for estimating the "depth of access" of parts has been developed. which allows the complexity of disassembly and structural reparability of the product.

Keywords: assembly, mechanical structure, structural analysis, hypergraph, CAD-system, overbasing, geometric coordination, disassembly

DOI: 10.17587/it.24.499-506

References

1. **Bourjault A.** Methodology of Assembly Automation: A New Approach, *Robotics and Factories of the Future '87*, 1988, pp. 37—45. DOI: 10.1007/978-3-642-73890-6_6.

2. **Cao Y., Kou X., Cao S.** A sub-assembly identification algorithm for assembly sequence planning, *International Industrial Informatics and Computer Engineering Conference*, 2015. DOI: 10.2991/iiiiec-15.2015.127.

3. **Ghandi S., Masehian El.** Review and taxonomies of assembly and disassembly path planning problems and approaches, *Computer-Aided Design*, 2015, vol. 67—68, pp. 58—86. DOI: 10.1016/j.cad.2015.05.001.

4. **Whitney D. E.** *Mechanical Assemblies: Their Design, Manufacture, and Role in Product Development*, New York, Oxford University Press, 2004, 518 p.

5. **Vigano R., Gomez G.** Automatic assembly sequence exploration without precedence definition, *International Journal on Interactive Design and Manufacturing*, 2013, vol. 7, iss. 2, pp. 79—89. DOI: 10.1007/s12008-012-0165-9.

6. **Mathew A. T., Rao C. S. P.** A novel method of using API to generate liaison relationships from an assembly, *Software Engineering & Applications*, 2010, no. 3, pp. 167—175. DOI: 10.4236/jsea.2010.32021.

7. **Bahubalendruni M., Biswal B., Deepak B.** Computer aided assembly attributes retrieval methods for automated assembly sequence generation, *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, 2017, vol. 11, no. 4, available at: scholar.waset.org/1999.8/10006876 (date of access: 21.08.2017).

8. **Delchambre A.** Computer-aided Assembly Planning, 2012, 276 p. DOI: 10.1007/978-94-011-2322-8.

9. **Pan Ch.** Integrating CAD files and automatic assembly sequence planning, PhD thesis. Iowa State University, 2005, 107 p.

10. **Homem de Mello L., Sanderson A.** Planning repair sequences using the AND/OR graph representation of assembly plans, *Robotics and Automation. Proceedings 1988 IEEE International Conference*, 1988, vol. 3, pp. 1861—1862. DOI: 10.1109/ROBOT.1988.12341.

11. **Hermansson T., Bohlin R., Carlson J. S., Söderberg R.** Automatic assembly path planning for wiring harness installations, *Journal of Manufacturing Systems*, 2013, vol. 32 (3), pp. 417—422. DOI: 10.1016/j.jmsy.2013.04.006.

12. **Hsu Y., Tai P., Wang M.** A knowledge-based engineering system for assembly sequence planning, *The International Journal of Advanced Manufacturing Technology*, 2011, vol. 55, iss. 5, pp. 763—782. DOI: 10.1007/s00170-010-3093-5.

13. **Bozhko A. N.** Modelirovaniye mekhanicheskikh svyazey izdeliya (Modeling of mechanical connections of a product), *Nauka i obrazovaniye. MGTU im. N. E. Bauman. Elektron. zhurn.* 2011, no. 3, available at: <http://technomag.neicon.ru/doc/168373.html> (date of access: 03.03.015) (in Russian).

14. **Bozhko A. N.** Modelirovaniye pozitsionnykh svyazey v mekhanicheskikh sistemakh (Modeling of positional links in mechanical systems), *Informatsionnyye Tekhnologii*, 2012, no. 10, pp. 27—33 (in Russian).

15. **Bazrov B. M., Taratynov O. V., Klepikov V. V.** *Tekhnologiya sborki mashin* (Technology of assembling machines), Moscow, Spektr, 2011, 368 p. (in Russian).

В. Я. Цветков, д-р техн. наук, проф., зам. руководителя Центра перспективных фундаментальных и прикладных исследований,
Научно-исследовательский и проектно-конструкторский институт информатизации, автоматизации и связи на железнодорожном транспорте (НИИАС),
Москва, 109029, Россия, e-mail: cvj2@mail.ru

Алгоритмы как средство познания

Исследуется содержание алгоритмов первого и второго рода. Рассмотрено шесть типовых алгоритмов первого рода. Показано сходство и различие между алгоритмами. Определен критерий отнесения алгоритма к первому или второму роду. Показано, что главным разграничением алгоритмов являются когнитивные признаки обзорности и воспринимаемости. Дан анализ алгоритмов второго рода и определены методы их построения.

Ключевые слова: вычисление, познание, алгоритм, линейный алгоритм, прямой алгоритм, алгоритм первого рода, алгоритм второго рода, воспринимаемость задачи

Введение

Современное понятие алгоритма можно рассматривать не только как описание пути вычисления, но и как средства познания и передачи знаний. Опыт решения задач с применением алгоритмов показал, что по топологии решения можно выделить два типа задач и соответствующие им два типа алгоритмов. В одном типе алгоритмов, его называют алгоритм первого рода, решение удается получить от постановки задачи до окончательного решения за один проход или за один цикл. Этот тип алгоритма широко применяется в вычислительных системах.

В другом типе алгоритмов, его называют алгоритм второго рода, решение за один проход или за один цикл получить в принципе невозможно. Оно получается последовательно путем частичного решения задачи и последующим выбором дальнейшего частичного решения и затем выбором нового этапа и т. д. до получения окончательного решения, которое удовлетворяет начальным условиям.

Ярким примером задач и алгоритмов второго рода является шахматная партия. В начальном состоянии шахматной игры невозможно сказать, кто из соперников победит и чем закончится данная партия. Это невозмож-

но сказать наблюдателю со стороны и каждому из игроков. Когда игроки начинают делать первые ходы, ситуация проясняется для каждого из игроков и для стороннего наблюдателя. Каждый ход шахматной партии можно рассматривать как цикл решения задачи. Чем больше ходов, тем ближе окончание партии, т. е. решение данной проблемной ситуации. Значение результата шахматной игры является частным для каждого из игроков и общим для опыта игры и накопления знаний о методах игры. Точно также решение задач второго рода имеет частное и общее значение.

По существу, решение задач второго рода описывает решение сложных проблем и позволяет накапливать опыт в решении проблем и познания мира. Именно в таком аспекте рассматривается алгоритм в монографии академика Н. Моисеева "Алгоритмы развития" [1]. Н. Моисеев использует термин "алгоритм" в расширительном смысле, не как вычислительный механизм, а как средство решения сложных проблем и познания мира.

Другой относительно новый термин, используемый в настоящей работе, — это "экстернализация". Он означает преобразование неявного неформализованного знания в явное формализованное знание. Этот термин введен японским ученым Нонака, в его известной работе [2] (бо-

лее 22 000 цитирований), посвященной преобразованию неявного знания в явное знание. Идеи трансформации знания развиты в работе [3].

С позиций экстернализации знания алгоритм — это средство для решения такой задачи. Алгоритм представляет собой преобразование одной формы знания (условие задачи) в другую форму знаний (решение задачи). В своей работе Нонака описывает четырехступенчатый алгоритм, который решает четыре качественно разные задачи.

В обычных вычислениях алгоритм решает одну качественную задачу: он преобразует начальные условия в решение. При этом, как правило, это либо алгоритм только первого рода — прямое решение, либо алгоритм второго рода — поэтапное решение.

В четырехступенчатой схеме [2] Нонака описывает совместно алгоритмы первого и второго рода, хотя такие термины не употребляет. Альтернативой точке зрения автора работы [1] является рассмотрение алгоритма как технологической компоненты при обработке информации. Эта точка зрения отражена в известной работе Т. Кормена [4], выпущенной в 1990 г., многократно переизданной [5]. В данной работе заложены канонические основы построения только вычислительных алгоритмов первого рода. Первым базовым принципом построения алгоритмов по Т. Кормену представлен эвристический принцип "разделяй и властвуй". По сути дела, это алгоритм второго рода, использующий качественный анализ. То есть для построения алгоритма первого рода Т. Комен использует алгоритм второго рода.

По мнению автора данной статьи многие современные алгоритмы можно рассматривать также как сложные системы, поскольку они обладают признаками сложных систем. Алгоритм можно отнести к классу закрытых детерминированных сложных систем и рассматривать как функциональную систему, поскольку он всегда выполняет определенную функцию. Функциональное описание сложных систем исследовано М. Месаровичем [6], поэтому можно использовать предложенный им формализм для описания алгоритмов.

Алгоритмы имеют устойчивые связи и относятся к системам с постоянной структурой. Описание алгоритма можно рассматривать как явное знание, которое воспринимается разными людьми и может передаваться от одного человека к другому. Совокупность алгоритмов, решающих родственные задачи, дает возможность делать научное обобщение, которое вы-

ходит за рамки одного технического решения задачи, также как отдельная шахматная партия дает основание обобщать теорию игры в шахматы. Это дает основание связывать алгоритмы с получением знания.

Общеизвестно, что люди потенциально способны воспринимать примерно одинаковые объемы информации. Однако выводы, которые делают субъекты на основе одинаковых порций информации, получаются разными. Это создает противоречия и необходимость поиска компромиссов. В отличие от субъектов, алгоритмы на основе одинаковых порций информации получают одинаковые результаты. С этих позиций алгоритмы можно рассматривать как механизм, который способен найти компромиссы и разрешить противоречия.

Формализация алгоритмов первого рода

Построение алгоритмов связано с функциональными преобразованиями. Простейшее функциональное преобразование входной информации X в выходную информацию Y с помощью функции f имеет следующий вид:

$$Y = f(X) \quad (1)$$

или в форме описания сложной системы, даваемой Месаровичем [6]

$$A_1 : X_1 \rightarrow Y_1. \quad (2)$$

Выражение (2) интерпретируется так: алгоритм (процесс) A_1 осуществляет функциональное преобразование над входным множеством X_1 и преобразует его в промежуточное множество Y_1 . Выражений типа (2) может быть достаточно много для разных последовательных алгоритмических преобразований $A_1, A_2, A_3, \dots, A_i, \dots, A_n$. Все они могут быть обобщенно описаны выражением

$$A_S : X \rightarrow Y. \quad (3)$$

Здесь $A_S = f_A(A_1, A_2, A_3, \dots, A_i, \dots, A_n)$ — составной алгоритм. Выражение (3) интерпретируется так: составной алгоритм A_S осуществляет функциональное преобразование над входным множеством X и преобразует его в выходное множество Y . Если функция f_A — линейная, то алгоритм A_S является линейным. Независимо от вида функции f_A выражение (3) описывает прямой алгоритм.

Условно назовем этот алгоритм "номер один" Al_1 : для него существует описание

$$\exists Al_1(X) \rightarrow \{f(X) \rightarrow Y\}. \quad (4)$$

На рис. 1 приведены модифицированные схемы алгоритмов по отношению к (3). Алгоритм на схеме рис. 1, а называют циклическим. Его описание имеет следующий вид:

$$Al_2 : F((X, n, \delta)) \rightarrow Y(n, \delta), \quad (5)$$

где n — число циклов; δ — погрешность. Выражение (5) означает, что выходное множество $Y(n, \delta)$ получается с применением циклических вычислений и содержит погрешность δ . Недостатком циклического алгоритма является возможность закливания или зависания. Эти ситуации возникают, если алгоритм составлен некорректно или если происходит компьютерный сбой. Алгоритм на рис. 1, а имеет один вход и один выход. Н. Моисеев [1] отмечает, что живые формы имеют обратные связи, без которых невозможно обеспечить сохранение их гомеостазиса и развития. Поэтому циклические алгоритмы важны для моделирования развивающихся и саморазвивающихся систем.

Алгоритм на рис. 1, б называют стратифицированным [7]. Он содержит множество входов и выходов. Его отличие в том, что он создает промежуточные результаты L , которые могут быть использованы независимо или в совокупности. Его описание имеет вид

$$Al_3 = \Sigma Al_i, \quad (6)$$

где $i = 1, \dots, n$ — число слоев L . Выражение (6) означает, что общий алгоритм стратифицированной обработки Al_3 представляет собой сумму частных алгоритмов, каждый из которых имеет свой вход и выход, но в совокупности они обрабатывают общий исходный массив X , который представляет собой сумму непересекающихся множеств. Алгоритм получения отдельного слоя имеет вид

$$Al_i: F_i(X_i) \rightarrow L_i. \quad (7)$$

Выражение (7) означает, что алгоритм обработки отдельного слоя Al_i преобразует входное подмножество X_i в выходное подмножество L_i , которое представляет собой частный результат в общей системе слоев. Система слоев формируется как композиция

$$Y = \Sigma L_i + Con(i, j). \quad (8)$$

Выражение (8) означает, что результат обработки Y представляет собой совокупность слоев L_i и множество связей между ними $Con(i, j)$. Результат стратифицированного алгоритма формируется как сумма результатов, получаемых

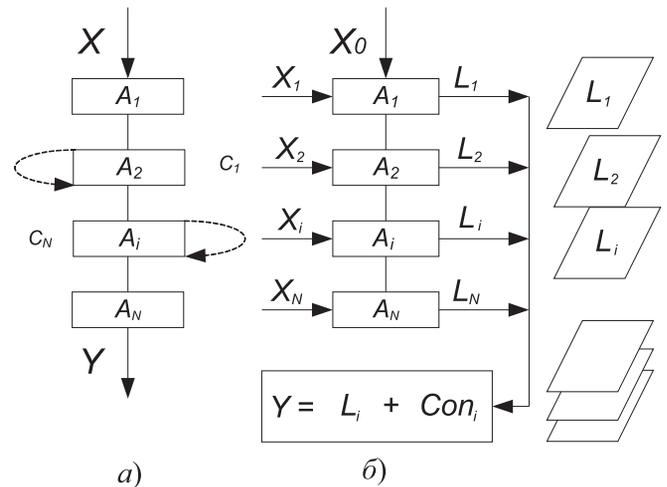


Рис. 1. Циклический (а) и стратифицированный (б) алгоритмы

на каждом слое. При этом сумма результатов может комбинироваться в любой последовательности и любой комбинации. Механизм стратификации применяют в геоинформационных системах [8] и в системах электронной картографии. Стратифицированный алгоритм в отличие от двух предыдущих алгоритмов формирует не окончательный результат, а набор стратов для получения нужного результата путем любой комбинации из данного набора. Стратификацию и стратифицированный алгоритм применяют при выполнении условий

$$X_i \cap X_k = \emptyset; Y_i \cap Y_k = \emptyset, \quad (9)$$

где $i \neq k$,

$$X_i \otimes X_k; Y_i \otimes Y_k, \quad (10)$$

где \otimes — произведение Декарта, $i \neq k$.

Альтернативой стратифицированному алгоритму является итеративный алгоритм [9]. Он имеет схожую структуру, но иной механизм, который формирует окончательный единственный результат. Он не требует выполнения условий (9, 10). На рис. 2 приведены схемы итеративного (рис. 2, а) и линейного инкрементного (рис. 2, б) алгоритмов. При их использовании задается некое начальное условие X_0 , базовое входное множество X_1 и совокупность входных множеств X_i . Это первое принципиальное отличие. В линейных алгоритмах входное множество одно. Итеративный алгоритм (рис. 2, а) имеет следующее описание:

$$Al_4 = \Sigma Al_i, \quad (11)$$

где $i = 1, \dots, n$ — число итераций;

$$Al_i: F_i(X_i) \rightarrow Y_i + d_i. \quad (12)$$

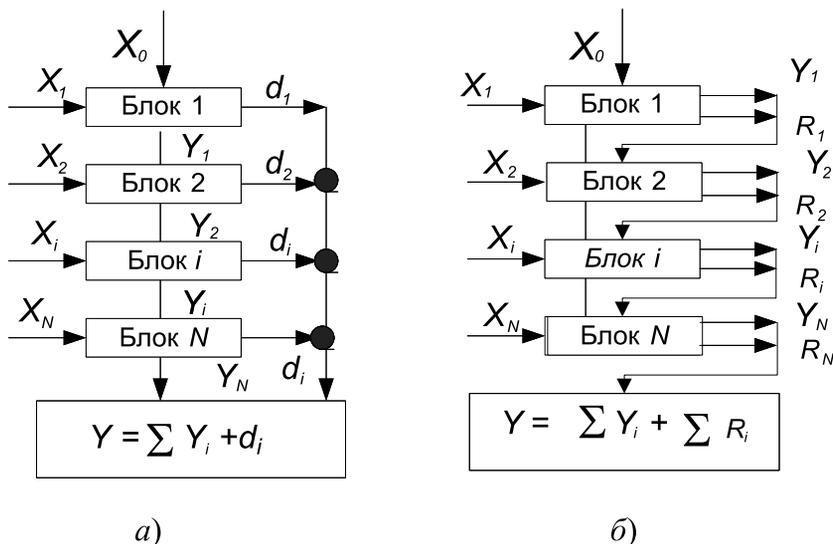


Рис. 2. Итеративный (а) и инкрементный алгоритмы (б)

Выражение (12) означает, что частный алгоритм обработки Al_i преобразует входное подмножество X_i в выходное подмножество Y_i , плюс итерацию d_i , которое представляет собой аддитивную часть в общем выходном множестве Y . Вычисления происходят по двум контурам: основному (Y_i) и вспомогательному (d_i). Результат обработки Y представляет собой аддитивную систему выходных множеств Y_i и итеративных добавок. Для итеративных алгоритмов существует проблема сходимости. Она состоит в том, что не при всяких начальных условиях X_0 результат вычислений должен сходиться:

$$\lim_{i \rightarrow \infty} (Y_i) = Y; \lim_{i \rightarrow \infty} (d_i) = 0. \quad (13)$$

Результат итеративного вычисления Y_i стремится к некоторому устойчивому значению, а значение поправки d_i стремится к нулю.



Рис. 3. Спиральная схема инкрементного алгоритма

Инкрементный алгоритм (рис. 2, б) имеет следующее описание [10]:

$$Al_5 = \sum Al_i, \quad (14)$$

где $i = 1, \dots, n$ — число инкрементных операций,

$$Al_i: F_i[X_i + R_{(i-1)}] \rightarrow Y_i + R_i. \quad (15)$$

Выражение (14) означает, что общий алгоритм инкрементной обработки Al_5 представляет собой сумму частных алгоритмов, как при стратифицированной (3) и итеративной (6) обработке. Результат инкрементной обработки представляет собой сумму частных решений и сумму полученных

ресурсов, которые не являются решением, но служат инструментом для возможного решения и анализа других задач:

$$Y = \sum (Y_i + R_i). \quad (16)$$

Выражение (16) означает, что при инкрементной обработке частный алгоритм обработки Al_i преобразует входное подмножество X_i в выходное подмножество Y_i и создает дополнительный ресурс R_i , который используют на последующем этапе обработки информации. Выражение (16) означает, что результат инкрементной обработки Y представляет собой аддитивное множество Y и плюс дополнительно некий информационный ресурс R , который можно использовать при решении других задач. Алгоритм на рис. 2, б называют линейным в силу того, что он последовательно решает задачу без возврата к предыдущим этапам. В отличие от уменьшающейся поправки при итеративном алгоритме ресурс при инкрементной обработке имеет тенденцию к росту:

$$\lim_{i \rightarrow \infty} (Y_i) = Y; \lim_{i \rightarrow \infty} (R_i) = R_0. \quad (17)$$

Существует модификация инкрементного алгоритма, которую называют "спиральной" или "циклической". Каждый цикл формирует очередную версию решения. Спиральный алгоритм приведен на рис. 3. Схема этого алгоритма показывает, что каждая версия приближает результат к области решения задачи. Решение формируется как функция в полярных координатах $Y = \phi(\varphi)$. Здесь Y — решение или результат обработки; ϕ — радиальная функция; φ — угловой аргумент, характеризующий стадию

(цикл) процесса обработки. При $\varphi = 2\pi$ происходит завершение цикла обработки и формирование очередной версии решения.

Инкрементный спиральный алгоритм имеет описание, аналогичное (14):

$$Al_6 = \Sigma Al_i. \quad (18)$$

Выражение (18) означает, что общий алгоритм обработки Al_6 представляет собой сумму частных алгоритмов, как при линейной инкрементной (14) обработке. Здесь $i = 1, \dots, n$ — число инкрементных операций. Введем понятие инкрементной функции:

$$\begin{aligned} \text{Incf}(Y_i, R_i) &= \\ &= \text{Incf}(Y_{i-1}, R_{i-1}) + dY_{i-1}(dR_{i-1}). \end{aligned} \quad (19)$$

Конкретный вид функции может быть разным. Смысл формулы (19) в том, что на каждой витке или цикле решения формируется новая версия решения, которая зависит от предыдущей версии и прироста ресурса, используемого на начальной стадии формирования данной версии. Используя системный подход [11, 12], можно описать спиральный инкрементный алгоритм IncA:

$$\begin{aligned} \text{IncA: Incf}(Y_i, R_i) &= \\ &= \text{Incf}(Y_{i-1}, R_{i-1}) + dY_{i-1}(dR_{i-1}), \end{aligned} \quad (20)$$

где Y_i — частные решения на каждом шаге; $i = 1, \dots, n$, n — число инкрементных спиралей; R_i — частный ресурс. Выражение (20) означает, что инкрементный алгоритм обработки преобразует входное подмножество (версию решения) Y_{i-1} в выходное подмножество Y_i (новую версию) и создает дополнительный ресурс R_i , который используют, если необходимо, в следующем цикле обработки. Дополнительно на каждом витке спирали формируется дополнительное условие решения задачи, которое меняет вектор решения. Обработка осуществляется с возвратом к линии контроля, но с получением дополнительных ресурсов и дополнительных условий.

Итоговое решение получают как

$$Y = \Sigma [Y_i + R_i] + \Sigma C_j, \quad (21)$$

где C_j — дополнительные условия.

Выражение (21) означает, что результат инкрементной обработки Y представляет собой аддитивное множество Y , в которое интегрирован инкрементный ресурс, на-

капливаемый при реализации каждой версии обработки, и дополнительные условия, которые могут дополнять или уточнять первоначальные условия решения задачи.

Таким образом, при инкрементном решении получают три результата: решение задачи (Y); дополнительный ресурс (R); уточненные (или дополненные) исходные условия решения задачи (C). Последнее подразумевает, что инкрементный алгоритм можно применять тогда, когда исходные условия нечетко сформулированы или являются некорректными. Это расширяет сферу применения алгоритма в область некорректно поставленных задач.

Алгоритм на рис. 3 называют спиральным в силу того, что он циклически решает задачу с возвратом к предыдущим этапам, но с наращиванием ресурса решения задачи и изменением (уточнением) исходных условий. Спираль раскручивается до тех пор, пока очередная версия не попадет в область решения. Алгоритм на рис. 3 обобщенно описывает генетические алгоритмы и мультиагентные системы [13]. Схема на рис. 3 является частным случаем фазового портрета линейной однородной системы:

$$x' = ax + by = \bar{\xi}_1; \quad y' = cx + dy = \bar{\xi}_2. \quad (22)$$

Здесь x', y' — производные. Одной из интегрирующих функций для системы (22) является функция

$$M(x, y) = -cx^2 + (a - d)xy + by^2. \quad (23)$$

Она представляет собой квадратичную форму относительно (x, y) . Не вдаваясь в детали, выделим две фазовые характеристики, связанные с векторными полями $\bar{\xi}_1, \bar{\xi}_2$. Они приведены на рис. 4.

Рис. 4, а соответствует схеме рис. 3 и описывает развитие инкрементного процесса, при

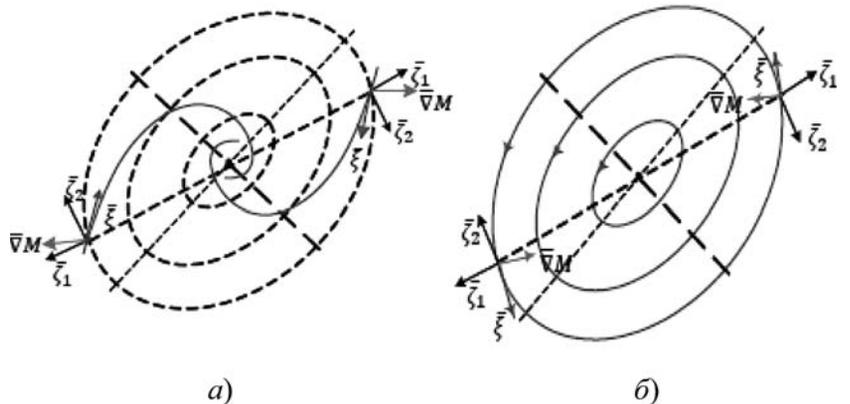


Рис. 4. Варианты развития инкрементного решения

котором рано или поздно спираль попадает в область решений. Рис. 4, б описывает состояние инкрементного процесса, при котором спираль трансформируется в эллипс, процесс решения заикливается и не попадает в область решений. Такая информационная ситуация обусловлена трансформацией инкрементных условий. Они могут как "разворачивать" спираль, так и "сворачивать" ее. Поэтому не во всех случаях применение инкрементного алгоритма приводит к получению решения.

Сравнение алгоритмов первого и второго рода

Рассмотренные алгоритмы относятся к алгоритмам первого рода. Алгоритм первого рода возможен при выполнении когнитивных условий: обзримости, воспринимаемости и интерпретируемости условий задачи. Он приведен на рис. 5.

В этом случае человек в силу своего естественного интеллекта способен воспринять информационную ситуацию, которая обладает свойствами обзримости, воспринимаемости и интерпретируемости. В этом случае субъект может применить один из алгоритмов, рассмотренных выше: $Al_1, Al_2, Al_3, Al_4, Al_5, Al_6$. Возможна ситуация, при которой возникают оппозиционные условия: "необозримость", "невоспринимаемость" [14, 15], "неинтерпретируемость". В этом случае ясна начальная ситуация, но неясен путь достижения цели и возникает сложность в выборе маршрута достижения цели. Важным фактором для алгоритма является то, что его можно рассматривать как систему, которая имеет собственные цели и располагает определенными ресурсами для их достижения. Если ресурсов недостаточно, то возникает задача второго рода [16] и алгоритм второго рода [17]. Это показано на рис. 6. Задачи второго рода характеризуются неоднозначностью решения и отсутствием единственного алгоритма, который приводит от начальной ситуации к конечной точке. Граф задачи второго рода имеет циклы. На начальной стадии возможны промежуточные решения Z .

В результате решения задачи второго рода возможны варианты решения. Решение может быть оптимальным Y_o . Решение может быть не оптимальным Y_{NO1} . Решение может быть ошибочным Y_{false} . Это обусловлено частичной воспринимаемостью условий задачи и возможными путями решения. В этом случае применяют эвристические методы поиска решений, в чис-

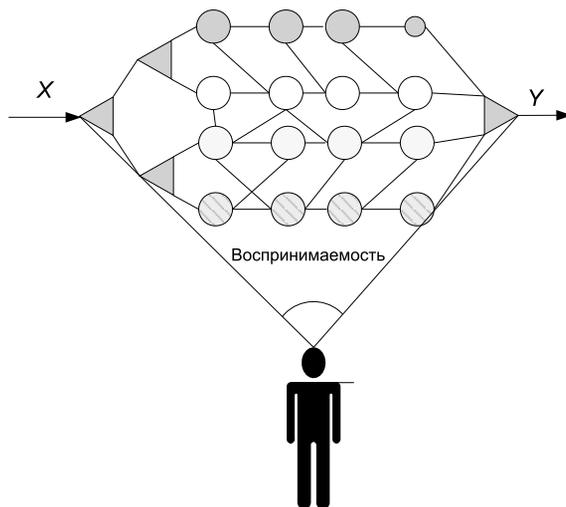


Рис. 5. Алгоритм первого рода

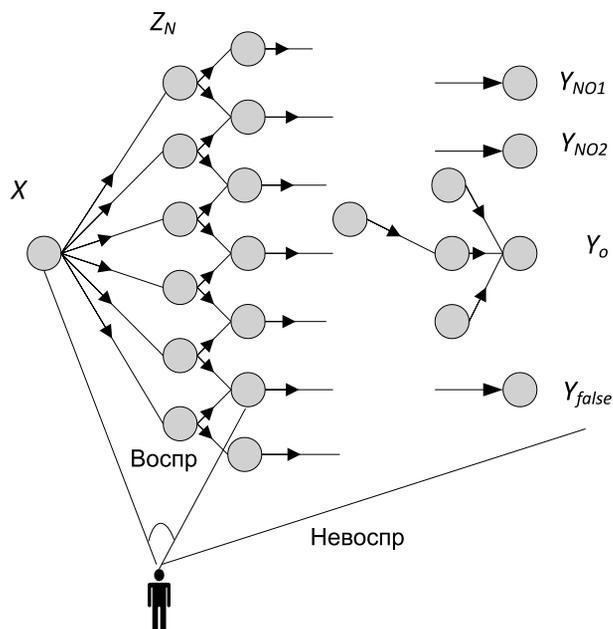


Рис. 6. Информационная ситуация возникновения алгоритма второго рода

ло которых входит также инкрементный алгоритм. Для задач первого рода инкрементный алгоритм характерен схемой на рис. 3, для задач второго рода инкрементный алгоритм имеет варианты на рис. 4.

Для формирования алгоритмов второго рода применяют кроме инкрементного подхода систему правил поиска, включая игровые методы, например метод Монте Карло [18]. Широко распространенным алгоритмом второго рода является муравьиный алгоритм [19]. Муравьиный алгоритм моделирует алгоритм оптимизации поведения муравьиной колонии (*ant colony optimization*) при поиске пищи. Он используется для нахождения приближенных решений транс-

портной задачи, а также для поиска маршрутов на графах. Отдаленно он напоминает "метод Монте Карло с условием". Алгоритм заключается в использовании системы правил модели группового поведения. В версии алгоритма Марко Дориго [20] осуществлялся поиск оптимального пути в графе на основе двух групп правил. Первая группа использовала модель "жадного алгоритма" [21], вторая группа дополняла его правилами "стадности" [22]. Правила "жадность" характеризуют локальную мотивацию, правила "стадность" характеризуют глобальную мотивацию и исключают "случайность". Передача информации создает возможность организации колонии муравьев в достижении цели. Такая модель используется в мультиагентных системах. В алгоритмах второго рода, основанных на множестве правил коллективного поведения, обратные связи возникают на основе установившихся (применяемых) правил коллективного поведения. Мультиагентная система [11] ищет оптимальное решение только на основе системы правил, находя его под конкретные начальные условия. В этом случае оптимальным решением является такое решение, на которое потрачено наименьшее количество ресурсов. Это опять связывает такой подход с инкрементным алгоритмом, в котором существует параметр "ресурсы".

Заключение

Проведенный анализ дает основание утверждать, что современное расширенное понятие алгоритма охватывает не только область вычислений, но и область рассуждений, и область решения сложных проблем. Расширенное понимание алгоритма связано в первую очередь с алгоритмами второго рода, которые представляют собой поэтапные методы решения сложных задач или проблем. Алгоритм первого рода или прямое решение позволяет исключить участие человека в процессе решения или рассуждения. Этим повышается скорость вычислений. Алгоритм второго рода или поэтапное решение проблемы требует включения человека или систем искусственного интеллекта в решение задачи или проблемы. Обобщение применения алгоритмов первого и второго рода позволяет получить новое знание.

Расширенное понимание алгоритма дает возможность привлечения методов построения алгоритмов из разных областей: биологии, генетики, искусственного интеллекта. Принципи-

альным является использование когнитивных характеристик воспринимаемости и обозримости при начальном формировании алгоритмов. При наличии воспринимаемости и обозримости существует возможность построения алгоритмов первого рода. При невоспринимаемости и необозримости требуется использовать алгоритмы второго рода. При недостаточности ресурсов для достижения цели также необходимо применять алгоритмы второго рода. Некоторые виды алгоритмов содержат познавательные функции. Стратифицированные алгоритмы и спиральные (инкрементные) алгоритмы можно использовать для решения задач второго рода с некоторыми условностями. Стратифицированный алгоритм приводит только к разбиению начального множества без установления полной системы связей между слоями. Современное развитие методов вычислений и алгоритмов требует исследований алгоритмов второго рода. Важной поддержкой этого направления может служить системная инженерия. Перспективным следует считать развитие методов когнитивного формирования алгоритмов. Онтологический инжиниринг также может служить основой современного построения алгоритмов первого и второго рода.

Список литературы

1. **Моисеев Н.** Алгоритмы развития. М.: Наука, 1987. 304 с. (Переиздана в 2017 издательством Litres).
2. **Nonaka I.** A dynamic theory of organizational knowledge creation // *Organization Science*. 1994. Vol. 1. N. 5. P. 14–37.
3. **Sigov A. S., Tsvetkov V. Ya.** Tacit Knowledge: Oppositional Logical Analysis and Typologization // *Herald of the Russian Academy of Sciences*. 2015. Vol. 85, N. 5. P. 429–433. DOI: 10.1134/S1019331615040073.
4. **Cormen T. H., Leiserson Ch. E., Rivest R. L.** Introduction to Algorithms. 1st. MIT Press and McGraw-Hill, 1990. 863 p.
5. **Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К.** Алгоритмы. Построение и анализ: пер. с англ. М.: Вильямс, 2009. 1296 с.
6. **Месарович М., Такахара Н.** Общая теория систем: математические основы. М.: Мир, 1978. 311 с.
7. **Буравцев А. В.** Стратифицированный метод построения сложной системы // *Образовательные ресурсы и технологии*. 2017. № 3 (20). С. 23–32.
8. **Цветков В. Я.** Геоинформационные системы и технологии. М.: Финансы и статистика, 1998. 288 с.
9. **Wang Z., Han B.** Advanced iterative algorithm for phase extraction of randomly phase-shifted interferograms // *Optics letters*. 2004. Vol. 29. N. 14. P. 1671–1673.
10. **Цветков В. Я., Железняков В. А.** Инкрементальный метод проектирования электронных карт // *Инженерные изыскания*. 2011. № 1. С. 66–68.
11. **Кудж С. А.** Системный подход // *Славянский форум*. 2014. № 1 (5). С. 252–257.
12. **Цветков В. Я.** Решение проблем с использованием системного анализа // *Перспективы науки и образования*. 2015. № 1. С. 50–55.

13. **Розенберг И. Н., Цветков В. Я.** Применение мультиагентных систем в интеллектуальных логистических системах // *Международный журнал экспериментального образования*. 2012. № 6. С. 107—109.

14. **Буравцев А. В.** Серый управленческий анализ // *Перспективы науки и образования*. 2017. № 5 (29). С. 74—79.

15. **Цветков В. Я.** Когнитивные аспекты построения виртуальных образовательных моделей // *Перспективы науки и образования*. 2013. № 3. С. 38—46.

16. **Tsvetkov V. Ya.** Incremental Solution of the Second Kind Problem on the Example of Living System // *Biosciences biotechnology research Asia*. November 2014. Vol. 11 (Spl. Edn.). P. 177—180.

17. **Doha E. H., Abd-Elhameed W. M., Youssri Y. H.** Second kind Chebyshev operational matrix algorithm for solving differential

equations of Lane—Emden type // *New Astronomy*. 2013. Vol. 23. P. 113—117.

18. **Robert C. P.** Monte Carlo methods. John Wiley & Sons, Ltd, 2004.

19. **Bautista J., Pereira J.** Ant algorithms for assembly line balancing // *Ant algorithms*. 2002. P. 49—61.

20. **Dorigo M., Di Caro G., Gambardella L. M.** Ant algorithms for discrete optimization // *Artificial life*. 1999. Vol. 5. N. 2. P. 137—172.

21. **DeVore R. A., Temlyakov V. N.** Some remarks on greedy algorithms // *Advances in computational Mathematics*. 1996. Vol. 5. N. 1. P. 173—187.

22. **Wang Gai-Ge, Guo Lihong, Gandomi Amir H.** et al. Chaotic krill herd algorithm // *Information Sciences*. 2014. N. 274. P. 17—34.

V. Ya. Tsvetkov, D. Sc., Professor, Deputy Head,
Center Fundamental and Advanced Research, Research and Design Institute of Design Information,
Automation and Communication on Railway Transport, SC NIIAS,
Moscow, 109029, Russia, e-mail: cvj2@mail.ru

The Algorithms as a Means of Knowledge

The article explores algorithms as a means of cognition. It is shown that the algorithm is not only a means of calculation, but a means of storing knowledge. Algorithm is also a means of transferring knowledge. The article claims that the algorithm is a means of transforming knowledge. The connection between the algorithm and the model of transformation of implicit knowledge is shown. The article claims that many complex algorithms can be considered as complex systems. This makes it possible to apply system analysis to the analysis of algorithms.

The paper describes two classes of problems and two classes of algorithms based on the receipt of a solution. Algorithms of the first kind and algorithms of the second kind. Algorithms of the first kind allow one to obtain a direct solution of the problem in one step. Algorithms of the second kind solve the problem stage after stage. The article shows the difference between linear and nonlinear algorithms. The article introduces a formal description of algorithms based on system analysis. The article analyzes six typical algorithms of the first kind. An evolutionary relationship between algorithms of the first kind is shown. The article notes the similarity and difference between these algorithms. The article gives a criterion for assigning an algorithm to the first or second kind. The article contains structural diagrams of typical algorithms of the first kind. For each structural scheme, a formal description of the algorithm is given. The difference between the iterative algorithm and the incremental algorithm is shown. The article analyzes the reasons for the emergence of an algorithm of the second kind. It is shown that the main criterion for the appearance of algorithms of the second kind are cognitive signs of foreseeability and perceptibility. The article analyzes algorithms of the second kind and determines the methods of their construction. The methods of solution for the occurrence of problems of the second kind are described. The possibility of using multi-agent systems for solving problems of the second kind.

Keywords: calculation, cognition, algorithm, linear algorithm, direct algorithm, algorithm of the first kind, algorithm of the second kind, perceivability of the problem

DOI: 10.17587/it.24.507-515

References

1. **Moiseev N.** *Algoritmy razvitija*, Moscow, Nauka, 1987, 304 p. Pereizdana 2017, Izdatel'stvom Litres (in Russian).
2. **Nonaka I.** A dynamic theory of organizational knowledge creation. *Organization Science*, 1994, no. 5 (1), pp. 14—37.
3. **Sigov A. S., Tsvetkov V. Ya.** Tacit Knowledge: Oppositional Logical Analysis and Typologization, *Herald of the Russian Academy of Sciences*, 2015, vol. 85, no. 5, pp. 429—433. DOI: 10.1134/S1019331615040073.
4. **Cormen T. H., Leiserson C. E., Rivest R. L.** *Introduction to Algorithms*. 1st. MIT Press and McGraw-Hill, 1990. 863 p.
5. **Kormen T., Leiserson Ch., Rivest R.** et al. *Algoritmy. Postroenie i analiz*: [per. s angl.], Moscow, Vil'jams, 2009. 1296 p. (in Russian).
6. **Mesarovich M., Takahara N.** *Obshhaja teorija sistem: matematicheskie osnovy*, Moscow, Mir, 1978, 311 p. (in Russian).
7. **Buravcev A. V.** Stratificirovannyj metod postroenija slozhnoj sistemy, *Obrazovatel'nye resursy i tehnologii*, 2017, no. 3 (20), pp. 23—32 (in Russian).
8. **Tsvetkov V. Ya.** Geoinformacionnye sistemy i tehnologii, Moscow, Finansy i statistika, 1998, 288 p. (in Russian).
9. **Wang Z., Han B.** Advanced iterative algorithm for phase extraction of randomly phase-shifted interferograms, *Optics letters*, 2004, vol. 29, no. 14, pp. 1671—1673.
10. **Tsvetkov V. Ya., Zheleznyakov V. A.** Inkremental'nyj metod proektirovanija jelektronnyh kart, *Inzhenernye izyskanija*, 2011, no. 1, pp. 66—68 (in Russian).
11. **Kudzh S. A.** Sistemnyj podhod, *Slavjanskij forum*, 2014, no. 1 (5), pp. 252—257 (in Russian).
12. **Tsvetkov V. Ya.** Reshenie problem s ispol'zovaniem sistemnogo analiza, *Perspektivy nauki i obrazovanija*, 2015, no. 1, pp. 50—55 (in Russian).
13. **Rozenberg I. N., Tsvetkov V. Ya.** *Primenenie mul'tiagentnyh sistem v intellektual'nyh logisticheskijh sistemah*, *Mezhdunarodnyj*

zhurnal jeksperimental'nogo obrazovaniya, 2012, no. 6, pp. 107–109 (in Russian).

14. **Buravcev A. V.** Seryj upravlencheskij analiz, *Perspektivy nauki i obrazovaniya*, 2017, no. 5 (29), pp. 74–79 (in Russian).

15. **Tsvetkov V. Ya.** Kognitivnye aspekty postroeniya virtual'nyh obrazovatel'nyh modelej, *Perspektivy nauki i obrazovaniya*, 2013, no. 3, pp. 38–46 (in Russian).

16. **Tsvetkov V. Ya.** Incremental Solution of the Second Kind Problem on the Example of Living System, *Biosciences biotechnology research Asia*, November 2014, vol. 11 (Spl. Edn.), pp. 177–180.

17. **Doha E. H., Abd-Elhameed W. M., Youssri Y. H.** Second kind Chebyshev operational matrix algorithm for solving differential equations of Lane–Emden type, *New Astronomy*, 2013, vol. 23, pp. 113–117.

18. **Robert C. P.** Monte Carlo methods, John Wiley & Sons, Ltd, 2004.

19. **Bautista J., Pereira J.** Ant algorithms for assembly line balancing, *Ant algorithms*, 2002, pp. 49–61.

20. **Dorigo M., Di Caro G., Gambardella L. M.** Ant algorithms for discrete optimization, *Artificial life*, 1999, vol. 5, no. 2, pp. 137–172.

21. **DeVore R. A., Temlyakov V. N.** Some remarks on greedy algorithms, *Advances in computational Mathematics*, 1996, vol. 5, no. 1, pp. 173–187.

22. **Wang Gai-Ge, Guo Lihong, Gandomi Amir H.** et al. Chaotic krill herd algorithm, *Information Sciences*, 2014, vol. 274, pp. 17–34.

УДК 004.912

DOI: 10.17587/it.24.515-523

Ю. С. Шучалова, приглашенный преподаватель, iusshuchalova@hse.ru,

В. В. Ланин, ст. преподаватель, vlanin@hse.ru,

Национальный исследовательский университет "Высшая школа экономики", г. Пермь

Исследовательский портал для анализа и оценки стиля научных публикаций

Описан этап проектирования портала для проведения корпусных исследований английского языка. Сформулированы требования к решению, показаны лингвистические подходы к решению поставленных задач. Приведен процесс моделирования системы и рассмотрены особенности реализации с учетом специфики предметной области. Для интеграции гетерогенных компонентов предложена сервисная архитектура.

Ключевые слова: корпусные исследования, исследовательский портал, академический английский язык, моделирование информационных систем

Введение

Широкое распространение задач и инструментов, связанных с информационными технологиями, в науках о естественных языках позволило лингвистам наряду с традиционными методами развивать такое направление, как корпусная лингвистика. Данное направление основано на использовании так называемых корпусов, под которыми понимается представительная коллекция текстов по определенной тематике, доступная в электронной форме [1], т. е. набор документов или записей, собранных по определенному принципу: общий стиль, общий автор, общий жанр и т. д. Таким образом, в отличие от классической лингвистики, корпусная лингвистика ориентирована на изучение практического применения языка, а не его теоретических аспектов. Многие традиционные задачи языкознания решаются с помощью текстовых корпусов. Например, изучение отдельных словоупотреблений, общей динамики языка, машинный перевод и

обучение языку (*computer-assisted language learning*), некоторые виды анализа текста. В целом поиск новых возможностей применения корпусов в различных областях — одна из основных задач корпусной лингвистики.

Корпусная лингвистика [2] тесно связана с информационными технологиями. Сбор и обработка обширных корпусов требуют использования программного обеспечения и технологий, допускающих совместную работу множества удаленных пользователей. Программное обеспечение, предназначенное для работы лингвистов, как правило, можно разделить на общее — для выполнения базовых функций (например, автоматической морфологической разметки) — и специализированное, созданное для решения конкретных задач. Одной из таких задач является проверка соответствия текста академическому стилю. При написании научных статей на английском языке студенты сталкиваются с необходимостью использования стандартных языковых средств для соблюдения академической стилистики текста.

В то же время должен быть соблюден некий баланс в количестве этих языковых средств, чтобы не нарушалась читаемость текста. Сложность состоит в том, что не существует общепринятых стандартов, которых можно придерживаться при написании статьи на английском языке, так как стилистика и степень читаемости текста оцениваются экспертами.

Существующие инструменты для рецензирования и методы анализа стиля [3] при оценке качества и читаемости опираются на ограниченное число признаков (терминология, формальный стиль речи, читаемость текста), в то время как в методической литературе и научных статьях представлено гораздо больше характеристик академического стиля. Кроме того, изначально сформулированные правила определения качества текста необязательно могут отражать реальную ситуацию использования различных признаков научного стиля на практике. Именно такую возможность предоставляют методы корпусной лингвистики.

Таким образом, существует потребность в создании приложения, которое имело бы возможность оценивать научные статьи, написанные носителями на английском языке и одобренные к публикации влиятельными рецензируемыми изданиями (например, Springer), по некоторым лингвистическим характеристикам, которые в основном определяют академический стиль письменной речи. Усредненные значения встречаемости значимых характеристик, определенные в ходе исследования корпуса статей по определенной тематике, смогут служить ориентиром при написании собственного материала.

Для проверки значимости различных лингвистических характеристик при оценке качества стиля и читаемости текста будет проводиться исследование, основанное на сравнении встречаемости маркеров стиля в научных статьях, взятых в качестве эталонных, и учебных работ высокого и низкого качества (по мнению эксперта). Результаты исследования могут быть использованы для частичной автоматизации работы эксперта при проверке соответствия статьи стилистическим требованиям, а также для формирования рекомендаций. Наиболее подходящей для реализации метода анализа и оценки стиля научных текстов на английском языке с помощью эталонных корпусов, по мнению авторов, является форма исследовательского портала.

Целью работы является проектирование архитектуры портала для анализа и оценки стиля

научных публикаций с указанием необходимых свойств системы и возможных средств ее реализации. Идентификация функционального стиля текста, в том числе научного (академического), — одна из базовых задач лингвистики в целом [4]. Использование методов корпусной лингвистики, основанных на анализе коллекций текстов, позволяет рассмотреть не только теоретические признаки научных текстов, но и особенности их применения на практике.

Формирование требований к решению

Для реализации портала необходимо формализовать требования к функциональности портала, которые основываются на теоретических аспектах метода. В рассматриваемом методе оценка качества стиля статьи проводится на основе сравнения с корпусом научных статей, написанных носителями на английском языке и признанных качественными (прошедших рецензирование экспертами), по некоторым лингвистическим характеристикам, которые в основном определяют академический стиль письменной речи. Эти характеристики далее в тексте будем называть качественными критериями или маркерами академического стиля.

В качестве теоретической базы для создания приложения экспертом был предоставлен список качественных критериев ("маркеров") академического стиля речи. Список составлен на основе справочных и учебных материалов, а также интернет-ресурсов по обучению академическому письму.

Все вошедшие в список критерии можно разделить на три группы: лексические, грамматические и синтаксические. Внутри групп также возможно разделение критериев, например, по особенностям их проверки. Рассмотрим критерии, входящие в каждую группу, подробнее.

Лексические критерии можно условно разделить на три подгруппы:

- критерии частотности появления в тексте конкретных слов, терминологии;
- критерии частотности появления в тексте слов, соответствующих определенным словообразовательным схемам;
- критерии частотности появления в тексте слов определенных частей речи.

К первой подгруппе относятся следующие критерии:

- активное использование терминологии, соответствующей предметной области;

- близкая к нулевой встречаемость личных местоимений;
- десемантизированные глаголы;
- глаголы широкой абстрактной семантики;
- усилительные наречия.

Вторая группа включает в себя такие маркеры, как:

- наличие абстрактных существительных, образованных с помощью суффиксов;
- наличие суффиксов, обозначающих термины и технические понятия.

Наконец, к третьей группе мы отнесем следующие критерии:

- номинативность текста — преобладание существительных;
- предположительно низкая встречаемость личных местоимений.

В группу грамматических маркеров входят два критерия:

- широкое использование глаголов в пассивном залоге;
- преобладание глаголов настоящего времени.

Синтаксические критерии, как и лексические, можно разделить на подгруппы:

- критерии, описываемые структурами;
- критерии, учитывающие встречаемость определенных союзов, предлогов, средств связи и др.

В первую подгруппу входят следующие критерии:

- преобладание предложений с простой, сложноподчиненной или сложносочиненной структурой;
- наличие постпозитивных и препозитивных определений;
- преобладание препозитивных определительных групп.

Во второй группе выделяются следующие маркеры:

- использование двойных и составных союзов;
- использование слов, являющихся в литературном языке архаизмами;
- составные предлоги;
- средства логической связи.

Предполагается, что числовые оценки значений описанных выше характеристик для определенной предметной области возможно получить, проведя анализ корпуса текстов, состоящего из статей по соответствующей тематике. Разрабатываемый сервис для анализа и оценки стиля должен получать на вход корпус текстов, состоящий из научных статей на английском языке, посвященных сходной тематике. Последнее уточнение нужно для более

удобного выделения терминов в тексте, например, с использованием терминологического словаря на заданную тематику.

После автоматической токенизации текста и нанесения разметки на слова и конструкции, которые описаны в качественных критериях академической речи, приложение предоставляет пользователю возможность проверить и редактировать автоматически нанесенную разметку.

Проанализировав аннотированный корпус, приложение выдает по каждому из критериев, представленных выше, статистическую информацию, например:

- среднее число маркеров данного типа, встречающееся в документах корпуса;
- наибольшее и наименьшее количественные значения характеристики;
- опционально: распределение числа маркеров в каждом документе;
- для критериев, оценивающих встречаемость частей речи, типов предложений и др. процентное соотношение относительно общего числа слов/предложений.

Должен учитываться тот факт, что размер статей может быть различным, поэтому статистическую информацию необходимо отражать как в абсолютных числах, так и относительно общего числа слов в документе. Полученная в результате работы сервиса информация может быть использована экспертом для интерпретации и проверки ее валидности.

Для формирования рекомендаций сервис должен получить отдельную статью, разметить ее по тем же типам аннотации (маркерам стиля), которые оцениваются в корпусе, и сравнить расстояния между значениями статистических характеристик корпуса и статьи по некоторой метрике.

Кроме функциональных требований, выделяются также нефункциональные, касающиеся особенностей целевой аудитории портала и особенностей использования.

Академический стиль речи и методы его идентификации и анализа

Научный (академический) стиль речи представляет научную сферу общения и речевой деятельности, связанную с реализацией науки как формы общественного сознания [6]. Иными словами, это особый функциональный стиль речи, который в широком смысле используют для сообщения нового знания

о действительности и доказательства ее истинности. Этот стиль используют в научных статьях, учебной литературе, монографиях и т. д.

В настоящее время у людей, занимающихся научными исследованиями, нередко возникает необходимость освоения академического стиля английского языка, для того чтобы их научные работы и статьи могли быть признаны международным академическим сообществом. Для помощи в изучении особенностей данного стиля создаются справочные и учебные материалы, а также обучающие интернет-ресурсы. Существует множество учебных пособий на английском языке, предназначенных для обучения академическому письму. Методические пособия могут быть адресованы исследователям и студентам [21], преподавателям [12, 14, 17, 23, 26], редакторам изданий [15, 25]. Имеются также учебные пособия от русскоязычных авторов [8–10].

Кроме того, доступны интернет-ресурсы, содержащие рекомендации к написанию научных текстов. В частности ресурсы, называемые *academic phrasebanks*, содержат списки конструкций, которые могут быть использованы в тех или иных ситуациях: когда нужно выразить критический взгляд, обозначить дистанцию между приводимым мнением и мнением автора, описать классификацию, привести результаты сравнения и примеры, ввести понятие, и т. д. Пример подобного ресурса — *Academic Phrasebank (University of Manchester)* [11].

Особенностям академического стиля английского языка были посвящены многие научные работы, в том числе русскоязычные диссертации и исследования как 80-х годов прошлого века, так и относительно недавнего времени. В исследованиях охвачены как общие структурные и функциональные особенности научных текстов [4], так и узкие темы, например, выражение экспрессивности [7], причинно-следственных отношений и др.

Классификация текстов, в том числе по функциональным стилям (разговорный, научный, художественный и др.) — одна из задач обработки естественного языка. Определение функционального стиля текста можно использовать в информационном поиске, машинном переводе, генерации текстов [5] для получения более точных и удовлетворяющих пользователей результатов.

В работе [5] представлены методы машинного обучения для классификации текстов, в которых использованы в том числе описанные выше признаки, разделенные авторами на лек-

сические (число слов, N-грамм, глубина дерева синтаксического разбора и т. д.) и количественные (число символов, число слогов и т. д.). Классифицируемые документы сопоставляют с векторами признаков.

Подход к анализу академического стиля был представлен в работе [20]. Выделяются признаки текста, отвечающие за формальный стиль речи (пассивный залог, субъективные выражения, вопросы), читаемость (союзы и другие соединяющие фразы, использование существительных вместо глаголов) и научный язык (по списку из 200 научных слов и др.), и на основе выделенных признаков создается самоорганизующаяся карта (*Self-Organizing Map* — особая разновидность нейронной сети).

Еще один метод был использован для анализа характеристик в заголовках статей по теме "Computer Science". Значения параметров, таких как длина заголовка, использование пунктуации и предлогов, частота слов, были исследованы на примере корпуса статей из научных журналов [12].

Метод, рассматриваемый в данной работе, предполагает сравнение статей, стиль которых признан экспертами качественным, со статьями, которые, несмотря на соблюдение рекомендаций из методической литературы, являются плохо читаемыми и не соответствуют стандартам. Сравнение проводят по признакам, связанным с лексическими и синтаксическими характеристиками текста, которые были выделены при анализе учебных пособий и научных работ об особенностях академического стиля речи. В качестве опоры для исследования берется коллекция "эталонных" статей из рецензируемых источников, каждой из которых ставится в соответствие вектор, описывающий встречаемость тех или иных лингвистических характеристик. В дальнейшем, при сравнении эталонных статей с работами студентов будет исследоваться влияние каждой из этих характеристик на оценку качества и читаемости текста.

Особенностью этого метода является ориентация на практическое применение теоретических рекомендаций по соблюдению академического стиля. Проведение исследований на коллекции текстов, собранных по общим признакам — в данном случае по языку и стилю, — относит метод к области корпусной лингвистики.

Существуют различные инструменты для работы с корпусами текстов — от обычных сайтов для простого просмотра и поиска данных до систем, позволяющих создавать собственные приложения. Некоторые из этих

инструментов могут быть использованы при реализации метода анализа и получения оценки соответствия статей академическому стилю английского языка, описанного в работе [22].

Моделирование портала

На рис. 1 (см. вторую сторону обложки) представлен порядок действий пользователя на портале.

Под выбором сценария подразумевается выбор процесса анализа корпуса или формирования рекомендаций для отдельной статьи. В случае последнего варианта пользователь загружает статью перед переходом к выбору корпуса. Помимо пользователя в качестве акторов выделены различные компоненты, которые могут представлять собой отдельные сервисы (см. таблицу).

После открытия корпуса пользователь может отредактировать разметку вручную (данная активность опущена на диаграмме) или перейти на следующий шаг анализа. Рассмотрим вариант, когда пользователю необходимо автоматически разметить корпус, создав собственный тип аннотаций.

Представленная последовательность описывает цикл работы пользователя портала по основным сценариям без учета некоторых вариантов использования, которые будут восстановлены при дальнейшей реализации портала. При проектировании также были выделены компоненты, сервисы портала, которые выполняют отдельную смысловую функцию.

Архитектура портала

Как показано в предыдущем разделе, различные функции портала могут быть разделены на взаимодействующие друг с другом, но сравнительно независимые компоненты-сервисы. Подобное разделение хорошо подходит для распределенных систем и позволяет оптимально распределять нагрузку между серверами и повысить надежность системы.

Функции описанных сервисов могут быть условно разделены на функции интерфейса, отвечающие за взаимодействие с пользователем, бизнес-логику — основные инструменты портала, предназначенные для анализа корпусов и формирования рекомендаций для отдельных статей, и работу с данными. Портал должен иметь базу данных для хранения пользовательской информации, а также отдельное хранилище для корпусов, так как они могут занимать большие объемы памяти. Таким образом, архитектура портала является частным случаем трехслойной архитектуры. Упрощенное описание архитектуры представлено на рис. 2 (см. вторую сторону обложки). Трехслойная архитектура предполагает наличие следующих элементов: слой интерфейса (представления), слой приложения (домена, бизнес-логики), сервер БД (источник данных).

Портал является браузерным приложением и работает с тонким клиентом. Слой интерфейса отвечает за взаимодействие с пользователем, а также содержит некоторые связанные с этим модули, такие как визуальный редак-

Описание сервисов портала

Обозначение	Название компонента	Функции
System	Компонент взаимодействия с пользователем (далее Система)	Осуществляет взаимодействие с пользователем (функции интерфейса), управление остальными сервисами (функции диспетчера)
Search Engine	Сервис поиска корпусов	Выполняет поиск корпуса по запросу
Corpus Visualizer	Визуальный редактор разметки	Визуализирует разметку корпуса для пользователя, предоставляет возможность ручного редактирования разметки (добавления/редактирования/удаления аннотаций)
PR Editor	Визуальный редактор лексико-синтаксических шаблонов	Предоставляет интерфейс для создания лексико-синтаксических шаблонов и формирует на их основе обрабатываемые ресурсы для автоматической разметки корпуса
Annotator	Компонент разметки корпуса	Наносит разметку документа/корпуса автоматически, используя обрабатываемые ресурсы, соответствующие типам аннотаций
Statistics Processor	Компонент сбора статистики	Собирает статистические данные на основе разметки корпуса
Report Generator	Компонент формирования отчетов	Формирует отчеты-анализы и отчеты-сравнения, используя статистические данные; проводит сравнение характеристик документа и корпуса

тор разметки и визуальный редактор лексико-синтаксических шаблонов. Последние два модуля могли бы быть вынесены на слой бизнес-логики, однако их роль больше связана с интерфейсом и обеспечением более удобной и понятной для пользователя работы по анализу корпусов, нежели с основной задачей портала.

Слой приложения содержит модули для выполнения основных функций по работе с корпусами, а также социальную часть портала. Сервис поиска корпусов позволит оперативно подбирать корпус в зависимости от указанных параметров, ключевых слов или даже, возможно, от всего текста загруженной пользователем статьи. Разработка этого модуля чрезвычайно важна для портала, так как качество оценки стиля статьи по реализуемому методу будет напрямую зависеть от степени семантической близости пользовательской статьи и эталонного корпуса.

Социальная часть портала включает в себя форум, инструкции по работе с порталом и коллекцию материалов, в том числе элементы, созданные пользователями и переведенные ими в общий доступ: например, новые обрабатываемые ресурсы, списки слов и т. д. с описаниями, а также результаты анализов корпусов, проведенных с помощью портала. Теоретически социальная часть может считаться частью слоя взаимодействия с пользователем, однако вследствие более тесной связи с базой данных и необходимости обеспечивать интеллектуальный поиск (по аналогии с поиском корпусов) в данной архитектуре он является частью слоя приложения. Стоит отметить, что указанные в данном слое сервисы, как и модули визуальных редакторов в слое представления, являются достаточно независимыми и могут быть развернуты на различных серверах в случае большой нагрузки.

Наиболее важным для портала является сервис обработки корпусов, который и выполняет основную функцию системы — анализ корпусов и формирование рекомендаций по качеству стиля для отдельных статей. Согласно предложенному методу анализа корпусов его следует выполнять в три этапа, каждый из которых реализован в отдельном компоненте архитектуры. Предполагается, что данный сервис будет активно использовать ресурсы библиотеки GATE Embedded для работы с корпусами текстов и созданные с ее помощью специальные элементы.

Первый компонент — компонент разметки корпуса. Он отвечает за обработку выбранного

корпуса с помощью имеющихся ресурсов, созданных на основе имеющихся плагинов среды GATE (например, базовые плагины для токенизации, разбиения на предложения, морфологического разбора), специализированных, изначально добавленных в систему обрабатывающих ресурсов (к таким будут относиться средства для нанесения маркеров стиля по критериям, обозначенным в предыдущих разделах) или обрабатывающих ресурсов, созданных пользователем с помощью визуального редактора лексико-синтаксических шаблонов. Последние будут также использовать специализированные плагины, написанные с помощью библиотек GATE Embedded, только настроенные на выделение пользовательских лингвистических конструкций, списков слов и др.

Разметка корпуса является необходимым для дальнейшего анализа этапом, однако при использовании корпусов, уже имеющихся в системе, дополнительная разметка может не понадобиться, особенно при использовании портала "не лингвистами", для оценки качества стиля собственных статей. Поэтому работа данного компонента может быть пропущена. Стоит уточнить, что при разметке корпуса с помощью некоторых лексико-синтаксических шаблонов может понадобиться ручное редактирование аннотаций, поэтому между этапом разметки и этапом сбора статистики нужна возможность возвращения результатов в интерфейс для работы с пользователем. Этим, в частности, объясняется разделение компонентов разметки и сбора статистики.

Таким образом, компонент разметки корпуса получает на вход корпус и список обрабатываемых ресурсов, аннотации от которых должны быть нанесены на корпус. Для новых корпусов, которые еще не имеют разметки и, соответственно, аннотаций никаких типов, перед запуском обрабатываемых ресурсов по умолчанию проводится базовая обработка от токенизации до морфологического разбора. На выходе сервис, реализуемый данным компонентом, выдаст корпус с обновленной разметкой. Корпус и разметка принимаются и выдаются в формате, пригодном для вывода в интерфейс, обработки с помощью ресурсов GATE и последующих компонентов. При необходимости можно встроить механизмы конвертации в удобные для пересылки данных форматы и обратно для модулей визуализации и компонентов сервиса обработки корпусов.

Вторым этапом как анализа, так и формирования рекомендаций по стилю статьи, явля-

ется сбор статистики по имеющейся разметке. Данный сервис принимает в качестве входных данных: размеченный корпус; список "тегов", т. е. типов аннотаций, которые будут учитываться при сборе статистики; типы статистических характеристик, которые должны быть вычислены для отчета.

Этот компонент подсчитывает базовые характеристики, например, число слов в документе, число аннотаций определенного типа в документе, число предложений, содержащих такие аннотации и т. д. Затем на их основе вычисляются указанные во входных данных статистические характеристики — как для корпуса в целом, так и для отдельных документов. В качестве выходных данных выводятся вычисленные значения характеристик в удобном формате, например, XML-файла.

Последний компонент сервиса отвечает за создание отчетов по анализу корпуса и формирование рекомендаций для отдельной статьи на основе сравнения с эталонным корпусом по теме. Для анализа корпуса необходимо только вывести его в определенном удобном для пользователя формате, формирование же рекомендаций требует отдельной обработки статистических характеристик документа и эталонного корпуса. Результат также выводится в удобном для пользователя формате в виде рекомендаций с указаниями на количественные различия между документом и средними результатами по корпусу.

Последний слой архитектуры — работа с данными: базой пользователей и их документов, корпусов, отчетов и других файлов, а также отдельное хранилище корпусов. Хранилище корпусов должно быть достаточно большим или по крайней мере масштабируемым, так как в ходе развития популярности портала предполагается размещение крупных коллекций документов.

Предложенная архитектура обеспечивает высокую степень масштабируемости и надежность системы за счет полной или относительной независимости отдельных ее компонентов. Важным при дальнейшей разработке продукта будет являться выбор стандартов передачи данных между отдельными сервисами для повышения скорости, так как корпус текстов может быть достаточно объемным — вплоть до нескольких тысяч документов.

Заключение

В ходе данного исследования были выделены функциональные и нефункциональные

требования к реализации портала для анализа и оценки стиля научных публикаций на английском языке. В целом есть два основных пути использования портала. Первый путь — это анализ статистических характеристик разметки корпуса для профессиональных лингвистов, проводящих исследования собственных корпусов. Второй путь — сравнение характеристик статьи пользователя с характеристиками эталонного корпуса для пользователей, которым не требуются дополнительные настройки с точки зрения лингвистики, только рекомендации системы. Первый сценарий требует большой гибкости сервиса и работы с объемными данными, второй — максимальной простоты интерфейса. Функциональные требования к portalу представлены в виде диаграмм вариантов использования, основные процессы жизненного цикла системы и взаимодействие компонентов внутри системы также описаны с помощью диаграмм.

Функции портала естественным образом подразделяются на функции взаимодействия с пользователем, основные сервисы и работу с данными. В связи с этим для реализации портала предложена трехслойная архитектура, каждый слой которой составляют взаимодействующие друг с другом, но фактически независимые компоненты. Подобная архитектура способствует улучшению масштабируемости и надежности системы, так как каждый компонент может располагаться на отдельных серверах. Весь портал и хранилища данных могут быть размещены в облачном сервисе.

Список литературы

1. Шаров С. А. Представительный корпус русского языка в контексте мирового опыта // НТИ. Сер. 2. 2003. № 6. С. 9—17.
2. Корпусная лингвистика / Теория. URL: <http://corpora.iling.spb.ru/theory.htm> (дата обращения: 04.02.2017).
3. Scholz T., Conrad S. Style Analysis of Academic Writing // Natural Language Processing and Information Systems: proceedings of 16th International Conference on Applications of Natural Language to Information Systems (Alicante, Spain, June 28—30, 2011). 2011. P. 246—249.
4. Бартков Б. И., Минина Л. И., Миронец Ю. А. Структурные и функциональные особенности научного текста. Академия наук СССР, Дальневосточный науч. центр, Кафедра иностранных языков, 1985. 147 с.
5. Ермакова Л. М., Абашев М. А., Никитин Р. В., Ушаков Р. И. Методы автоматической классификации текстов по функциональным стилям // Вестник Пермского университета. Математика. Механика. Информатика. 2014. Вып. 4 (27). С. 78—83.
6. Научный стиль — Стилистический энциклопедический словарь русского языка. URL: http://stylistics.academic.ru/89/Научный_стиль (дата обращения: 05.06.2015).

7. **Скрипак И. А.** Языковое выражение экспрессивности как способа речевого воздействия в современном научном дискурсе: на материале статей лингвистического профиля на русском и английском языках. Дис. канд. филол. наук. Ставрополь, 2008. 199 с.
8. **Смирнова Л. Н.** Курс английского языка для научных работников. Л.: Наука, 1971. 330 с.
9. **Смирнова Л. Н.** Scientific English. Английский язык для научных работников. Курс для начинающих. Л.: Наука, 1980. 245 с.
10. **Цвиллинг М. Я.** Научная литература: язык, стиль, жанры. М.: Наука, 1985. 336 с.
11. **Academic Phrasebank.** URL: <http://www.phrasebank.manchester.ac.uk> (дата обращения: 20.01.2015).
12. **Anthony L.** Characteristic features of research article titles in computer science // IEEE Transactions on Professional Communication. 2001. N. 44 (3). P. 187–194.
13. **Guse J.** Communicative Activities for EAP. Cambridge: Cambridge University Press, 2013. 322 p.
14. **Hamp-Lyons L., Heasley B.** Study writing. A course in writing skills for academic purpose. Cambridge: Cambridge University Press, 2013. 213 p.
15. **IEEE** — The world's largest professional association for the advancement of technology. URL: <https://www.ieee.org/index.html> (дата обращения: 05.06.2015).
16. **Jacobs P. S., Krupka G. R., Rau L. F.** Lexico-Semantic Pattern Matching as a Companion to Parsing in Text Understanding // Workshop on Speech and Natural Language Colocated with the 6th Human Language Technology Conference. 1991. P. 337–341.
17. **Jordan R. R.** English for academic purposes. A guide and resource book for teachers. Cambridge: Cambridge University Press, 2012. 404 p.
18. **Khosmood F., Levinson R. A.** Automatic natural language style classification and transformation / BCS Corpus Profiling Workshop, 2008. URL: <https://style.soe.ucsc.edu/sites/default/files/CP08-KL-camera.pdf> (дата обращения: 08.02.2017).
19. **Laurence Anthony's AntConc.** URL: <http://www.laurenceanthony.net/software/antconc/> (дата обращения: 08.02.2017).
20. **Luyckx K., Daelemans W.** Shallow text analysis and machine learning for authorship attribution // Computational Linguistics in the Netherlands 2004: selected papers from the Fifteenth CLIN Meeting. Utrecht, LOT. 2005. P. 149–160.
21. **Scolz T., Conrad S.** Style Analysis of Academic Writing // Natural Language Processing and Information Systems: 16th International Conference on Applications of Natural Language to Information Systems, Proceedings. NLDB 2011, Alicante, Spain, June 28–30, 2011. P. 246–249.
22. **Siepmann D., Gallagher J. D., Hannay M., Mackenzie J. L.** Writing in English: a guide for advanced learners. Francke Verlag, 2011. 469 p.
23. **Strinyuk S. A., Shuchalova Y., Lanin V.** Academic Papers Evaluation Software // Application of Information and Communication Technologies (AICT), 2015 9th International Conference (14–16 Oct. 2015. Rostov-on-Don). IEEE, 2015. P. 506–510.
24. **Swales J. M., Feak C. B.** Academic writing for graduate students. Essential tasks and skills. 3d. Ed. The University of Michigan, 2014. 418 p.
25. **TUPH** Научный портал. URL: <https://kaemus.psych.ut.ee/&lang=Rus> (дата обращения: 15.05.2017).
26. **Turabian K.** A Manual For Writers of Term Papers, Theses and Dissertations. 7th ed. The University of Chicago Press, 2007. 470 p.
27. **Wallwork A.** English for academic research: vocabulary exercises. London: Springer, 2013. 193 p.

Yu. S. Shuchalova, Invited Lecturer, e-mail: iusshuchalova@hse.ru,

V. V. Lanin, Senior Lecturer, e-mail: vlanin@hse.ru,

National Research University Higher School of Economics, Perm

Research Portal for Scientific Publication Style Analysis

As any other genre, academic paper can be characterized by its own specific rules and features. Academic writing has investigated in terms of grammar, structure, genre and other crucial features, however, not enough attention has been paid to building a systematic approach. Recommendations given in guides and handbooks for both competent and novice academic writers in English are not systematized and sometimes even have obvious internal contradictions. The project named "Paper Cat" is aimed to provide corpus analysis tools for scientific publications in English. Authors assume that it possible to develop special software tools which can be able to perform automatic analysis based on natural language processing and corpus linguistic methods. The article describes a development of the Internet portal as a part of a project. Functional and non-functional requirements to the system are given and appropriate portal architecture is suggested. Heterogeneous components integrate due to service architecture. Implementation features are discussed taking into account the specifics of the subject area. Described portal can be used both for research and study aims.

Keywords: corpus studies, research portal, academic English, information system modeling

DOI: 10.17587/it.24.515-523

References

1. **Sharov S. A.** Predstavitel'nyj korpus russkogo jazyka v kontekste mirovogo opyta, *NTI*, Ser. 2. 2003. N. 6. P. 9–17 (in Russian).
2. **Korpusnaja lingvistika / Teorija**, available at: <http://corpora.iling.spb.ru/theory.htm> (available date: 04.02.2017) (in Russian).
3. **Scholz T., Conrad S.** Style Analysis of Academic Writing. *Natural Language Processing and Information Systems: 16th International Conference on Applications of Natural Language to Information Systems* (Alicante, Spain, June 28–30, 2011), 2011, pp. 246–249.
4. **Bartkov B. I., Minina L. I., Mironec Ju. A.** *Strukturnye i funkcional'nye osobennosti nauchnogo teksta*. Akademiya nauk SSSR, Dal'nevostochnyj nauch. centr, Kafedra inostrannyh jazykov, 1985, 147 p. (in Russian).
5. **Ermakova L. M., Abashev M. A., Nikitin R. V., Ushakov R. I.** Metody avtomaticheskoy klassifikacii tekstov po funkcional'nyim stiljam, *Vestnik Permskogo universiteta. Matematika. Mehanika. Informatika*, 2014, no. 4 (27), pp. 78–83 (in Russian).
6. **Nauchnyj stil'** — Stilisticheskij jenciklopedicheskij slovar' russkogo jazyka / Eds by M. N. Kozhina; 2nd edition, Moscow, Flinta: Science, 2006, 696 p. (in Russian).

7. **Skripak I. A.** Jazykove vyrazhenie jekspresivnosti kak sposoba rechevogo vozdejstviya v sovremennom nauchnom diskurse: na materiale statej lingvisticheskogo profilja na russkom i anglijskom jazykah. Phd thesis. Stavropol', 2008. 199 p. (in Russian).
8. **Smirnova L. N.** *Kurs anglijskogo jazyka dlja nauchnyh rabotnikov*. Leningrad: Nauka, 1971, 330 p. (in Russian).
9. **Smirnova L. N.** *Scientific English. Anglijskij jazyk dlja nauchnyh rabotnikov. Kurs dlja nachinajushchih*. Leningrad: Nauka, 1980, 245 p. (in Russian).
10. **Cvilling M. Ja.** *Nauchnaja literatura: jazyk, stil', zhanry*, Moscow: Nauka, 1985, 336 p. (in Russian).
11. **Academic** Phrasebank, available at: <http://www.phrasebank.manchester.ac.uk/> (date of access: 20.01.2015).
12. **Anthony L.** Characteristic features of research article titles in computer science, *IEEE Transactions on Professional Communication*, 2001, no. 44 (3), pp. 187–194.
13. **Guse J.** *Communicative Activities for EAP*. Cambridge, Cambridge University Press, 2013, 322 p.
14. **Hamp-Lyons L., Heasley B.** *Study writing*. Cambridge, Cambridge University Press, 2013, 213 p.
15. **IEEE** — The world's largest professional association for the advancement of technology, URL: <https://www.ieee.org/index.html> (date of access: 05.06.2015).
16. **Jacobs P. S., Krupka G. R., Rau L. F.** Lexico-Semantic Pattern Matching as a Companion to Parsing in Text Understanding, *Workshop on Speech and Natural Language colocated with the 6th Human Language Technology Conference*, 1991, pp. 337–341.
17. **Jordan R. R.** *English for academic purposes*, Cambridge, Cambridge University Press, 2012, 404 p.
18. **Khosmood F., Levinson R. A.** Automatic natural language style classification and transformation, *BCS Corpus Profiling Workshop*, 2008, available at: <https://style.soe.ucsc.edu/sites/default/files/CP08-KL-camera.pdf> (date of access: 08.02.2017).
19. **Laurence Anthony's AntConc**, available at: <http://www.laurenceanthony.net/software/antconc/> (date of access: 08.02.2017).
20. **Luyckx K., Daelemans W.** Shallow text analysis and machine learning for authorship attribution, *Computational Linguistics in the Netherlands 2004: selected papers from the Fifteenth CLIN Meeting*, van der Wouden T. [Ed.], e. a., Utrecht, LOT, 2005, pp. 149–160.
21. **Scolz T., Conrad S.** Style Analysis of Academic Writing, *Natural Language Processing and Information Systems: 16th International Conference on Applications of Natural Language to Information Systems*, Proceedings. NLDB 2011, Alicante, Spain, June 28–30, 2011, pp. 246–249.
22. **Siepmann D.** *Writing in English: A Guide for Advanced Learners* / J. D. Gallagher, M. Hannay; J. L. Mackenzie. — UTB, 2011, 469 p.
23. **Strinyuk S. A., Shuchalova Y., Lanin V.** Academic Papers Evaluation Software, *Application of Information and Communication Technologies (AICT), 2015 9th International Conference (14–16 Oct. 2015. Rostov-on-Don)*: IEEE, 2015, pp. 506–510.
24. **Swales J. M., Feak C. B.** *Academic writing for graduate students. Essential tasks and skills*. The University of Michigan, 2014, 418 p.
25. **TŪPH.** Nauchnyy portal, available at: <https://kaemus.psych.ut.ee/&lang=Rus> (date of access: 15.05.2017) (in Russian).
26. **Turabian K.** *A Manual For Writers of Term Papers, Theses and Dissertations*. — 7th ed. — The University of Chicago Press, 2007, 470 p.
27. **Wallwork A.** *English for academic research: vocabulary exercises*, London, Springer, 2013, 193 p.

УДК 004.421.6

DOI: 10.17587/it.24.523-528

Н. И. Лиманова, д-р техн. наук, проф., e-mail: Nataliya.I.Limanova@gmail.com,
М. Н. Седов, аспирант, e-mail: SedovMN@inbox.ru,

Поволжский государственный университет телекоммуникаций и информатики, г. Самара

Алгоритм нечеткого поиска реквизитов физических лиц в базах данных на основе метрики Левенштейна

При передаче данных от одного учреждения к другому возникает проблема персональной идентификации физических лиц, у которых частично или полностью не совпадают реквизиты. В работе представлен алгоритм нечеткого поиска, использующий модифицированную метрику Левенштейна, позволяющий выполнять поиск физических лиц в базе данных на основе нечеткого сравнения. Алгоритм реализован на языке PL-SQL в СУБД Oracle 11g.

Ключевые слова: межведомственный информационный обмен, нечеткое сравнение, поиск персональных данных, функция интеллектуального сравнения, персональный идентификационный номер (ПИН)

Введение

В процессе обработки информации о физических лицах в базах данных для удобства обработки каждому набору реквизитов физических лиц (таких как ФИО, адрес, номера паспорта, СНИЛС и т. п.) присваивается так называемый персональный идентификационный номер (ПИН). В случае обработки или пере-

дачи данных о физическом лице вся привязка осуществляется именно к этому ПИНу. При осуществлении обмена информацией о физических лицах между различными учреждениями возникает проблема сопоставления реквизитов из одной базы данных реквизитам в другой. Если проводить данное сопоставление методом простого сравнения реквизитов (метод прямого сравнения), то в случае ошибочных данных,

полученных, например, при ошибках ввода или ином искажении реквизитов, такие данные найдены не будут. Для решения этой проблемы применяют так называемый нечеткий поиск, при котором выполняется сопоставление информации заданному образцу поиска или близкому к этому образцу значению.

Соответственно, для однозначной привязки реквизитов физического лица из базы-источника необходимо выполнять нечеткий поиск реквизитов в базе-приемнике, который должен учитывать множество факторов: потенциальные ошибки при ручном вводе, отсутствующие или устаревшие реквизиты и т. п. Такой поиск целесообразно реализовать в виде алгоритма нечеткого поиска и основанного на нем специализированного программного обеспечения [1].

Известные алгоритмы нечеткого поиска строк

Рассмотрим существующие алгоритмы нечеткого поиска и проанализируем их [2]. Начнем с разработанного Робертом Расселом (Robert C. Russel) и Маргарет Кинг Оделл (Margaret King Odell) алгоритма Soundex [3]. Это один из алгоритмов сравнения двух строк по их звучанию. Он устанавливает одинаковый индекс для строк, имеющих схожее звучание в языке согласно заданной таблице схожих по звучанию символов и их сочетаний. Однако он имеет существенный недостаток: этот алгоритм привязан к языку, на котором написаны анализируемые строки. Данный алгоритм используется сейчас в основном в англоговорящей среде, для которой подобная таблица уже существует.

Следующий из рассматриваемых алгоритмов — алгоритм расширения выборки [4] — часто применяется в системах проверки орфографии. Он основан на сведении задачи о нечетком поиске к задаче о точном поиске. Этот алгоритм подразумевает построение наиболее вероятных "неправильных" вариантов поискового шаблона. Основное достоинство алгоритма заключается в легкости его модификации для генерации "ошибочных" вариантов по произвольным правилам. У алгоритма есть и недостатки, главный из которых — большое число проверок для слов существенной длины, поскольку из них можно получить много "ошибочных" слов.

Широко известен алгоритм на основе кода Хэмминга, который применяется при кодировании и декодировании данных. Линейные коды, как правило, хорошо справляются с редкими и большими опечатками. Однако их эффектив-

ность при сравнении слов с частыми, но небольшими ошибками достаточно низкая. В данном алгоритме также присутствуют дополнительные затраты на кодирование информации.

Следующий из рассматриваемых алгоритмов не совсем подходит под поставленную задачу, но для полноты картины не упомянуть о нем все же нельзя. Это алгоритм, использующий триангуляционные деревья, которые позволяют индексировать множества произвольной структуры при условии, что на них задана метрика. Существует довольно много различных модификаций данного алгоритма, но все они не слишком эффективны в случае текстового поиска и чаще используются в базе данных изображений или других сложных объектов.

Алгоритм Bitap (также известный как Shift-Or или Baeza-Yates-Gonnet) и различные его модификации наиболее часто используют для нечеткого поиска без индексации [5]. Впервые идею этого алгоритма предложили Ricardo Baeza-Yates и Gaston Gonnet, опубликовав соответствующую статью в 1992 г. Оригинальная версия алгоритма имеет дело только с заменами символов и фактически вычисляет расстояние Хемминга. Но немного позже Sun Wu и Udi Manber предложили модификацию этого алгоритма для вычисления расстояния Левенштейна, т. е. привнесли поддержку вставок и удалений и разработали на его основе первую версию утилиты Unix — *agrep*. Высокая скорость работы этого алгоритма обеспечивается за счет битового параллелизма вычислений — за одну операцию возможно провести вычисления над 32 и более битами одновременно. При этом тривиальная реализация поддерживает поиск слов длиной не более 32 символов. Использование типов больших размерностей замедляет работу алгоритма.

Рассмотрим далее алгоритм Вагнера—Фишера [6], который позволяет для двух строк найти расстояние Левенштейна — минимальное число операций вставки одного символа, удаления одного символа и замены одного символа на другой, необходимых для превращения одной строки в другую. Данный алгоритм имеет ряд значительных преимуществ перед всеми описанными выше, а именно: относительно невысокую сложность реализации; возможность качественного сравнения схожести более чем двух строк; несколько вариантов реализации, которые можно использовать в зависимости от конфигурации системы; универсальность для всевозможных алфавитов. Также у данного алгоритма существует одна интересная модифи-

кация, которая позволяет находить расстояние Дамерау—Левенштейна [7]. В нем к операциям вставки, удаления и замены символов, определенных в расстоянии Левенштейна, добавлена операция транспозиции (перестановки) символов. Ф. Дамерау показал, что 80 % ошибок при наборе текста человеком являются транспозициями. Вследствие всех перечисленных выше плюсов именно метрика Левенштейна была выбрана для основы рассматриваемого алгоритма нечеткого поиска реквизитов физических лиц в базах данных.

Математическая модель

Рассмотрим общую метрику Левенштейна, которая поддерживает три операции со строкой: вставки, замены и удаления символа, причем все три операции имеют одинаковый вес [8, 9]. Для дальнейшей работы была построена лингвистическая переменная "схожесть строк". Решено выделить следующие термы: "строки совпадают", "строки почти совпадают", "строки похожи", "строки и похожи, и непохожи одновременно", "строки не похожи".

В результате анализа функций принадлежности лингвистических термов возникла необходимость модификации метода вычисления метрики Левенштейна. Потребовалось модифицировать метрику таким образом, чтобы расстояние между строками зависело в том числе и от длины сравниваемых строк.

Теорема. Обозначим как $p(s_1, s_2)$ метрику Левенштейна, а $\|s_i\|$ длину строки s_i . Тогда функция

$$r(s_1, s_2) = \frac{p(s_1, s_2)}{\max\{\|s_1\|, \|s_2\|\}} \quad (1)$$

является метрикой.

Доказательство. Поскольку $p(s_1, s_2)$ — метрика, то имеем

$$\begin{aligned} p(s_1, s_2) &\geq 0, \\ p(s_1, s_2) &= p(s_2, s_1), \\ p(s_1, s_2) + p(s_2, s_3) &\geq p(s_1, s_3) \end{aligned}$$

для любых строк s_1, s_2 и s_3 .

Учитывая эти соотношения и равенство (1), приходим к выводу, что $r(s_1, s_2)$ удовлетворяет первым двум аксиомам, определяющим метрику. Остается доказать, что для любых строк s_1, s_2 и s_3 функция $r(s_1, s_2)$ удовлетворяет неравенству треугольника:

$$r(s_1, s_2) + r(s_2, s_3) \geq r(s_1, s_3).$$

Запишем это неравенство в виде:

$$\frac{p(s_1, s_2)}{\max\{\|s_1\|, \|s_2\|\}} + \frac{p(s_2, s_3)}{\max\{\|s_2\|, \|s_3\|\}} - \frac{p(s_1, s_3)}{\max\{\|s_1\|, \|s_3\|\}} \geq 0.$$

Возможны следующие случаи:

1. $\|s_1\| \leq \|s_2\| \leq \|s_3\|$.
2. $\|s_2\| \leq \|s_3\| \leq \|s_1\|$.
3. $\|s_3\| \leq \|s_1\| \leq \|s_2\|$.
4. $\|s_2\| \leq \|s_1\| \leq \|s_3\|$.
5. $\|s_1\| \leq \|s_3\| \leq \|s_2\|$.
6. $\|s_3\| \leq \|s_2\| \leq \|s_1\|$.

Рассмотрим первый случай. Имеем:

$$\begin{aligned} &\frac{p(s_1, s_2)}{\max\{\|s_1\|, \|s_2\|\}} + \frac{p(s_2, s_3)}{\max\{\|s_2\|, \|s_3\|\}} - \\ &- \frac{p(s_1, s_3)}{\max\{\|s_1\|, \|s_3\|\}} = \frac{p(s_1, s_2)}{\|s_2\|} + \frac{p(s_2, s_3)}{\|s_3\|} - \frac{p(s_1, s_3)}{\|s_3\|} \geq \\ &\geq \frac{1}{\|s_3\|} (p(s_1, s_2) + p(s_2, s_3) - p(s_1, s_3)) \geq 0. \end{aligned}$$

Таким образом, для первого случая неравенство треугольника выполняется. Поскольку второй случай аналогичен первому, на основании подобных выкладок делаем вывод, что для второго случая неравенство треугольника также выполняется.

Перейдем к рассмотрению третьего случая. Итак, в третьем случае имеем:

$$\begin{aligned} &r(s_1, s_2) + r(s_2, s_3) - r(s_1, s_3) = \\ &= \frac{1}{\|s_2\|} (r(s_1, s_2) + r(s_2, s_3)) - \frac{1}{\|s_1\|} r(s_1, s_3). \quad (2) \end{aligned}$$

Рассмотрим вопрос о том, когда достигается минимум функции, находящейся в правой части этого равенства. Понятно, что если выражение $r(s_1, s_2) + r(s_2, s_3)$ достигает минимума, а $r(s_1, s_3)$ — максимума, то значение всего выражения будет минимальным. Указанные два условия могут выполняться одновременно, если одновременно выполняются два следующих утверждения:

- строки s_1 и s_3 не имеют общих символов;
- строки s_1 и s_3 входят в качестве подстрок в s_2 .

Тогда:

$$\begin{aligned} r(s_1, s_3) &= \max\{\|s_1\|, \|s_3\|\} = \|s_1\|, \\ r(s_1, s_2) &= \|s_3\| + \|C\|, \quad r(s_2, s_3) = \|s_1\| + \|C\|, \end{aligned}$$

где C — вспомогательная строка, и, таким образом, минимальное значение выражения (2) можно записать в следующем виде:

$$\frac{\|s_3\| + \|C\| + \|s_1\| + \|C\|}{\|s_3\| + \|s_1\| + \|C\|} - \frac{\|s_1\|}{\|s_1\|} = \frac{\|C\|}{\|s_3\| + \|s_1\| + \|C\|} \geq 0.$$

Следовательно, в третьем случае для функции $r(s_1, s_3)$ также выполняется неравенство треугольника. Остальные случаи аналогичны уже рассмотренным. Таким образом, функция $r(s_1, s_2)$ является метрикой, заданной на множестве строк. Теорема доказана.

Замечание. Функция $r(s_1, s_2)$ принадлежит отрезку $[0, 1]$ для любых s_1 и s_2 .

В предложенном алгоритме данная метрика применяется для работы со строковыми реквизитами физических лиц, к которым относятся ФИО, адрес, документ и т. д. В связи

с этим построенная с использованием данной метрики лингвистическая переменная позволяет обрабатывать запросы поиска для человека, похожего на другого человека, по реквизитам. Приняв от пользователя такой запрос, мы фактически получаем два значения: значение искомого реквизита и радиус поиска.

Алгоритм нечеткого поиска реквизитов физических лиц

Укрупненная блок-схема разработанного алгоритма нечеткого поиска реквизитов физических лиц в базах данных представлена на рис. 1.

В реализации алгоритма на языке PL-SQL СУБД Oracle 11g за предварительную выборку всех записей, отдаленно похожих на искомую, отвечает блок "Запрос количества идентичных людей в базе данных". Этот блок работает

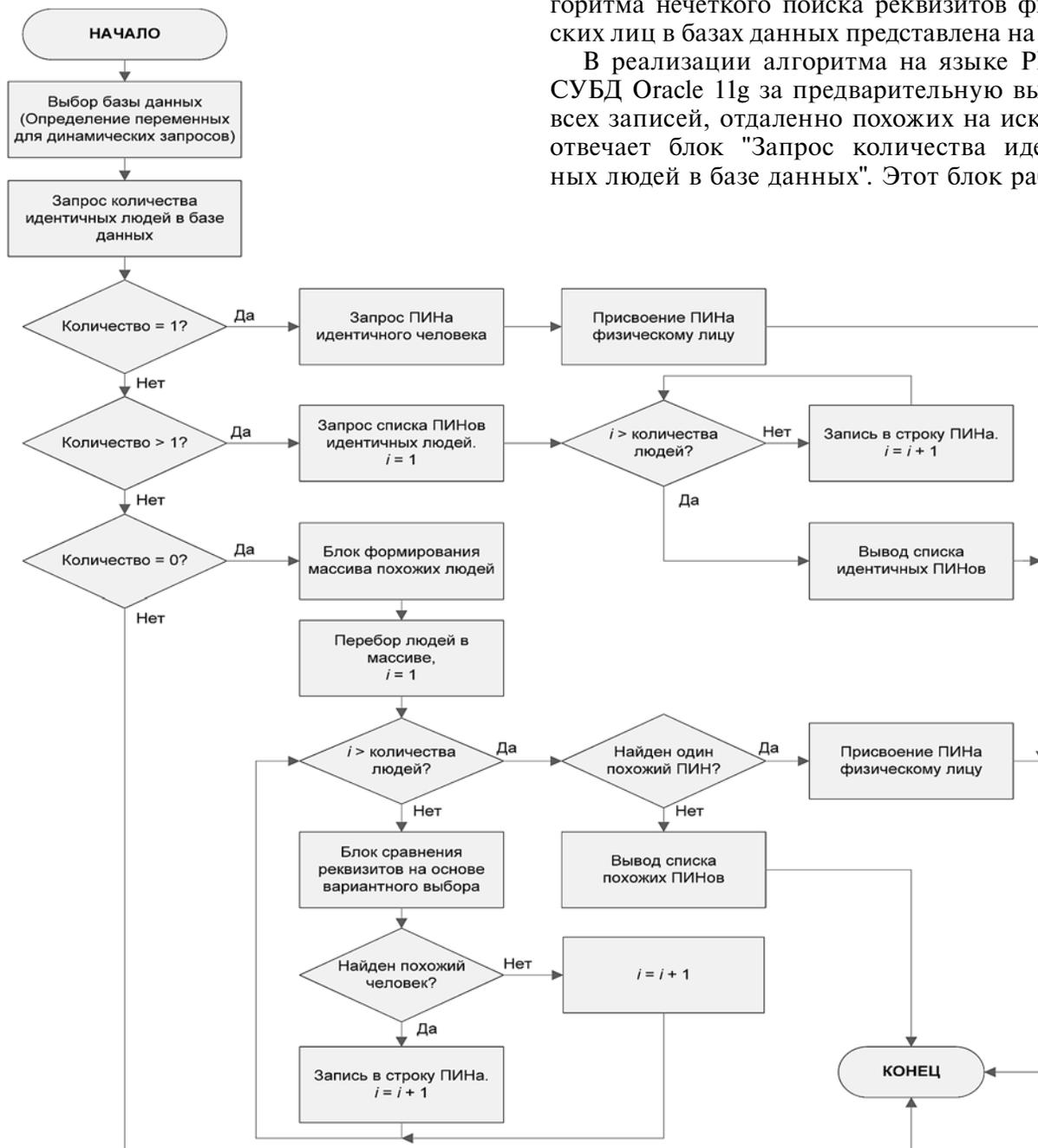


Рис. 1. Укрупненная блок-схема алгоритма нечеткого поиска реквизитов физических лиц в базах данных

по алгоритму прямого частичного сравнения разных наборов реквизитов, например, имени, отчества и даты рождения, формируя, тем самым, рабочий набор данных для рассматриваемого алгоритма идентификации. Затем в работу вступает "Блок сравнения реквизитов", ключевые функции которого отводятся логически выделенным процедурам COMPARISON_STRING и COMPARISON_NUMBER, созданным на основе модифицированного метода вычисления метрики Левенштейна, которые позволяют проводить интеллектуальное сравнение двух похожих строк или чисел, с учетом возможных неточностей или ошибок ввода. С помощью указанных процедур программа формирует набор совпадений и по результатам обработки предлагаемой и искомой записи выносит решение об идентичности строк. Например, у человека совпадает имя, отчество, дата рождения и номер паспорта, а в фамилии допущена ошибка в одну букву. В таком случае программа однозначно идентифицирует реквизиты. Данные процедуры могут применяться не только для идентификации реквизитов, но также везде, где требуется полнотекстовый поиск с нечетко заданными входными данными.

Алгоритм идентификации аккумулирует так называемый "опыт прошлых идентификаций" и записывает его в специально отведенное место в базе данных для использования в последующих идентификациях. Это позволяет сохранить не только результаты автоматической работы программы, но и решения операторов после отработки ими оставшихся не найденных реквизитов.

Технические и экономические показатели алгоритма

Для сравнительного анализа разработанного алгоритма рассмотрим метод на основе прямого сравнения. При использовании данной технологии упор идет на скорость обработки данных, а не на качество принятия решения системой. В итоге, после окончания работы процедуры на основе прямого сравнения остается много данных (около 20—30 % от общего числа строк), не связанных с исходными, которые необходимо обрабатывать вручную, что крайне затруднительно при больших объемах обрабатываемых данных.

При экспериментальном сравнении рабочих показателей двух алгоритмов получены следующие результаты.

Алгоритм прямого сравнения:

скорость обработки данных: ~ 100 000 строк в час;

точность идентификации (вероятность точного поиска реквизитов): ~ 80 %.

Алгоритм идентификации на основе нечеткого сравнения:

скорость обработки данных: ~ 80 000 строк в час;

точность идентификации (вероятность точного поиска реквизитов) ~ 99,9 %.

Отсюда можно сделать вывод, что у разработанного алгоритма минимизирована работа оператора по ручной отработке результатов, т. е. хотя скорость обработки несколько меньше, но алгоритм позволяет существенно разгрузить операторов за счет интеллектуальной системы принятия решений, чего не может предложить алгоритм прямого сравнения.

При сравнении экономических характеристик разработанного программного обеспечения на основе описываемого алгоритма с процедурой прямого сравнения для годового объема нечеткого поиска в 1 200 000 физических лиц были получены следующие данные: трудовые затраты на обработку информации по методу нечеткого сравнения по сравнению с методом прямого сравнения уменьшены в 6,7 раза; абсолютное снижение трудовых затрат составило 1446 ч; годовые затраты при использовании метода нечеткого сравнения уменьшились в 3 раза по сравнению с аналогичным периодом применения метода прямого сравнения, а годовой экономический эффект превысил 580 000 руб. Для наглядности некоторые стоимостные показатели, формирующиеся при использовании разработанного и применявшегося до настоящего времени программного обеспечения, отображены на диаграмме, приведенной на рис. 2 (см. вторую сторону обложки). Значения затрат отложены по оси ординат в рублях.

Заключение

Рассмотренный алгоритм нечеткого поиска персональных данных позволяет выполнять поиск реквизитов физических лиц в базах данных, используя данные ранее проведенного поиска, имеет высокую точность поиска и скорость работы по сравнению с методом прямого сравнения. Встроенная система приоритета реквизитов позволяет идентифицировать человека в таких случаях, как смена фамилии, имени, переезд, ошибки при ручном вводе данных, а также при частично отсутствующих реквизитах.

В перспективе данный алгоритм имеет возможность успешного внедрения в системы глобального объединения хранилищ государственных или коммерческих организаций, для ведения единой базы данных населения любой страны мира. Логическая структура разработанного алгоритма позволяет реализовать его на любом популярном языке программирования. Масштабируемость алгоритма позволяет применять программные процедуры на его основе, как в малых организациях, так и в крупных корпорациях, везде, где ведется и актуализируется реестр данных физических лиц. Возможные примеры использования: портал госуслуг; медицинские электронные системы; кадровые и бухгалтерские системы учета служащих; банковские системы хранения данных о клиентах и т. п.

Алгоритм реализован на языке PL-SQL системы управления базами данных Oracle 11g. Разработанное программное обеспечение, реализующее алгоритм нечеткого поиска персональных данных, внедрено и успешно функционирует с 2007 г. в нескольких муниципальных учреждениях г. Тольятти Самарской области.

1. **Международный фонд** автоматической идентификации. Технологии автоматической идентификации. URL: <http://www.fond-ai.ru/art1/art223.html> (дата обращения: 28.01.2018).
2. **Желудков А. В., Макаров Д. В., Фадеев П. В.** Особенности алгоритмов нечеткого поиска // Инженерный вестник МГТУ им. Н. Э. Баумана, 2014. С. 502—503.
3. **Soundex** метод нечеткого поиска. URL: <https://ru.wikipedia.org/wiki/Soundex> (дата обращения: 28.01.2018).
4. **Харитonenков А. В.** Поиск на неточное соответствие: коды Хемминга. URL: <http://www.jurnal.org/articles/2009/inf32.html> (дата обращения: 28.01.2018).
5. **Двоичный** алгоритм поиска подстроки. URL: https://ru.wikipedia.org/wiki/Двоичный_алгоритм_поиска_подстроки (дата обращения: 28.01.2018).
6. **Задача** о редакционном расстоянии, алгоритм Вагнера-Фишера. URL: http://neerc.ifmo.ru/wiki/index.php?title=Задача_о_редакционном_расстоянии,_алгоритм_Вагнера-Фишера (дата обращения: 28.01.2018).
7. **Расстояние** Дамерау — Левенштейна. URL: https://ru.wikipedia.org/wiki/Расстояние_Дамерау_—_Левенштейна (дата обращения: 28.01.2018).
8. **Левенштейн В. И.** Двоичные коды с исправлением выпадений, вставок и замещений символов // Доклады Академии наук СССР. 1965. Т. 163, № 4. С. 845—848.
9. **Бойцов Л. М.** Анализ строк. URL: http://itman.narod.ru/articles/infoscope/string_search.1-3.html (дата обращения: 28.01.2018).

N. I. Limanova, D. Sc., Professor, M. N. Sedov, Postgraduate Student, e-mail: SedovMN@inbox.ru, Volga State University of Telecommunications and Informatics, Samara, Russian Federation

Fuzzy Searching Algorithm of Personal Details on the Basis of Levenshtein Distance

During the information exchange from one department to another there is a problem of personal identification. This problem concerns people who have partially or completely not coinciding personal details. In the represented work the new algorithm for identification of such people is elaborated. The algorithm is based on the fuzzy comparison and the metrics of Levenshtein. It allows us to find persons who have partial or complete not matching in surnames, names and other requisites in databases. The algorithm is implemented in PL-SQL in the Oracle database 11g.

Keywords: interdepartmental exchange of information; fuzzy comparison; search of personal details; function of intellectual matching; personal identification number (PIN)

DOI: 10.17587/it.24.523-528

References

1. **Mezhdunarodnyj** fond avtomaticheskoy identifikacii. Tehnologii avtomaticheskoy identifikacii (International Fund for Automatic Identification. Automatic identification technologies), available at: <http://www.fond-ai.ru/art1/art223.html>, free. rus. lang (date of access: 28.01.2018). (in Russian).
2. **Zheludkov A. V., Makarov D. V., Fadeev P. V.** Osobnosti algoritmov nechotkogo poiska (Features of fuzzy search algorithms). Moscow, Inzhenernyj vestnik MGTU im. N. Je. Bauman, 2014, pp. 502—503 (in Russian).
3. **Soundex** metod nechotkogo poiska (Soundex fuzzy search method), available at: <https://ru.wikipedia.org/wiki/Soundex> (date of access: 28.01.2018) (in Russian).
4. **Haritonov A. V.** Poisk na netochnoe sootvetstvie: kody Hemming (Search for inaccurate matching: Hamming codes), available at: <http://www.jurnal.org/articles/2009/inf32.html> (date of access: 28.01.2018) (in Russian).
5. **Dvoichnyj** algoritm poiska podstroki (Binary search algorithm for substring), available at: https://ru.wikipedia.org/wiki/Двоичный_алгоритм_поиска_подстроки (date of access: 28.01.2018) (in Russian).
6. **Zadacha** o redakcionnom rasstojanii, algoritm Vagnera-Fishera (The problem of the editorial distance, Wagner-Fisher algorithm), available at: http://neerc.ifmo.ru/wiki/index.php?title=Задача_о_редакционном_расстоянии,_алгоритм_Вагнера-Фишера (date of access: 28.01.2018) (in Russian).
7. **Rasstoianie** Damerau — Levenshteina (Distance Damerau — Levenshtein), available at: https://ru.wikipedia.org/wiki/Расстояние_Дамерау_—_Левенштейна (date of access: 28.01.2018) (in Russian).
8. **Levenshtejn V. I.** Dvoichnye kody s ispravleniem vypadenij, vstavok i zameshenij simvolov (Binary codes with correction of fallouts, inserts and substitutions of symbols), *Doklady Akademii nauk SSSR*, 1965, vol. 163, no. 4, pp. 845—848 (in Russian).
9. **Bojcov L. M.** Analiz strok (Analysis of strings), available at: http://itman.narod.ru/articles/infoscope/string_search.1-3.html. (date of access: 28.01.2018) (in Russian).

А. Е. Александров, д-р техн. наук, проф., e-mail: femsystem@yandex.ru,
Московский технологический университет (МИРЭА)

Оценка достоверности результатов контроля металла на основе альтернативного алгоритма

Поставлена задача восстановления характеристик прибора для проведения неразрушающего контроля и обоснования исходной дефектности для использования ее при построении математических моделей для расчета и прогнозирования уровня безопасности объектов энергетики. Приведено описание решения прямой задачи для корректировки распределения обнаруженных дефектов и построения их действительного распределения. Разработан метод решения обратной задачи восстановления характеристик используемого прибора для измерения дефектности исследуемых объектов на основе альтернативного алгоритма.

Ключевые слова: вероятностный анализ безопасности, исходная дефектность, вероятностные методы механики разрушения, решение обратных задач

Особенности анализа дефектности при обосновании уровня безопасности объектов энергетики

При построении математических моделей для расчета и прогнозирования уровня безопасности объектов энергетики важное значение приобретает задача задания и обоснования исходной дефектности, которая выявляется в результате неразрушающего контроля металла, проводимого в период производственного изготовления (заводской контроль), в период подготовки к сдаче в эксплуатацию (входной и предэксплуатационный контроль) и в период эксплуатации объектов (эксплуатационный контроль). Наличие исходных дефектов связано с технологическими операциями изготовления и эксплуатации материалов конструкций. Большое число дефектов появляется в сварных соединениях. Существенно меньше их можно обнаружить в основном металле после операцийковки и штамповки.

Сложность рассматриваемой задачи состоит в оценке достоверности исходной информации, получаемой в результате проведения контроля различными методами контроля, и ее представления в расчетных моделях. В большинстве случаев исходная информация о дефектности представляется в виде случайных распределе-

ний размеров дефектов и плотности распределения их вероятности [1]. Зная характеристики средств измерения дефектности, можно перейти от обнаруженных распределений к действительным. Однако, как показано в работах [2, 6], определение одной из этих характеристик — функции вероятности обнаружения дефекта заданного размера — связано с рядом трудностей и, в первую очередь, с высокими ресурсными затратами ее получения.

Использование для восстановления функции вероятности обнаружения дефекта метода обратных задач, когда сама характеристика может быть восстановлена по результатам исходных распределений обнаруженных дефектов, представляется в этой ситуации весьма эффективным средством решения данной проблемы.

Приборы, используемые при проведении контроля металла, характеризуются следующими параметрами:

- чувствительностью метода, т. е. минимальным размером дефекта a_0 , обнаруживаемым данным методом с заданной погрешностью;
- погрешностью определения размера дефекта Δa ;
- функцией вероятности обнаружения дефекта заданного размера $P_r(a)$.

Для нахождения функции $P_r(a)$ используют тест — образцы дефектов, максимально

приближенных к реальным. Дефекты закладываются в изделие, аналогичное эксплуатируемому, и далее осуществляется контроль с использованием различных средств и методов контроля. По результатам проведенных экспериментов строится аппроксимирующая зависимость — кривая вероятности выявления дефектов от их размера. В качестве аппроксимирующей во многих случаях применяется следующая зависимость [3]:

$$P_r(a) = 1 - \exp(-r(a - a_0)), \quad (1)$$

где r — коэффициент достоверности проводимого контроля, a — размер дефекта.

На рис. 1 представлены функции вероятности обнаружения дефекта, полученные с использованием формулы (1), для различных значений коэффициента достоверности r . Из-за несовершенных средств измерения часть дефектов остаются пропущенными. Число пропущенных дефектов и их размеры можно восстановить, если для данного метода контроля известна зависимость $P_r(a)$.

Метод корректировки размеров и числа обнаруженных дефектов и получения распределения действительных их размеров представлен в работах [1, 4]. Согласно этим работам действительное среднее число дефектов m_a в единице объема, а также действительное распределение дефектов по размеру $p_a(a)$ с учетом их пропуска при контроле определяются по следующим формулам:

$$m_a = m_f / \int_0^{\infty} p_f(a) / P_r(a) da; \quad (2)$$

$$p_a(a) = p_f(a) / P_r(a) / \int_{a_0}^{\infty} p_f(a) / P_r(a) da, \quad (3)$$

где $p_a(a)$ — плотность распределения вероятности действительных размеров дефектов; m_f — среднее число обнаруженных дефектов в единице объема; $p_f(a)$ — плотность распределения вероятности обнаруженных размеров дефектов.

Рассчитанное в соответствии с формулой (3) распределение действительных размеров дефектов применяется в математических моделях расчета и прогнозирования уровня безопасности. Заметим, что полученное действительное распределение характеризуется монотонно убывающей зависимостью от их размеров [2]. Для задания плотности распре-

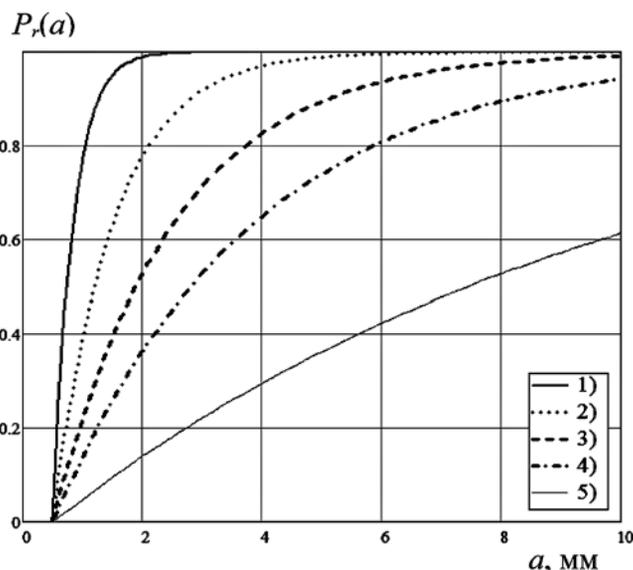


Рис. 1. Зависимость вероятности обнаружения дефекта $P_r(a)$ для разных значений коэффициента достоверности r : 1) $r = 3,0$; 2) $r = 1,0$; 3) $r = 0,5$; 4) $r = 0,3$; 5) $r = 0,1$

деления действительных размеров $p_a(a)$ был использован экспоненциальный закон в виде следующей зависимости:

$$p_a(a) = \frac{(1/\lambda) \exp(-a/\lambda)}{\int_{a_0}^{\infty} (1/\lambda) \exp(-a/\lambda) da}, \quad (4)$$

где постоянная λ — параметр распределения; a — размер дефекта.

С учетом зависимостей (1) и (4) плотность распределения вероятности обнаруженных дефектов $p_f(a)$ можно записать в виде

$$p_f(a) = \frac{(1/\lambda) \exp(-a/\lambda) [1 - \exp(-r(a - a_0))]}{\int_{a_0}^{\infty} (1/\lambda) \exp(-a/\lambda) [1 - \exp(-r(a - a_0))] da} \quad (5)$$

Из формулы (5) следует, что при известном распределении $p_f(a)$ необходимо найти два параметра: параметр распределения λ и коэффициент достоверности r .

Алгоритм решения обратной задачи по нахождению действительного распределения дефектов

Задачи подобного типа (определение параметров прибора и действительного распределения измеряемой величины по показанию при-

бора) согласно работе [5] относятся к классу задач интерпретации данных наблюдений или диагностики. Там же приведена общая постановка такого класса обратных задач, сформулированная в виде операторного уравнения:

$$Az = u, z \in Z, u \in U, \quad (6)$$

где Z — метрическое пространство искомого в обратной задаче интерпретации характеристик объекта или процесса; U — пространство характеристик наблюдаемого явления (косвенных характеристик объекта); A — оператор, ставящий каждому значению $z \in Z$ в соответствие значение $u \in U$.

Оператор A , как правило, обладает интегральным характером, т. е. его значения Az слабо чувствительны к возмущениям z . Если же внести в величину u возмущения, вызванные погрешностью измерений, то даже малые возмущения приводят к большим возмущениям z . Эта причина вызывает неустойчивость при реализации расчетных алгоритмов решения обратных задач.

Исходный алгоритм решения обратной задачи. Для построения данного алгоритма решения обратной задачи зададим номинальные значения исходных параметров λ_{nom} и r_{nom} . Выберем две точки с координатами a_1 и a_2 и по зависимости (5) для заданных значений λ_{nom} и r_{nom} вычислим ординаты в этих точках: $p_{f1}(a_1)$ и $p_{f2}(a_2)$. Зная две эти точки, принадлежащие кривой (5), можно, решая систему двух нелинейных уравнений относительно λ и r

$$\begin{aligned} p_{f1}(a_1) &= \\ &= \frac{(1/\lambda) \exp(-a_1/\lambda) [1 - \exp(-r(a_1 - a_0))]}{\int_{a_0}^{\infty} (1/\lambda) \exp(-a_1/\lambda) [1 - \exp(-r(a_1 - a_0))] da}; \\ p_{f2}(a_2) &= \\ &= \frac{(1/\lambda) \exp(-a_2/\lambda) [1 - \exp(-r(a_2 - a_0))]}{\int_{a_0}^{\infty} (1/\lambda) \exp(-a_2/\lambda) [1 - \exp(-r(a_2 - a_0))] da}, \end{aligned} \quad (7)$$

получить значения λ_{nom} и r_{nom} , по сути, решая обратную задачу.

Запишем данный исходный алгоритм в виде следующей последовательности действий.

1. Для исследуемых типовых объектов формируется исходный массив размеров обнаруженных дефектов a .

2. На основе сформированного исходного массива строится гистограмма распределения

числа обнаруженных дефектов в зависимости от его размера.

3. Для построенной гистограммы подбирается подходящее распределение плотности вероятности размеров обнаруженных дефектов $p_f(a)$.

4. Для полученной плотности распределения размеров обнаруженных дефектов $p_f(a)$ выбираются две точки с координатами a_1 и a_2 и вычисляются ординаты в этих точках $p_{f1}(a_1)$ и $p_{f2}(a_2)$.

5. Для значений a_1 , a_2 и $p_{f1}(a_1)$, $p_{f2}(a_2)$ формируется система нелинейных уравнений аналогично системе (7).

6. Решая полученную систему уравнений, получаем искомые значения параметров λ и r .

В результате решения обратной задачи с исходными номинальными значениями были получены значения параметров λ и r , равные исходным с точностью $\varepsilon = 10^{-10}$.

С учетом того, что оператор решения прямой задачи рассматриваемого класса, как указано выше, обладает интегральным характером, было исследовано влияние погрешности входных данных на выходные результаты для параметров λ и r . При этом значения a_1 , a_2 оставались равными номинальным значениям, а ординаты изменялись в соответствии с соотношениями

$$\begin{aligned} p_{f1}^{\delta}(a_1) &= p_{f1}(a_1)(1 + \delta); \\ p_{f2}^{\delta}(a_2) &= p_{f2}(a_2)(1 + \delta), \end{aligned} \quad (8)$$

где величину δ задавали от начального значения 0,001 с шагом 0,001 до тех пор, пока не получали расхождение искомого решения. При этом значения δ задавали как положительными, так и отрицательными.

Полученные в соответствии с выражениями (8) значения ординат $p_{f1}^{\delta}(a_1)$ и $p_{f2}^{\delta}(a_2)$ подставляли в уравнения (7) и решали систему нелинейных уравнений с возмущенными входными значениями. По результатам решения системы вычисляли погрешности искомого параметров λ и r относительно их номинальных значений: E_{λ}^+ — выходная погрешность для параметра λ при положительном возмущении δ ; E_r^+ — выходная погрешность для параметра r при положительном возмущении δ ; E_{λ}^- — выходная погрешность для параметра λ при отрицательном возмущении δ ; E_r^- — выходная погрешность для параметра r при отрицательном возмущении δ .

Результаты проведенного исследования представлены на рис. 2 и 3.

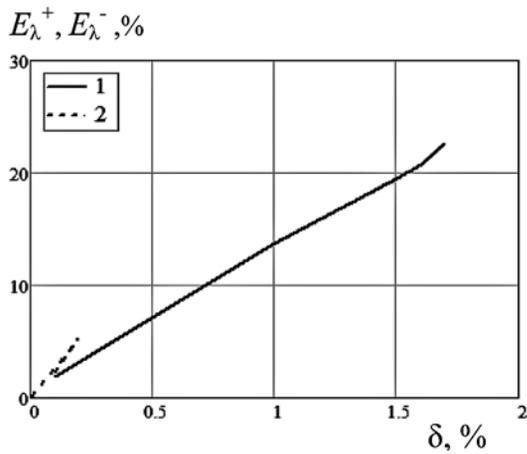


Рис. 2. Зависимость выходной погрешности для параметра λ в % от входного возмущения δ в %:
 1 — E_{λ}^{+} — выходная погрешность для λ при положительном возмущении δ ; 2 — E_{λ}^{-} — выходная погрешность для λ при отрицательном возмущении δ

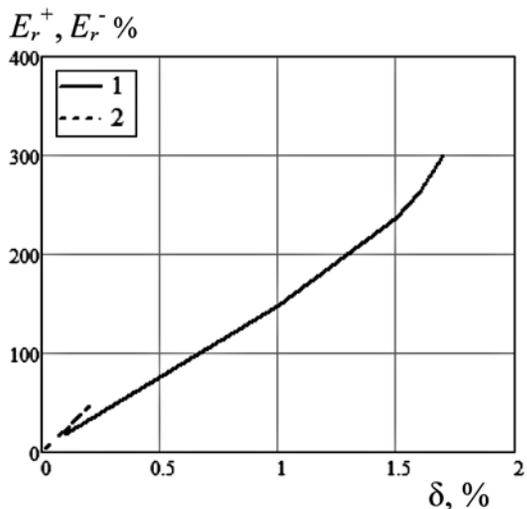


Рис. 3. Зависимость выходной погрешности для параметра r в % от входного возмущения δ в %:
 1 — E_r^{+} — выходная погрешность для r при положительном возмущении δ ; 2 — E_r^{-} — выходная погрешность для r при отрицательном возмущении δ

Как следует из представленных на рис. 2, 3 результатов, изменение выходной погрешности для обоих параметров в исследованном диапазоне практически линейно зависит от входной погрешности. Обращает на себя внимание тот факт, что наклоны линий на рисунках имеют разные углы для положительных и отрицательных входных погрешностей.

Вычисленные коэффициенты чувствительности для параметра λ составили: при положительных входных погрешностях $k_{\lambda}^{+} = 14$, при отрицательных — $k_{\lambda}^{-} = 30$. Коэффициенты чувствительности для параметра r составили: при положительных входных погрешностях

$k_r^{+} = 238$, при отрицательных — $k_r^{-} = 263$. Коэффициент чувствительности находился как отношение изменения выходного параметра к изменению входного.

Следует заметить, что начиная с входной погрешности $\delta = 1,71\%$ при расчете как параметра λ , так и параметра r алгоритм расходится, и получить решение не удается. Для отрицательной входной погрешности алгоритм расходится уже при значении $\delta = -0,21\%$.

Принимая во внимание эти результаты, можно сделать вывод, что использовать приведенный выше алгоритм для практических расчетов не представляется возможным, так как реальные погрешности измерения дефектов, как правило, составляют значения выше 5%.

Альтернативный алгоритм решения обратной задачи

В качестве альтернативы описанному выше исходному алгоритму был использован другой алгоритм, основанный на обнаруженной особенности распределений (4) и (5).

На основе анализа этих зависимостей было выявлено, что найденные экстремальные значения для нормированных функций плотности распределений вероятности обнаруженных дефектов (формула (5)) при разных коэффициентах достоверности r строго лежат на функции, описываемой экспоненциальным законом

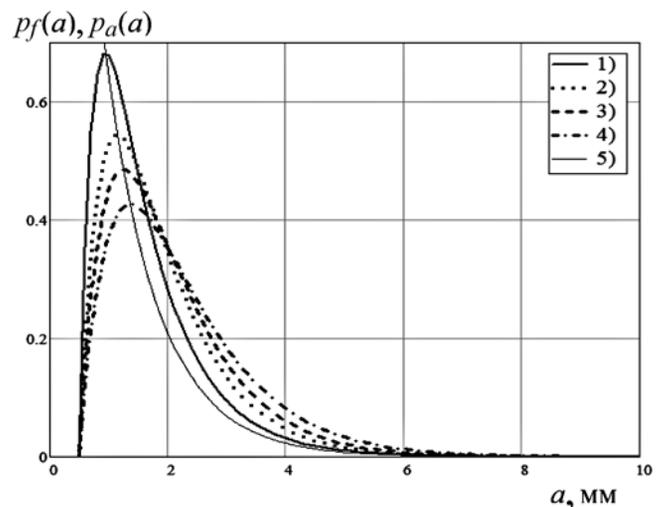


Рис. 4. Зависимость плотности распределения вероятности действительных и обнаруженных размеров дефектов для разных значений коэффициента достоверности r :
 1) $r = 3,0$; 2) $r = 1,0$; 3) $r = 0,5$; 4) $r = 0,1$; 5) действительное распределение

распределения (формула (4)), которая использовалась для описания плотности распределения вероятности действительных размеров. На рис. 4 представлены зависимости плотности распределений вероятности размеров действительных и обнаруженных дефектов при различных коэффициентах достоверности r .

Вычисленная абсцисса, соответствующая максимальному значению функции плотности распределения для обнаруженных дефектов, имеет следующее значение:

$$a_{\max} = a_0 - \ln(1/(1 + \lambda r))/r. \quad (9)$$

В результате подстановки вычисленного значения для абсциссы a_{\max} в функции (4) и (5) были получены ординаты значений этих функций, тождественно совпадающие друг с другом.

С учетом этой математической особенности используемых распределений был построен альтернативный алгоритм вычисления плотности распределения вероятности действительных размеров дефектов и коэффициентов достоверности r приборов контроля.

В результате альтернативный алгоритм решения обратной задачи можно представить в виде последовательности следующих действий:

1. Для исследуемых типовых объектов формируется исходный массив обнаруженных дефектов a .

2. На основе сформированного исходного массива строится гистограмма распределения числа обнаруженных дефектов в зависимости от его размера.

3. Для построенной гистограммы подбирается подходящая плотность распределения вероятности размеров обнаруженных дефектов $p_f(a)$.

4. Для полученной плотности распределения вероятности размеров обнаруженных дефектов $p_f(a)$ определяется абсцисса a_{\max} и значение максимальной плотности: $p_{f \max} = p_f(a_{\max})$.

5. Подставляя полученные значения $p_{f \max}$ и a_{\max} в зависимость (4), получаем нелинейное уравнение $F(\lambda)$ относительно λ :

$$F(\lambda) = p_{f \max} - \frac{(1/\lambda) \exp(-a_{\max}/\lambda)}{\int_{a_0}^{\infty} (1/\lambda) \exp(-a/\lambda) da}. \quad (10)$$

Решая уравнение (10), получаем неизвестный параметр λ^* для экспоненциального распределения действительных размеров дефектов.

6. Подставляя полученное на шаге 5 значение λ^* и известные значения $p_{f \max}$ и a_{\max} в соот-

ношение (5), получаем нелинейное уравнение относительно r :

$$G(r) = p_{f \max} - \frac{(1/\lambda^*) \exp(-a_{\max}/\lambda^*) [1 - \exp(-r(a_{\max} - a_0))]}{\int_{a_0}^{\infty} (1/\lambda^*) \exp(-a/\lambda^*) [1 - \exp(-r(a - a_0))] da} \quad (11)$$

Решая уравнение (11) относительно неизвестного значения r , получаем второй неизвестный параметр для исходной обратной задачи.

Тестовые расчеты решения обратной задачи для альтернативного алгоритма

Для исследования эффективности альтернативного алгоритма были проведены тестовые расчеты для анализа влияния погрешностей входных данных на выходные параметры λ и r . Для заданных номинальных значений параметра λ_{nom} и коэффициента достоверности r_{nom} были рассчитаны плотности распределения вероятности обнаруженных размеров дефектов. Для полученного распределения по формулам (9) и (5) были вычислены координаты экстремальной точки (a_{\max} , $p_{f \max}$). Имея полученную экстремальную точку по разработанному альтернативному алгоритму п. 1 — п. 6 были восстановлены значения неизвестных параметров λ_{nom} и r_{nom} . Точность расчета при определении номинальных значений составила $\varepsilon = 10^{-10}$.

Внесем в исходные значения координат экстремальной точки (a_{\max} , $p_{f \max}$) возмущения, имитирующие погрешности средств измерения. Погрешности были внесены как в значения абсциссы, так и в значение ординаты экстремальной точки. В результате были сформированы пять точек со следующими координатами:

- т. 0 — $T_0 = (a_{\max}, p_{f \max})$;
- т. 1 — $T_1 = (a_{\max} (1 + \delta), p_{f \max} (1 - \delta))$;
- т. 2 — $T_2 = (a_{\max} (1 + \delta), p_{f \max} (1 + \delta))$;
- т. 3 — $T_3 = (a_{\max} (1 - \delta), p_{f \max} (1 + \delta))$;
- т. 4 — $T_4 = (a_{\max} (1 - \delta), p_{f \max} (1 - \delta))$.

Здесь значения входных возмущений δ были выбраны следующими: 1 %, 5 %, 10 %, 15 % и 20 %. Расположение полученных точек относительно номинальной приведено на рис. 5.

Для анализа были исследованы также различные варианты сочетаний значений параметра λ и коэффициента достоверности r . В ра-

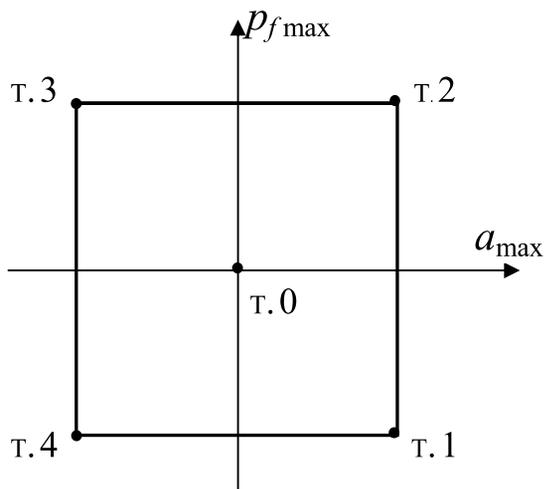


Рис. 5. Расположение точек при анализе входных возмущений

боте представлены лишь выборочные результаты для следующих вариантов:

- 1) $\lambda = 0,9$; $r = 0,35$; 2) $\lambda = 0,9$; $r = 0,20$;
- 3) $\lambda = 0,9$; $r = 0,80$; 4) $\lambda = 0,5$; $r = 0,35$,

которые приводятся в табл. 1—4 соответственно, а также на рис. 6—13. Незаполненные поля в таблицах означают отсутствие сходимости для рассматриваемого численного эксперимента.

Анализ полученных результатов показал, что зависимости результирующих погрешностей как для параметра λ , так и для коэффициента достоверности r от внесенных погрешностей входных данных близки к линейным (рис. 6—13). Для сопоставления результатов

Таблица 1

№ т.	Внесенные погрешности во входные данные				
	1 %	5 %	10 %	15 %	20 %
Результирующая погрешность для параметра λ , %					
1	4,234	—	—	—	—
2	—	—	—	—	—
3	2,784	9,878	15,692	20,242	24,151
4	13,21	44,566	74,92	103,256	131,7
Результирующая погрешность для коэффициента достоверности r , %					
1	42,899	—	—	—	—
2	—	—	—	—	—
3	32,695	142,135	278,83	438,1	636,8
4	99,475	333,64	580,52	846,3	1162,0

Таблица 2

№ т.	Внесенные погрешности во входные данные				
	1 %	5 %	10 %	15 %	20 %
Результирующая погрешность для параметра λ , %					
1	—	—	—	—	—
2	—	—	—	—	—
3	4,3	13,021	19,51	24,436	28,611
4	16,617	50,518	82,589	112,398	142,214
Результирующая погрешность для коэффициента достоверности r , %					
1	—	—	—	—	—
2	—	—	—	—	—
3	70,65	264,091	488,965	745,253	1061,0
4	199,519	596,31	1001,0	1432,0	1938,0

Таблица 3

№ т.	Внесенные погрешности во входные данные				
	1 %	5 %	10 %	15 %	20 %
Результирующая погрешность для параметра λ , %					
1	1,32	9,753	—	—	—
2	11,799	—	—	—	—
3	1,194	5,186	9,197	12,645	15,759
4	8,343	33,608	59,9	84,875	110,11
Результирующая погрешность для коэффициента достоверности r , %					
1	11,9	67,451	—	—	—
2	50,171	—	—	—	—
3	11,852	60,943	131,169	218,01	330,556
4	34,956	143,915	270,932	413,335	586,89

Таблица 4

№ т.	Внесенные погрешности во входные данные				
	1 %	5 %	10 %	15 %	20 %
Результирующая погрешность для параметра λ , %					
1	—	—	—	—	—
2	—	—	—	—	—
3	7,429	20,071	30,827	38,196	44,343
4	18,692	55,668	90,355	122,394	154,323
Результирующая погрешность для коэффициента достоверности r , %					
1	—	—	—	—	—
2	—	—	—	—	—
3	114,879	399,283	740,963	1163	1743
4	232,704	701,767	1222,0	1834	2646

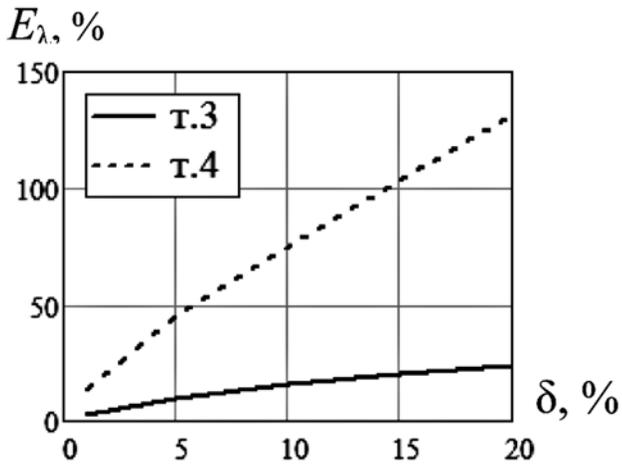


Рис. 6. Зависимость выходной погрешности E_λ в % от входного возмущения δ в % для варианта 1

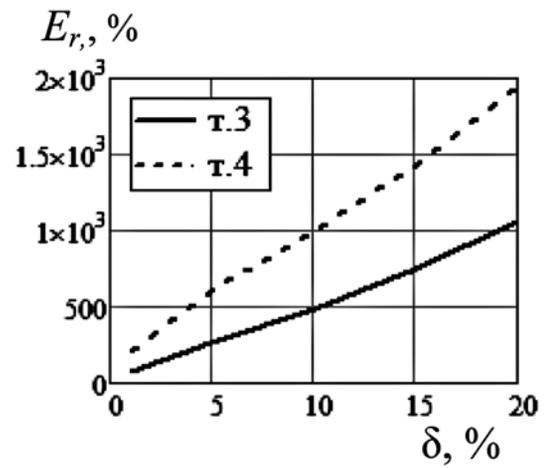


Рис. 9. Зависимость выходной погрешности E_r в % от входного возмущения δ в % для варианта 2

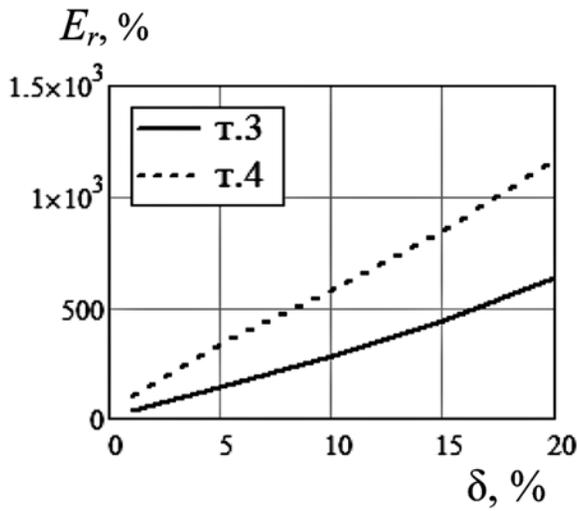


Рис. 7. Зависимость выходной погрешности E_r в % от входного возмущения δ в % для варианта 1

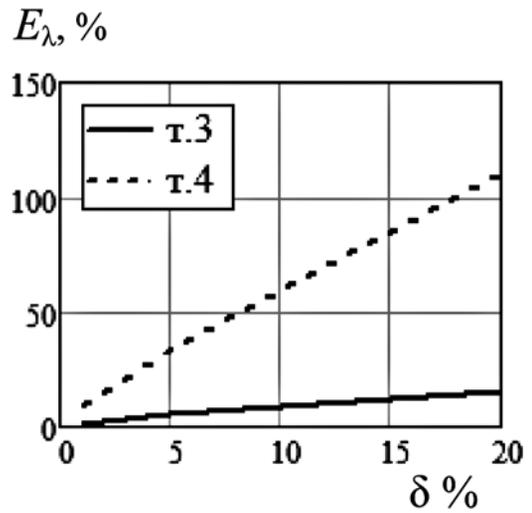


Рис. 10. Зависимость выходной погрешности E_λ в % от входного возмущения δ в % для варианта 3

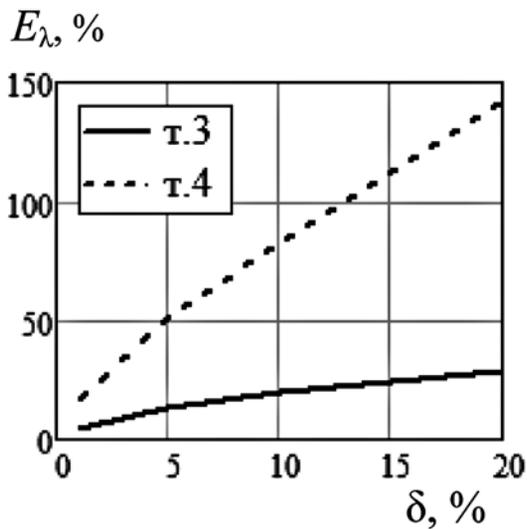


Рис. 8. Зависимость выходной погрешности E_λ в % от входного возмущения δ в % для варианта 2

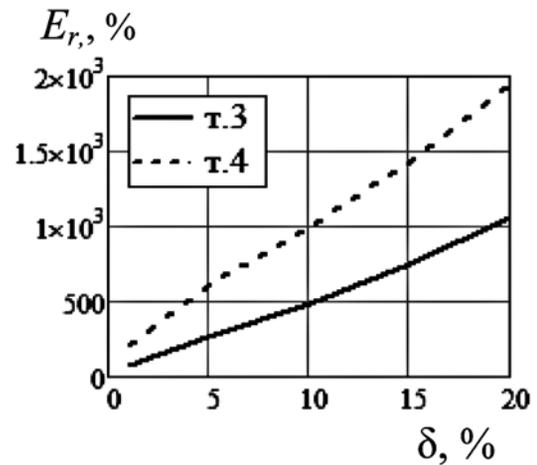


Рис. 11. Зависимость выходной погрешности E_r в % от входного возмущения δ в % для варианта 3

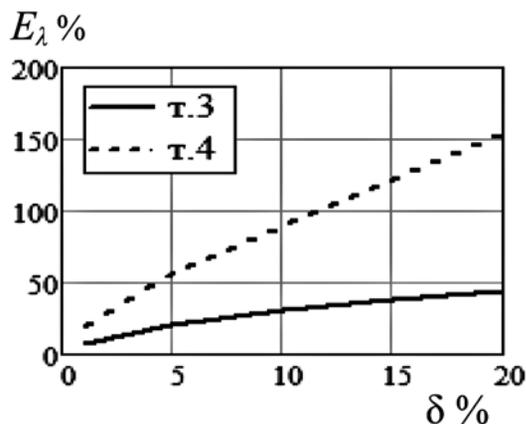


Рис. 12. Зависимость выходной погрешности E_λ в % от входного возмущения δ в % для варианта 4

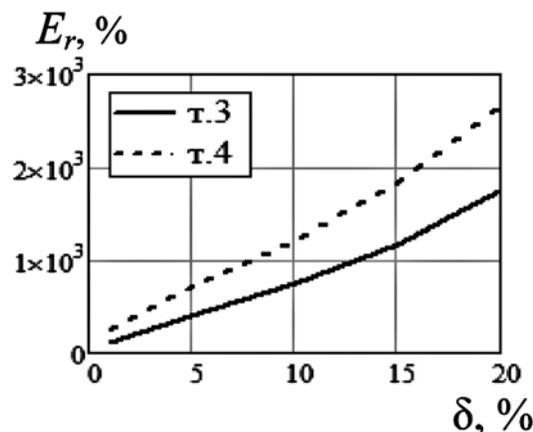


Рис. 13. Зависимость выходной погрешности E_r в % от входного возмущения δ в % для варианта 4

различных вариантов были вычислены средние значения коэффициентов чувствительности для всей последовательности входных возмущений, которые приведены в табл. 5.

По результатам проведенных численных экспериментов можно сделать следующие выводы.

1. Сопоставление значений результирующих погрешностей параметров λ и r для одноименных точек и всех вариантов показывает, что

1) погрешности для λ в десятки раз меньше, чем для r ;

2) коэффициенты чувствительности к входным ошибкам уменьшаются с увеличением коэффициента достоверности r ;

3) с уменьшением параметра λ коэффициенты чувствительности к входным ошибкам увеличиваются;

4) отношение погрешностей параметров r к λ для одноименных точек меняется не суще-

ственно: от 10 до 40 для точки 3, и от 4 до 17 для точки 4.

2. Альтернативный алгоритм более устойчив к входным погрешностям по сравнению с исходным: для альтернативного алгоритма устойчивость к входным погрешностям была обеспечена даже при 20%-ной погрешности входных данных, для исходного алгоритма — только при 1,71%-ной погрешности.

3. Для альтернативного алгоритма результирующие погрешности более чем в 7 раз меньше, чем для исходного (коэффициенты чувствительности для варианта 1 для альтернативного алгоритма в т. 3 составили $k_\lambda = 1,777$; $k_r = 30,01$; для исходного алгоритма $k_\lambda^+ = 14$, $k_r^+ = 238$).

Заключение

Разработан альтернативный алгоритм восстановления действительных размеров дефектов и характеристик средств измерения (функции вероятности обнаружения дефекта заданного размера) по результатам проводимого контроля металлов на основе выявленной особенности выбранных функций распределения обнаруженных и действительных размеров дефектов. Для разработанных исходного и альтернативного алгоритмов проведено численное исследование по влиянию входных возмущений на результаты восстанавливаемых параметров.

Представленный в работе альтернативный алгоритм восстановления действительных размеров дефектов и характеристик средств измерения обладает меньшей чувствительностью к возмущающим входным погрешностям

Таблица 5

№ т.	Коэффициенты чувствительности к погрешностям входных данных для параметра λ			
	Вариант 1 $\lambda = 0,9$; $r = 0,35$	Вариант 2 $\lambda = 0,9$; $r = 0,20$	Вариант 3 $\lambda = 0,9$; $r = 0,80$	Вариант 4 $\lambda = 0,5$; $r = 0,35$
т. 3	1,777	2,383	0,956	3,858
т. 4	8,617	9,917	6,444	10,947
Коэффициенты чувствительности к погрешностям входных данных для коэффициента достоверности r				
т. 3	30,01	55,02	13,644	86,703
т. 4	67,755	122,25	29,546	149,964

по сравнению с исходным алгоритмом, более устойчив к входным погрешностям и позволяет получить более достоверные данные для построения математических моделей расчета и прогнозирования уровня безопасности энергетических объектов.

Список литературы

1. **Отчет** о верификации программного средства "Прогноз_Р" для ПЭВМ по расчету вероятности хрупкого разрушения корпуса реактора РУ ВВЭР-1000. Рег. номер № 01201460046. МГУПИ. 2014 г. 220 стр.

2. **Гетман А. Ф., Козин Ю. Н.** Неразрушающий контроль и безопасность эксплуатации сосудов и трубопроводов давления. М.: Энергоиздат, 1997. 288 с.

3. **Волченко В. Н.** Вероятность и достоверность оценки качества металлопродукции. М.: Metallургия, 1987. 88 с.

4. **Ткачев В. В., Желтухин К. К.** Анализ вероятности разрушения трубопроводов и оборудования при продлении срока эксплуатации АЭС // Атомная энергия. 2008. Т. 104, Вып. 5. С. 284—290.

5. **Тихонов А. Н., Кальнер В. Д., Гласко В. Б.** Математическое моделирование и метод обратных задач в машиностроении. М.: Машиностроение, 1990. 264 с.

6. **Nichols R., Grutzen S., Miller A.** PISC III: Status report // Proceed. Third International Conference on Material Science Problems in NPP Equipment Production and Operation. 17—19 June 1994. М.—S.—Pb.: Prometey—IAEA, 1994. Vol. 1. P. 1—8.

A. E. Aleksandrov, Ph. D., Professor, e-mail: femsystem@yandex.ru,
Moscow Technological University (MIREA) Moscow, 107996, Russian Federation

Evaluation of the Reliability of Metal Testing Results Based on an Alternative Algorithm

The problem was set to reconstruct the characteristics of the instrument for non-destructive testing and justify the initial defectiveness for using it in construction of mathematical models for calculating and predicting the level of safety of energy facilities. A description of the solution of the direct problem for correcting the distribution of detected defects and constructing their actual distribution is given. A method for solving the inverse problem is developed to reconstruct the characteristics of the instrument used to measure the defectiveness of the objects under investigation on the basis of an alternative algorithm. A technique is proposed for processing the results of defectiveness in order to justify its use in calculating and predicting the level of safety of energy facilities. An alternative algorithm for reconstructing the actual size of the defects and the characteristics of the measuring instruments (the probability function for detecting a defect of a given size) on the results of the performed metal testing is developed on the basis of the revealed feature of the selected distribution functions of the detected and actual defect sizes. For the developed original and alternative algorithms, a numerical study of the effect of input disturbances on the results of the reconstructed parameters is conducted. The alternative algorithm for reconstructing the actual sizes of defects and characteristics of measuring instruments presented in this paper has a lower sensitivity to perturbing input errors than the original algorithm, is more resistant to input errors, and allows obtaining more reliable data for constructing mathematical models for calculating and predicting the safety level of energy objects.

Keywords: probabilistic safety analysis, initial defectiveness, probabilistic methods of fracture mechanics, solution of inverse problems

DOI: 10.17587/it.24.529-537

References

1. **Report** on the verification of the software "Prognoz_R" for PC on the calculation of the probability of brittle fracture of the reactor vessel of VVER-1000. Registration number № 01201460046. MGUPI, 2014.

2. **Getman A. F., Kozin Yu. N.** *Nerazhrushausshuj kontrol' i bezopasnost' sosudov i truboprovodov davlenija* (Non-destructive testing and safety of operation of vessels and pressure pipelines), Moscow, Energoizdat, 1997 (in Russian).

3. **Volchenko V. N.** *Verojatnost' i dostovernost' ocenki kachestva metalloprodukcii*. (Probability and reliability of the evaluation of the quality of metal products), Moscow, Metallurgy, 1987 (in Russian).

4. **Tkachev V. V., Zheltukhin K. K.** Analysis of the failure probability of pipelines and equipment with service life extension of nuclear power plants, *Atomic Energy*, 2008, vol. 104, no. 5, pp. 372—380.

5. **Tikhonov A. N., Kalner V. D., Glasko V. B.** *Matematicheskoe modelirovanie i metod obratnyh zadach v mashinostroenii* (Mathematical modeling and the method of inverse problems in engineering), Moscow, *Mechanical Engineering*, 1990 (in Russian).

6. **Nichols R., Grutzen S., Miller A.** PISC III: Status report, *Proceed. Third International Conference on Material Science Problems in NPP Equipment Production and Operation*, 17—19 June 1994, 1994, vol. 1, pp. 1—8.

Ле Ба Чунг, аспирант, e-mail: chungbaumanvietnam@gmail.com,
Московский физико-технический институт (государственный университет),
Ю. А. Холопов, вед. инженер,
Институт точной механики и вычислительной техники им. С. А. Лебедева РАН

Информационная среда цифровых систем управления, использующая для управления периферией концепцию "вынесенной руки"

Предложена архитектура цифровой централизованной системы управления, построенной по принципу "вынесенной руки". Эта архитектура предоставляет возможность упростить состав оконечных устройств до автоматного уровня и исключить тем самым необходимость их изменения при модернизации системы. Периферийные устройства теперь превращаются в циклические автоматы, выполняющие простые функции нижнего уровня: измерения и обработки управляющих воздействий.

Ключевые слова: информационная среда цифровых систем управления, распределенная и централизованная системы управления, периферийные устройства, УСО, концепция "вынесенной руки"

Введение

В большинстве цифровых систем управления (ЦСУ) алгоритмы управления объектами исполняются в программно управляемых вычислителях. Вычислители, кроме чисто алгоритмических расчетов, выполняют функцию управления устройствами ввода параметров состояния и вывода параметров управления объектом, т. е. задачу коммуникации вычислителя с устройствами сопряжения с объектом (УСО). Последняя задача может быть решена несколькими способами, один из которых, а именно управление периферийными устройствами вычислителя в режиме "вынесенной руки", рассмотрен далее.

Необходимо отметить, что в ситуации, когда производительность программно-управляемых вычислителей постоянно растет и подается изготовителями микропроцессоров как основной критерий эффективности вычислителя, разработчики алгоритмов управления не уделяли должного внимания вопросам комплексирования вычислителя и периферии (датчиков и исполнительных устройств) способами, отвечающими требованиям задач управления. Комплексирование периферийных устройств, за редким исключением, происходит в режиме программного управления разовыми операциями опроса состояния датчиков или передачи параметров в исполнительные устройства. Для выделения круга проблем, связанных с технологией эффективного объединения составных

частей ЦСУ в единую систему, авторами используется термин "информационная среда" цифровых систем управления. Элементами информационной среды (ИС) являются все составные части ЦСУ, участвующие в цепочках информационных взаимодействий объекта управления (ОУ) с программно-управляемыми вычислителями системы управления, и их подсистемы ввода-вывода, т. е. в ИС входят: интерфейсы и логика управления датчиками, центральный вычислитель, интерфейсы и логика управления исполнительными устройствами, коммутаторы и совокупность физических связей составных частей.

Постоянное усложнение задач и самих объектов управления требует поиска новых способов построения ИС цифровых систем управления, которые позволили бы уточнять и дополнять задачи управления на уже функционирующих объектах. В более широкой постановке речь идет о создании цифровых систем управления для объектов с вариативной моделью функционирования. В аналоговых системах управления такая ситуация однозначно требует изменения аппаратуры. В цифровых системах управления возможности по разработке универсальных решений гораздо шире. Поэтому авторами была поставлена задача разработки ИС цифровых систем управления, которая не зависит от алгоритмов управления.

В статье обосновывается возможность создания информационной среды ЦСУ для объектов с вариативной моделью функциони-

рования. Искомая технология предполагает изначальное отсутствие ориентации на конкретный класс задач управления и привязки архитектурных решений к конкретной модели функционирования объекта управления.

1. Типовой подход к построению цифровых распределенных систем управления

Современные задачи управления техническими объектами в промышленности, робототехнике, авиации и т. д. часто реализуются в системах с распределенной функцией управления. Такие системы традиционно называют распределенными системами управления (PCY). Предыдущие поколения PCY строились на основе аналоговых узлов. В аналоговых системах управления существует проблема с комплексированием большого числа периферийных устройств (ПУ) к вычислителю. Поэтому было принято разбивать общую задачу управления на несколько слабосвязанных подзадач, т. е. распределять функции управления по нескольким аналоговым вычислителям [1].

Переход от аналоговой к цифровой обработке информации сохранил эту концепцию построения PCY, но придал ей новые качества: типизацию архитектурных и схемотехнических решений, возможность использования единообразного стиля управления целым рядом разнотипных периферийных устройств, легкую масштабируемость систем управления и возможность помехоустойчивых обменов информацией между пространственно удаленными узлами системы. Настройка и наладка цифровых систем управления в существенной части сводится к отладке программного обеспечения, для чего используются давно отработанные технологии. Вместе с тем, рациональность сохранения иерархической структуры системы управления, реализованной на элементной базе с принципиально иными свойствами, никем не подвергается сомнению.

Рассмотрим обобщенную структуру цифровой распределенной системы управления — системы, построенной по принципу разбиения общей задачи управления на

несколько отдельных, слабосвязанных между собой подзадач (рис. 1).

Возможность разбиения общей задачи управления на слабосвязанные подзадачи объясняется наличием в сложных объектах управления множества степеней свободы. Управление по каждой степени свободы относительно независимо от управления по другим степеням свободы. Поэтому подзадачи управления являются слабосвязанными.

В качестве примера рационального применения распределенной ЦСУ можно привести реализацию задачи управления многозвенным манипулятором. Манипулятор может управляться совокупностью распределенных вычислителей, каждый из которых выполняет почти автономную задачу управления положением звена относительно соседнего — опорного. Согласованная работа таких контуров регулирования нижнего уровня (КРНУ) задается уставками из центрального вычислителя, выполняющего задачу расчета и распределения уставок по подчиненным вычислителям нижнего уровня. Вычислители объединяются в единую ИС цифровой системы управления сетевыми или сетеподобными интерфейсами, причем большой скорости передачи данных не требуется, так как в вычислители нижнего уровня передают лишь команды — уставки, например скорость углового движения в сочленении.

В процессе эксплуатации ЦСУ часто возникает необходимость коррекции алгоритмов управления. Кроме того, при отладке систем алгоритмы управления объектом отлажива-

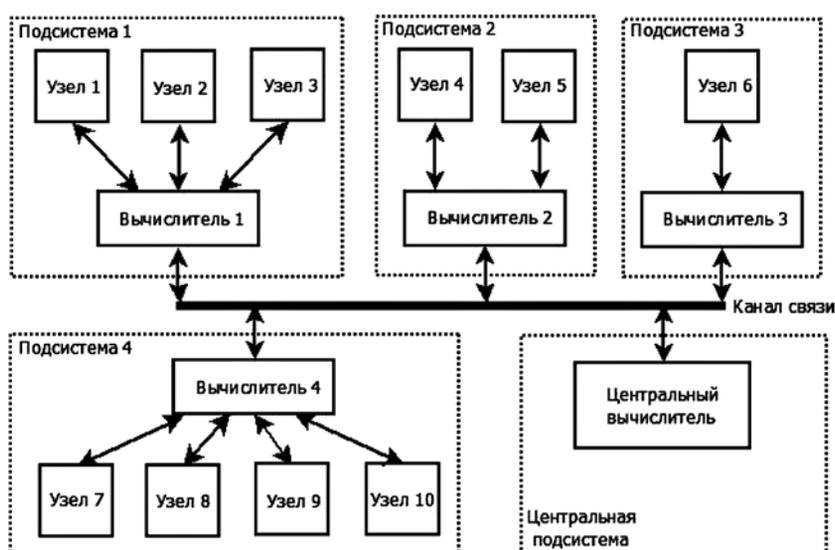


Рис. 1. Структура цифровой распределенной системы управления

ются поэтапно. Сначала происходит отладка функций узлов нижнего уровня до их установки на объекте управления и с использованием для перепрограммирования технологических интерфейсов. Общие алгоритмы управления объектом отлаживаются после монтажа вынесенных узлов КРНУ на их штатные, часто труднодоступные, места. Изменение алгоритмов работы узлов на этом этапе становится невозможным из-за недоступности технологических интерфейсов.

Зафиксируем основные характеристики ИС цифровых распределенных систем управления:

- ✓ модель объекта и алгоритмы управления заранее определены и согласованы со структурой ИС цифровых распределенных систем управления;
- ✓ функция управления распределена и реализована как в центральном вычислителе, так и в контурах регулирования нижнего уровня;
- ✓ скорость сетевой коммуникационной среды, передающей короткие высокоуровневые команды управления и параметры состояния, относительно низкая.

Далее попробуем показать, что для современных задач и объектов управления подход к построению цифровых систем управления, основанный на разбиении общей задачи на несколько локальных, слабосвязанных подзадач, может быть недостаточно универсальным и, как минимум, недостаточно гибким.

2. Возможность изменения алгоритмов управления в цифровых распределенных системах управления

В период использования аналоговых систем управления сложность объектов управления ограничивалась возможностью иерархического подхода к построению систем управления, который являлся единственным способом построения систем управления для сложных объектов. Технологии цифровой обработки и передачи информации принципиально не требуют разбиения общей задачи управления на подзадачи, так как ограничения по подключению практически любого числа устройств сопряжения с ОУ отсутствуют, нет ограничений и в скорости отработки алгоритмов управления. Вместе с тем, разбиение общей задачи управления на локальные слабосвязанные подзадачи снижает темп управления из центрального

вычислителя. Это свойство позволяло на этапе перехода с аналоговой на цифровую элементную базу использовать стандартные низкоскоростные интерфейсы для комплексирования периферийных устройств к вычислителю в цифровых распределенных системах управления. В распределенных ЦСУ часть алгоритмов управления реализована в контурах регулирования нижнего уровня. Изменение алгоритмов управления может затрагивать КРНУ, что является отдельной технической и, часто, сложной организационной задачей, так как узлы КРНУ, фактически, подключены к центральному вычислителю как удаленные внешние устройства и производятся, как правило, сторонними организациями. Для разработчиков современных ЦСУ крайне желательно выбрать структуру информационной среды ЦСУ, которая позволяла бы легко менять алгоритмы управления любого уровня.

Обоснованно низкая рабочая скорость сетевой коммуникационной среды в РСУ не позволяет реализовать управление объектом, например, в случае, когда движения частей (звеньев) многозвенного манипулятора стало необходимо выполнять в режиме жесткой связности задач регулирования, с взаимным учетом состояния смежных звеньев. Такая необходимость возникает, например, при отладке функционирования ОУ в критичных режимах и при работе нескольких исполнительных устройств на общую нагрузку. Недостаточная скорость обмена данными между КРНУ может стать причиной "грубости" траекторных движений конечных звеньев многозвенного манипулятора. Решение этой задачи возможно при переносе всех задач регулирования в центральный вычислитель, с организацией взаимодействия между ними самым быстрым способом — через общую память.

В настоящее время существуют перспективные объекты управления, для которых на этапе проектирования системы управления модель функционирования неизвестна, поэтому разбиение общей задачи управления на отдельные подзадачи невозможно. Алгоритмы управления такими объектами разрабатываются итеративно, при поэтапном введении ОУ в эксплуатацию, и поэтому постоянно меняются. Примерами являются системы автономного адаптивного управления, методология решения которых разработана д. ф-м. н., профессором А. А. Ждановым [2—4]. Такой класс задач тоже относится

к задачам управления объектом с вариативной моделью функционирования.

Еще одним из типов объектов с неопределенной моделью управления являются объекты с переменной структурой. У таких объектов либо у их рабочей среды частично меняются характеристики в результате различных процессов: деградации узлов при постоянном составе (например, появляются люфты в механических системах); накопления паразитных зарядов в электронных системах; накопления механической, электронной, физической ошибок во взаимодействии объекта и среды с системой управления; вследствие изменения состава ОУ, которое изначально не было предусмотрено, например, выхода из строя части датчиков или исполнительных устройств из-за повреждения объекта. Часть таких изменений ОУ можно компенсировать за счет коррекции алгоритмов управления, но информационная среда ЦСУ должна позволять изменять алгоритмы управления.

В сложных робототехнических комплексах при изменении условий и целей функционирования вариативность состава объекта управления и свойств его частей также требует создания ИС, позволяющей изменять алгоритмы управления без перестройки самой ИС.

Поиск нового подхода к организации унифицированной ИС цифровых систем управления велся в трех направлениях:

- ✓ фиксация определенностей в функционировании составных частей ИС цифровых систем управления;
- ✓ организация ИС, адаптивной к различным структурным решениям алгоритмов управления и не требующей жесткой настройки под конкретное алгоритмическое управление;
- ✓ поиск технологий, обеспечивающих необходимую связность элементов искомой информационной среды.

По мнению авторов, правильным подходом к построению ИС с корректируемыми алгоритмами управления является перенос всех алгоритмов управления из контуров регулирования нижнего уровня в центральный вычислитель. Возможность построения централизованной ИС для цифровых систем управления объектами и технология ее реализации пояснены далее.

3. Типовые функции информационной среды цифровых систем управления

В цифровых системах управления события взаимодействия вычислителя с периферийным окружением не случайны. Они носят упорядоченный, циклический характер. Функциональная последовательность информационных событий в ЦСУ показана на рис. 2.

Согласно рис. 2 начало цикла регулирования — измерительная фаза или фаза ввода параметров, в которой происходит фиксация состояния ОУ в датчиках, преобразование аналоговых параметров в цифровые коды — набор параметров состояния. За ней следует фаза передачи набора состояния в память центрального вычислителя. Затем наступает расчетная фаза, т. е. вычисление управляющих воздействий на ОУ. После формирования в памяти вычислителя набора параметров управления его компоненты должны быть доставлены через коммуникационную среду к исполнительным устройствам. Цикл управления заканчивает фаза отработки новых управляющих воздействий. Вышеперечисленные функции или фазы повторяются в каждом цикле управления с постоянным периодом — периодом регулирования.

Очередной цикл управления логично начинать после формирования реакции объекта управления на управляющие воздействия предыдущего цикла, поэтому имеет место



Рис. 2. Функциональная последовательность событий в цифровой системе управления

длительная пауза между окончанием выдачи уставок на исполнительные устройства и началом измерительной функции следующего цикла регулирования. Данная пауза обусловлена инерционными свойствами ОУ.

В каждом цикле управления центральный вычислитель должен осуществить опрос датчиков и доставить параметры управления в исполнительные устройства. Это означает, что набор задач ввода-вывода параметров статичен, задачи повторяются в каждом цикле регулирования и функционируют независимо от алгоритмов управления. Иными словами, режим автономной работы периферийных устройств принципиально возможен, и остается определить способ их подключения к центральному вычислителю, а также механизм синхронизации обменов параметрами между периферией и памятью вычислителя.

4. Цифровая централизованная система управления, построенная по принципу "вынесенной руки"

Как было сказано выше, функции измерения, расчета и выдачи уставок присутствуют в любой системе управления независимо от структуры вычислительной среды. Набор этих функций ЦСУ статичен и привязан к составу периферии, в то время как вычислительная функция может быть реализована как в централизованном вычислителе, так и в распределенных, локальных вычислительных ресурсах.

В распределенных системах управления функции измерения, локального управления и выдачи уставок сосредоточены в удаленном узле, реализующем функции КРНУ. В централизованной ЦСУ, на уровне удаленного узла, может и должен быть сохранен только обязательный, минимальный набор автоматных функций, реализующих ввод-вывод параметров состояния и управления. Отработка всех, возможно переменных, алгоритмов управления перекладывается на центральный вычислитель. Локальные задачи управления (задачи регулирования) теперь могут иметь сильную связность, так как все они исполняются в одном вычислителе.

Существенным изменением ИС, необходимым для реализации централизованного управления, является необходимость повышения скорости обмена информацией между периферией

и центральным вычислителем. Межмодульный интерфейс как раз и был разработан для выполнения этой задачи. Задачу синхронизации обменов параметрами между периферией и памятью вычислителя также решает межмодульный интерфейс [5]. Механизм синхронизации нацелен на достижение максимальной скорости циклических обменов параметрами.

Статичный набор простых обязательных функций уровня конечного узла РСУ и их циклический характер позволяют свести функции конечных узлов до автоматных, циклических процедур с чисто аппаратной реализацией в виде жесткой логики. Узлы РСУ упрощаются до автоматов, осуществляющих измерения и отработку уставок от уровня широтно-импульсной модуляции и ниже, например коммутации обмоток в бесколлекторных двигателях.

Результат исследований по определению принципа построения архитектуры универсальной информационной среды ЦСУ, представленный ниже, назван авторами концепцией, или моделью, "вынесенной руки" (рис. 3).

Идеологически концепция "вынесенной руки" основана на том, что время передачи параметров управления объектом из центрального вычислителя (ЦВ) намного меньше времени реакции ОУ на их изменение. При достаточной скорости обменов возможность управления удаленным объектом или его разнесенными составными частями в реальном времени не зависит от удаленности конечных устройств относительно центрального вычислителя. Передача сигналов через межмодульный интерфейс занимает достаточно малое время в цикле управления, поэтому можно не обращать внимания на задержку в каналах.

Концепция, или модель, "вынесенной руки" была выбрана как принцип и обозначение способа комплексирования периферии ЦСУ к ЦВ из следующих соображений.

Наиболее простым способом обмена параметрами для задач управления, исполняемых в ЦВ, является способ прямого подключения или отображения параметров в его адресное пространство как в ячейки памяти. Такой способ применим лишь для периферии ЦСУ, расположенной рядом с ЦВ, и не пригоден для периферийных устройств, распределенных по объекту управления, а это наиболее распространенный вариант размещения элементов ЦСУ. Распределенные ПУ подключаются к ЦВ как удаленные

устройства [6], с использованием каналов связи — интерфейсов и программно-управляемых процедур взаимодействия через каналы. Суть принципа "вынесенной руки" состоит в построении технологии взаимодействия ЦВ с удаленными ПУ как с ПУ, непосредственно подключенными в его адресное пространство, причем для организации такого подключения счетный ресурс ЦВ не используется. Скорость каналов связи выбирается такой, чтобы в системе не была заметна эта "удаленность", а взаимодействие центрального вычислителя с оконечными устройствами происходило бы в режиме непосредственного управления, т. е. с прямым отображением параметров управления и состояния в память центрального вычислителя.

Элементы концепции "вынесенной руки" были использованы в модели организации обменов интерфейса CAN [7]. В нем аппаратно поддержана процедура вхождения в связь и передача данных, которая осуществляется не в виде потока байтов, а пакетом, т. е. в структурированном виде, приближенном к формату размещения данных в нескольких смежных словах адресного пространства. Как и межмодульный интерфейс, CAN эффективно использует пропускную способность линий связи, так как способ вхождения CAN-абонента в связь является бесконфликтным, и наложения сообщений от нескольких абонентов в линии связи не возникает. Оптимальный для реализации межсоединений в ЦСУ межмодульный интерфейс имеет два существенных отличия от известных стандартных интерфейсов. Он ра-

ботает в автономном активном режиме и принципиально не требует синхронизации процесса передачи данных с источником и приемником информации, что радикально упрощает управление взаимодействием ЦВ и удаленных периферийных устройств.

В цифровых системах управления, построенных по концепции "вынесенной руки", (ЦСУВР) средства ввода-вывода параметров по-прежнему размещаются в удаленных оконечных устройствах, непосредственно взаимодействующих с частями объекта управления и окружающей средой. Из-за реализации режима непосредственного подключения к ЦВ вынесенные оконечные устройства могут выполнять теперь только функции ввода-вывода параметров, а вся переменная алгоритмическая часть, связывающая параметры состояния и управления, переносится в центральный вычислитель (далее — "центр"). В ЦСУВР изменение алгоритма в любой подсистеме управления проводится только в "центре".

Из рис. 3 видно, что все вынесенные оконечные устройства примерно одинаковы по интерфейсу и сходны по исполнению. Их состав минимален и включает в себя: трансивер межмодульного интерфейса, измерительную и исполнительную подсистемы, а также блок быстросрабатывающей автономной защиты от критичных режимов. Статичность набора функций оконечных устройств и простоту их функционального состава можно рассматривать как обоснование возможности реализации их в виде универсальных микросхем для подключе-

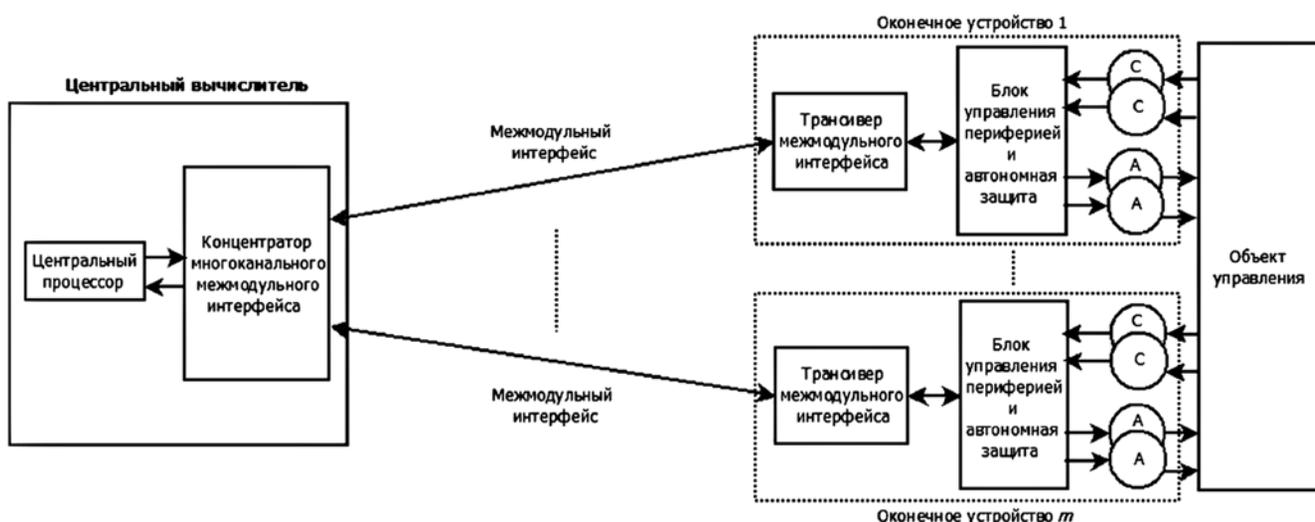


Рис. 3. Структура цифровой централизованной системы управления, построенной по принципу "вынесенной руки"

ния широкого класса периферии в ЦСУ, организованных по принципу "вынесенной руки".

В обязательный состав оконечных устройств должны быть включены блоки быстродействующей автономной защиты от критичных режимов. Реализация защиты такого типа через "центр" может оказаться невозможной из-за критичности дополнительных задержек в междуровневых связях. Поэтому функции автономной защиты реализуются непосредственно в оконечных устройствах. В качестве примера можно привести два типа защиты в бесколлекторных двигателях: от температуры и от токовой перегрузки. Температура меняется достаточно медленно, поэтому реагировать на изменение температуры можно из "центра". Вместе с тем, из-за случайного воздействия (например, блокировки подвижности исполнительных органов) ток потребления может резко возрасти до критичного значения. Реализация токовой защиты через "центр" оказывается невозможной, так как узел успеет сгореть до поступления сигнала отключения из-за задержек в канале связи. Поэтому защита от токовой перегрузки должна быть реализована локально, а состояние защиты необходимо отобразить в "центр" для принятия решения о режиме послеаварийной работы.

5. Свойства оконечных устройств в цифровых системах управления, построенных по принципу "вынесенной руки"

Как было показано выше, в информационной среде ЦСУВР простоту оконечных устройств можно довести до такой степени, что в них будут реализованы только функции нижнего уровня: измерения и обработки управляющих воздействий.

Оконечные устройства в ЦСУВР имеют следующие характеристики:

- ✓ минимальный и фиксированный набор функций нижнего уровня;
- ✓ возможность взаимодействия с "центром" через однотипный унифицированный модульный интерфейс;
- ✓ отсутствие функции управления на периферийном уровне, из-за чего не требуется перепроектировать периферийные устройства при модернизации системы;
- ✓ функциональная простота оконечных устройств, позволяющая реализовать их в виде

цифровых циклических автоматов с жесткой логикой;

- ✓ малое энергопотребление периферийных устройств, делающее возможным их питание по сигнальным линиям;
- ✓ высокая надежность работы функционально простых оконечных устройств.

Для центрального вычислителя ЦСУВР вынесенные периферийные устройства становятся управляемыми, активными и наблюдаемыми с задержками, соизмеримыми с непосредственным подключением к центральному вычислителю.

Заключение

Архитектура "вынесенной руки" разработана для построения информационной среды ЦСУ для объектов с вариативной моделью функционирования. Предложенная информационная среда ЦСУВР позволяет не только изменять алгоритмы управления, не затрагивая периферийную подсистему, но и упростить до автоматного уровня логику управления периферийными устройствами — узлами подключения к ЦСУ датчиков и исполнительных устройств. ЦВ обменивается данными с оконечными устройствами через неуправляемый, автономный, унифицированный интерфейс, на максимальной скорости, с фиксированным временем доставки данных.

Опишем архитектуру ЦСУВР в терминах ее отличия от стандартной архитектуры цифровой распределенной системы управления. Отличие состоит:

- в изменении модели организации вычислений в распределенной ЦСУ: от набора распределенных слабосвязанных вычислителей, назначенных по степеням свободы объекта, к централизованной модели, с возможной сильной связностью контуров регулирования нижнего уровня;
- в переносе всех составных частей алгоритмов управления из контуров регулирования нижнего уровня в мощный центральный вычислитель;
- в исключении из ЦВ функций управления периферийными устройствами ЦСУ;
- в упрощении оконечных устройств до уровня циклических автоматов с простым набором функций.

Список литературы

1. **Le Ba Chung, Holopov Y. A.** Information environment for neural-network adaptive control system // Springer. 09/2017. V. 736. С. 59—64.
2. **Жданов А. А.** Автономный искусственный интеллект (монография). М.: БИНОМ. Лаборатория знаний. 2008. 358 с.
3. **Жданов А. А.** О методе автономного адаптивного управления // Нейрокомпьютеры и их применение. Материалы Международной научной молодежной школы. Таганрог: Издательство ТТИ ЮФУ. 2007. С. 20—67.

4. **Жданов А. А., Преображенский Н. Б., Холопов Ю. А., Степанян И. В., Нгуен Хыу Чунг.** Аппаратная реализация нейронной сети в адаптивной системе управления // Нейрокомпьютеры: разработка, применение. 2016. № 6. С. 55—62.
5. **Ле Ба Чунг, Холопов Ю. А.** Асимметричный межмодульный интерфейс // Информационные технологии. 2/2018. № 2 (24). С. 138—143.
6. **Глушков В. М., Амосов Н. М., Артеменко И. А.** Энциклопедия кибернетики (в 2-х тт.). 1974. 1226 с.
7. **Та Дык Ань.** Controller area network (CAN) // PFIEV. 2006. 36 с.

Le Ba Chung, Ph. D. student, E-mail: chungbaumanvietnam@gmail.com,
Moscow Institute of Physics and Technology (State University),
Y. A. Holopov, Leader Engineer,
Lebedev Institute of Precision Mechanics and Computer Engineering

Information Environment of Digital Control Systems, Using the Concept of "Outstretched Hand" to Control Peripherals

The possibility of transformation from distributed control systems into systems with centralized control function is considered. The architecture of the digital centralized control system built on the principle of "outstretched hand" is proposed. This architecture makes it possible to simplify the structure of the terminals to the automatic level, and thus eliminate the necessity to change these devices when upgrading the system. Peripherals are now turning into cyclic automatons performing simple low-level functions: measurement and implementation of control actions. For the central computer, the issued terminals become self managed, active, and observed with delays, which are commensurable with the delays of direct connection to the central computer.

Keywords: information environment of digital control systems, distributed and centralized control systems, peripherals, USO, the concept of "outstretched hand".

DOI: 10.17587/it.24.538-545

References

1. **Le Ba Chung, Holopov Y. A.** Information environment for neural-network adaptive control system, Springer, 09/2017, vol. 736, pp. 59—64.
2. **Zhdanov A. A.** *Avtonomnyj iskusstvennyj intellekt* (Autonomous artificial intelligence), Moscow, BINOM. Laboratorija znanij, 2008, 358 p. (in Russian).
3. **Zhdanov A. A.** O metode avtonomnogo adaptivnogo upravlenija (About the method of autonomous adaptive control), *Nejrokomп'jutery i ih primeneniye. Materialy Mezhdunarodnoj nauchnoj molodezhnoj shkoly*, Taganrog, Publishing house of TTI JuFU, 2007, pp. 20—67 (in Russian).

4. **Zhdanov A. A., Preobrazhenskij N. B., Holopov Y. A., Stepanjan I. V., Nguyen Huu Trung.** Apparatsnaja realizacija nejronnoj seti v adaptivnoj sisteme upravlenija (Hardware implementation of a neural network in an adaptive control system), *Nejrokomп'jutery: Razrabotka, Primeneniye*, 2016, no. 6, pp. 55—62 (in Russian).
5. **Le Ba Chung, Holopov Y. A.** Asimmetrichnyj mezhmodul'nyj interfejs (Asymmetrical inter-module interface), *Informacionnye Tehnologii*, 2/2018, no. 2 (24), pp. 138—143 (in Russian).
6. **Glushkov V. M., Amosov N. M., Artemenko I. A.** *Jenciklopedija kibernetiki* (Encyclopedia of Cybernetics), 1974, 1226 p. (in Russian).
7. **Ta Duc Anh.** Controller area network (CAN), *PFIEV*, 2006, 36 p.

А. Г. Абрамов, канд. физ.-мат. наук, зам директора, доц., e-mail: abramov@runnet.ru,
Филиал ФГАУ ГНИИ ИТТ "Информика" в г. Санкт-Петербурге, Санкт-Петербургский
политехнический университет Петра Великого

Панъевропейский научно-образовательный сетевой консорциум GÉANT: особенности инфраструктуры, ключевые проекты и сервисы¹

Работа посвящена телекоммуникационному консорциуму GÉANT, который представляет собой общеевропейскую высокоскоростную сеть передачи данных, эксплуатируемую и развиваемую в нуждах сферы науки и образования. Рассматриваются принципы функционирования, особенности опорной инфраструктуры и сетевой связности. Обсуждаются решаемые задачи по поддержке глобальных научно-исследовательских проектов и региональных телекоммуникационных инициатив, приводятся сведения о развиваемых сервисах, а также о направлениях взаимодействия сети RUNNet и GÉANT.

Ключевые слова: GÉANT, сетевой консорциум, национальная научно-образовательная сеть, NREN, RUNNet, телекоммуникационная инфраструктура, сетевая связность, сетевые сервисы

Введение

Термин "национальная научно-образовательная сеть" (National Research and Education Network, NREN) хорошо знаком в профессиональной среде специалистов, вовлеченных в масштабные научно-исследовательские проекты, в результате выполнения которых генерируются сверхбольшие объемы научных данных, требующих совместной обработки участниками в распределенном режиме с использованием развитых средств телекоммуникаций [1–3].

Под национальной научно-образовательной сетью принято понимать высокопроизводительную телекоммуникационную инфраструктуру масштаба страны, которая эксплуатируется в интересах науки и образования, обеспечивает доступ целевых пользователей в глобальное пространство информационно-коммуникационных технологий (ИКТ-пространство), связность с мировыми NREN и сетевыми консорциумами, а также является ядром развития и провайдером востребованных сетевых сервисов и сервисов коллективного пользования [1, 4–6].

В настоящее время собственные NREN функционируют более чем в 140 странах мира [4, 6–8]. В ряду наиболее развитых NREN можно упомянуть сети Internet2, ESnet и NISN (США), NORDUnet (страны Северной Европы), SURFnet (Нидерланды), DFN (Германия), JANET (Великобритания), CESNET (Чехия), CANARIE (Канада), AARNet (Австралия) и некоторые другие [4, 7, 8].

Федеральная университетская компьютерная сеть России RUNNet (Russian UNiversity Network, <http://www.runnet.ru>), существующая и развивающаяся с 1994 г., является сегодня крупнейшей и, по сути, единственной в нашей стране телекоммуникационной сетью научно-образовательной направленности [4–6, 9, 10]. Административно-технологическое и научно-методическое управление сетью RUNNet выполняет ФГАУ ГНИИ ИТТ "Информика".

RUNNet представляет собой опорную сеть национального уровня, имеет точки присутствия во всех восьми федеральных округах (в 56 субъектах страны), предоставляя свои телекоммуникационные и информационные услуги более 400 организациям высшего образования, научно-исследовательским организациям, региональным научно-образовательным

¹ Статья подготовлена в рамках Государственного задания ФГАУ ГНИИ ИТТ "Информика" на 2017–2018 гг.

сетям, ряду учреждений культуры и здравоохранения.

Сеть RUNNet обеспечивает российским научным и образовательным организациям надежный высокоскоростной доступ к международным NREN и расположенным в них ресурсам и сервисам, функционирование единого научно-образовательного информационного пространства сферы образования и науки страны для реализации информационного взаимодействия между организациями при осуществлении ими основной деятельности, оказании образовательных услуг, совместном выполнении наукоемких проектов [4, 5, 9].

Текущий этап развития NREN характеризуется повсеместным формированием межгосударственных сетевых объединений, организуемых в целях повышения степени связности национальных сетей друг с другом, а также технологической и экономической эффективности сетевого взаимодействия [4, 6—8]. Крупные сетевые консорциумы обычно связаны друг с другом высокопроизводительными каналами.

В ряду наиболее масштабных по охвату стран-участниц, статусу организационного и технологического развития научно-образовательных сетевых консорциумов особое место занимает панъевропейский проект GÉANT (Gigabit European Academic Network Technology, <http://www.geant.net>), объединяющий и связывающий между собой NREN практически всей Европы [4, 7, 8].

GÉANT нацелен на реализацию парадигмы "открытой науки" ("open science") посредством комплексной поддержки европейских и мировых исследовательских сообществ, стимулирования и предоставления сетевых сервисов и технологий для расширения возможностей сотрудничества в самых разных научных, научно-технологических и иных областях.

Отдельного упоминания здесь заслуживают исследовательские коллаборации и проекты, при реализации которых интенсивно эксплуатируется ИКТ-инфраструктура GÉANT и мировых NREN, в том числе в областях физики частиц, космических исследований и спутниковых наблюдений, медицины и здравоохранения, энергетики, электронного обучения, культуры и искусства.

Общеизвестным примером таких международных проектов являются исследования по физике высоких энергий, проводимые на базе Большого адронного коллайдера (Large Hadron Collider, LHC) в ЦЕРН. Хранение, передача и обработка десятков терабайтов данных, еже-

дневно генерируемых экспериментами на LHC, в настоящее время обеспечиваются в рамках проекта WLCG (Worldwide LHC Computing Grid), который представляет собой глобальную коллаборацию, объединяющую почти 200 вычислительных центров разных уровней со всего мира, в том числе расположенные в научных и образовательных организациях России [11, 12].

В статье освещаются принципы функционирования проекта GÉANT, особенности опорной инфраструктуры и сетевой связности, решаемые задачи по поддержке научно-исследовательских проектов и региональных телекоммуникационных инициатив, приводятся сведения о развиваемых технологиях, сетевых сервисах и сервисах коллективного пользования, обсуждаются направления взаимодействия сети RUNNet и GÉANT.

1. Консорциум GÉANT: опорная инфраструктура, сетевая связность, координация региональных телекоммуникационных проектов

1.1. Особенности опорной инфраструктуры сети GÉANT

Консорциум GÉANT представляет собой общеевропейскую ультравысокоскоростную мультигигабитную опорную транспортную IP-сеть ("backbone"), функционирующую в интересах сферы науки и образования и обеспечивающую сетевую связность более 10 тыс. образовательных и научных организаций из 39 стран Европы, Турции и Израиля (рис. 1, см. третью сторону обложки, <http://map.geant.org>) [7, 8, 13]. Общее число пользователей GÉANT превышает 50 млн человек.

Пропускная способность опорной инфраструктуры составляет 100 Гбит/с, объединяемые консорциумом NREN обладают как высокой производительностью, так и обширным географическим охватом, предоставляя исследователям, преподавателям и студентам по всей Европе широкие возможности для международного сотрудничества и тематического общения. GÉANT обладает собственными (или арендуемыми) телекоммуникационными узлами во всех странах-участницах проекта, на которых организуется обмен трафиком между локальными NREN.

Опорная инфраструктура GÉANT обладает 100 %-ной доступностью в среднем в течение месяца, ежедневно по ней передается более

4 Пбайт данных. Инфраструктура и сервисы сети обладают высокой степенью гибкости и могут быть адаптированы к индивидуальным потребностям пользователей [7, 13].

Необходимость обеспечивать поддержку широкого спектра пользователей и сервисов побудила GÉANT к применению двухуровневой сетевой модели, способной интегрировать различные требования к услугам в рамках единой базовой инфраструктуры, включающей в себя уровень передачи (transmission layer) и пакетный уровень (packet layer) [7].

Уровень передачи построен либо на базе темных волокон, либо на точках присутствия GÉANT (PoP), либо на арендованных у коммерческих провайдеров и/или NREN длинах волн в получивших широкое распространение DWDM-системах. GÉANT использует одно из ведущих в отрасли оптическое оборудование Infinera DTN-X, объединяющее сотни оптических компонентов на одном чипе с возможностью формирования суперканалов емкостью 500 Гбит/с, оптические усилители для управления волоконно-оптической магистралью, интегрированный OTN (Optical Transport Network) коммутационный уровень с технологией управления оптической сетью GMPLS (Generalized MultiProtocol Label Switching), позволяющей быстро обеспечить требуемую пропускную способность и восстановить емкость сети при разрыве волокон.

Пакетный уровень, поддерживая сервисы Layer2 и Layer3, является конвергентным: пользователям предоставляются как Ethernet-соединения (GÉANT Plus), так и IP-сервисы. Консорциум реализует данный уровень на базе оборудования Juniper MX (которое используется также и в сети RUNNet). Сервис GÉANT Plus предоставляется с использованием технологии MPLS, а IP-сервисы включают в себя: GÉANT IP, Layer3 VPN, GÉANT World Service, сервис обмена трафиком.

Консорциум совершает постоянные масштабные инвестиции в исследования и разработку сетевых архитектур, технологий и парадигм в целях перехода в будущем на новые ультрасовременные сервисы, инструменты и сетевые возможности [7, 13]

1.2. Региональная и межконтинентальная связность GÉANT. Поддержка региональных телекоммуникационных проектов

Текущая связность GÉANT с национальными научно-образовательными сетями и ре-

гиональными сетевыми проектами по всему миру обеспечивает обмен данными с более чем 60 странами за пределами Европы (рис. 2, см. третью сторону обложки).

GÉANT является постоянным партнером ЕС во многих программах и инициативах при осуществлении глобального сотрудничества в сфере образования и науки, координирует и управляет реализацией проектов по развитию межгосударственных сетевых консорциумов в Европе, Средиземноморском регионе, Африке и Центральной Азии. Кроме того, поддерживает научные и образовательные сетевые проекты в Латинской Америке, Карибском бассейне и Азиатско-Тихоокеанском регионе, координирует сотрудничество между Европой и Китаем, взаимодействует с коллегами из Северной Америки.

Консорциум, в ряду других, обладает IP-связностью с NREN Северной Америки посредством нескольких разнесенных каналов емкостью по 100 Гбит/с, Латинской Америки, Азиатско-Тихоокеанского региона, Африки (несколько каналов по 10 Гбит/с), Ближнего Востока (1...10 Гбит/с).

В рамках исторически налаженного взаимодействия общеевропейских и североамериканских исследовательских сетей функционирует высокоскоростная трансатлантическая связность между континентами. Многолетними партнерами GÉANT в США являются научно-образовательные сети Internet2, ESnet (Energy Sciences Network), NISN (NASA Integrated Services Network), а также Университет Индианы, в Канаде — сеть CANARIE.

В совместно эксплуатируемой сетевой инфраструктуре GÉANT отвечает за работу канала производительностью 100 Гбит/с на направлении Лондон — Нью-Йорк. Дополнительные каналы, предоставляемые североамериканскими партнерами GÉANT, гарантируют множественную связность, резервирование сетевой инфраструктуры, отсутствие единых точек сбоя, что обеспечивает беспрепятственную и непрерывную межконтинентальную передачу данных.

Указанная связность является ключевой в обеспечении функционирования научных коллабораций при совместном выполнении проектов и, в первую очередь, экспериментальных исследований на базе LHC в ЦЕРН, ряд центров обработки результатов которых находится в Северной Америке [7, 11, 12].

Латиноамериканский научно-образовательный консорциум RedCLARA (Cooperacin Latino Americana de Redes Avanzadas, <http://>

www.redclara.net) интегрирует 15 NREN региона и обладает связностью с GÉANT посредством трансатлантического канала между Сан-Паулу и Лондоном (емкостью 10 Гбит/с). Можно заметить, что в период с 2003 по 2012 г. консорциум RedCLARA частично финансировался ЕС в рамках проектов ALICE/ALICE2 (Latin America Interconnected with Europe).

На африканском континенте развиваются сразу несколько научно-образовательных сетевых объединений, среди которых в настоящее время в активной фазе выполнения находится общеконтинентальный проект AfricaConnect (<https://www.africconnect2.net>). Это объединение, инициированное в 2011 г., было задумано как проект panaфриканской "зонтичной" связности, который базируется на кластерном подходе, адаптированном к географическим, культурным и организационным особенностям регионов континента и различным уровням развития NREN.

В настоящее время реализуется второй этап этого проекта, который софинансируется ЕС (при координации GÉANT) и африканскими странами-участниками, связывает три географические области континента ("кластеры") и включает в себя региональные сетевые объединения и NREN, в том числе консорциумы стран северной (ASREN, Arab States Research and Education Network, <http://asrenorg.net>), восточной и южной (UbuntuNet, <http://www.ubuntunet.net>), а также западной и центральной Африки (WACREN, West and Central African Research and Education Network, <http://www.wacren.net>).

Проект EUMEDCONNECT (Eastern Mediterranean Connect, <http://www.eumedconnect3.net>) нацелен на создание и поддержку функционирования высокопроизводительной сети в интересах исследовательских и образовательных сообществ в регионе Восточного Средиземноморья в составе NREN Иордании, Ливана, Палестинской автономии и Сирии. В настоящее время при поддержке ЕС, под управлением GÉANT и в партнерстве с ASREN реализуется третий этап проекта (2016—2019 гг.).

В Азиатско-Тихоокеанском регионе ключевые консорциумы, объединяющие локальные NREN, — это проекты Asi@Connect и APAN (Asia-Pacific Advanced Network, <http://www.apan.net>). Asi@Connect (<http://www.tein.asia>) является преемником проекта TEIN (Trans-Eurasia Information Network), софинансируется ЕС и локальными NREN и предоставляет телекоммуникационную транспортную инфраструктуру научным и образовательным сетевым

коллаборациям в азиатском регионе, связывая исследователей друг с другом и с коллегами в Европе через прямые стыки с GÉANT. Консорциум Asi@Connect объединяет 20 NREN; пропускная способность опорной инфраструктуры сети составляет 10 Гбит/с.

Сетевая связность GÉANT с Китаем обеспечивается на базе совместно управляемого и финансируемого консорциумом GÉANT и NREN Китая канала между Пекином и Лондоном (емкостью 10 Гбит/с); такое сотрудничество является естественным продолжением проекта ORIENTplus (<https://www.orientplus.eu>), который реализовывался в 2011—2014 гг.

На протяжении текущей декады целевую финансовую поддержку развитию NREN в странах бывшего СССР (за исключением России) оказывает ЕС в рамках координируемых GÉANT международных проектов EaPConnect (Eastern Partnership Connect) и CAREN (Central Asian Research and Education Network).

Программа EaPConnect (<https://www.eapconnect.eu>) рассчитана на 2015—2020 гг. и ориентирована на страны так называемого "Восточного партнерства" (Армения, Азербайджан, Беларусь, Грузия, Молдова и Украина).

Программа CAREN (<https://caren.geant.org>) была инициирована в 2009 г.; в настоящее время (до 2019 г.) в стадии реализации находится третий этап проекта с участием Киргизии, Таджикистана, Казахстана, Туркмении и Узбекистана. Консорциум объединяет более 300 научных и образовательных организаций и около полумиллиона пользователей.

Цели обеих программ схожи — построение и последующая эксплуатация сети широкополосного доступа в Интернет с высокой пропускной способностью для сферы науки и образования стран-участниц; интеграция локальных NREN в общеевропейскую сеть GÉANT, в том числе в целях сокращения имеющегося "цифрового разрыва", внедрение и эксплуатация развитых сетевых сервисов и технологий; содействие участию преподавателей, исследователей и студентов стран в совместных научно-образовательных проектах.

Российская научно-образовательная сеть RUNNet имеет постоянную высокоскоростную связность с консорциумом GÉANT, реализованную на базе прямых пиринговых стыков с использованием портов производительностью 10 Гбит/с на одном из зарубежных опорных узлов RUNNet в Хельсинки.

На рис. 3 (см. третью сторону обложки) показан график, демонстрирующий типичные

объемы обмена данными пользователей RUNNet с мировым научно-образовательным сообществом (через инфраструктуру GÉANT) [4]. Приведенные показатели подтверждают высокую востребованность в кооперации российских и зарубежных научных и образовательных организаций в целях участия в совместных исследовательских проектах, требующих интенсивного использования средств телекоммуникаций.

3. Сетевые сервисы и сервисы коллективного пользования GÉANT

GÉANT поддерживает и развивает широкий спектр сетевых сервисов и сервисов коллективного пользования, включая передачу IP-трафика и организацию выделенных сетей, виртуальные и облачные ресурсы и сервисы, сервисы федеративной аутентификации и роуминга в Wi-Fi сетях, службы коммуникаций в режиме реального времени, средства эффективного мониторинга и устранения неполадок при работе сети и т. д.

Сервисы, разрабатываемые и эксплуатируемые под управлением GÉANT в интересах сферы науки и образования, можно условно разделить на следующие функциональные категории [7, 13]:

- сервисы обеспечения сетевой связности и управления сетью (*Connectivity & network management*);
- сервисы доверия, идентификации и безопасности (*Trust, Identity & Security*);
- облачные сервисы (*Cloud services*);
- сервисы коммуникаций в режиме реального времени (*Real-time communications*).

Портфель сервисов обеспечения сетевой связности в нотации GÉANT включает в себя: транзит научного IP-трафика, организацию выделенных сетей с повышенными требованиями к качеству, VPN-сервисы, поддержку открытых исследовательских коллабораций (GÉANT Open), создание тестовых телекоммуникационных стендов для исследователей (GÉANT Testbeds).

Развиваемые GÉANT сервисы управления сетью для научных и образовательных организаций — это:

- *eduPERT* (EDUcation Performance Enhancement Response Team, <https://edupert.geant.net>), исследовательская и консультационная служба по вопросам производительности сети;
- *perfSONAR* (Performance focused Service Oriented Network monitoring ARchitecture, [https://](https://www.perfsonar.net)

www.perfsonar.net), сервис мониторинга производительности сети в режиме реального времени.

Основные сервисы доверия, идентификации и безопасности:

- *eduGAIN* (EDUcation Global Authentication Infrastructure, <http://edugain.org>), глобальная инфраструктура федеративной авторизации;
- *eduroam* (EDUcation ROAMing, <https://www.eduroam.org>), сервис международного роуминга в Wi-Fi сетях;
- *eduPKI* (EDUcation Public Key Infrastructure, <https://www.edupki.org>), инфраструктура открытых ключей шифрования;
- *TACAR* (Trusted Academic CA Repository, <https://www.tacar.org>), сервис хранения и безопасного распространения цифровых сертификатов CA.

Сервис eduGAIN объединяет системы федеративной авторизации, эксплуатируемые в разных странах по всему миру, предоставляя пользователям уникальные возможности доступа к контенту, сервисам и ресурсам глобального сообщества сферы образования и науки по технологии "единого входа" (SSO, Single Sign-On) [7, 13].

В сети RUNNet с 2017 г. ведутся работы по построению собственной платформы федеративной авторизации, в рамках которых, в частности, создана инфраструктура управления федерацией, разработаны технические инструкции по развертыванию сервисов в конечных организациях, создан демонстрационный полигон федеративного доступа к облачным сервисам коллективного пользования, утверждены организационно-правовые и технические регламенты удостоверяющей федерации, подготовлены необходимые документы и подана заявка на вступление удостоверяющей федерации России (RUNNetAAI) в международный проект eduGAIN [4, 14].

Проект eduroam представляет собой сервис роуминга в беспроводных Wi-Fi сетях для научно-образовательного сообщества, при использовании которого студенты, преподаватели и научные работники получают возможность бесплатного доступа к сети Интернет и ресурсам NREN в вузах и научных центрах страны и за рубежом — в кампусах организаций—участников проекта [4, 7, 14]. При этом доступ к сервису с личных портативных и мобильных устройств осуществляется по единым учетным данным (логину и паролю), выданным пользователю его "домашней" организацией.

На текущий момент в части развития данного сервиса в RUNNet создана интегрирован-

ная инфраструктура авторизации и аутентификации пользователей на базе сетей RUNNet и RASNet (корпоративная сеть РАН), организовано резервирование ключевых элементов программно-аппаратной инфраструктуры сервиса, разработаны организационно-технические решения; стартовали работы по подключению к проекту вузов и научных институтов, развернута пилотная зона роуминга в Wi-Fi сетях [4, 14].

Ключевые группы развиваемых GÉANT *облачных сервисов* — это Infrastructure as a Service (IaaS), сервисы хранения файлов и синхронизации данных, сервисы коммуникаций в режиме реального времени, дистанционного и электронного обучения, обеспечения информационной безопасности.

Среди облачных сервисов к наиболее востребованным можно отнести:

- *ownCloud* (<https://owncloud.org>) — сервис для синхронизации данных, обмена файлами и удаленного хранения с развитым дополнительным функционалом;
- *FileSender* (<http://www.filesender.org>) — сервис трансфера и обмена файлами большого размера.

Следует заметить, что оба эти сервиса развернуты и активно эксплуатируются в собственной облачной инфраструктуре RUNNet в пределах полигона сервисов коллективного пользования на базе технологии SSO [4, 14]. Платформа ownCloud, в числе других, предоставляет возможности совместной работы с документами офисных форматов, реализованные на основе API, с помощью которого осуществляется интеграция ownCloud с открытым офисным пакетом ONLYOFFICE Document Server.

Облачная инфраструктура RUNNet функционирует в виде приватного облака и базируется на открытой облачной платформе OpenStack [4, 15]. В облаке работают несколько десятков виртуальных машин, ориентированных в том числе на поддержку работоспособности сервисов сети RUNNet, таких как DNS (хранение зон, резолвинг), хостинг образовательных веб-сайтов, проверка электронной почты на спам и вирусы, поддержка служб VoIP-телефонии, блокировка доступа к запрещенным российским законодательством веб-ресурсам, сервисы видеоконференций и вебинаров, сервисы сбора и обработки сетевой статистики, анализа сетевого трафика пользователей, службы мониторинга работы сети и сервисов и др.

Многопользовательские сервисы коммуникаций режима реального времени, развиваемые GÉANT:

- *eduCONF* (EDUcation CONFerencing, <https://educonf.geant.org>) — сервис проверки видеоконференций NREN;
- *eduOER* (EDUcation Open Educational Resource, <https://oer.geant.org>) — сервис поддержки доступа к мультимедийному контенту разных репозиториям;
- *NRENum.net* (<https://nrenum.net>) — сервис сопровождения телефонных номеров международного телекоммуникационного плана E.164.

Администрация и технические специалисты RUNNet заинтересованы и предпринимают конкретные организационные и практические шаги по дальнейшей апробации и внедрению в сети ряда перечисленных выше сервисов, в том числе в целях более глубокой интеграции RUNNet в мировое научно-образовательное телекоммуникационное пространство и расширения областей взаимодействия с консорциумом GÉANT и зарубежными NREN.

Заключение

В работе рассмотрены и обсуждены решаемые задачи, принципы функционирования и основные направления деятельности сетевого консорциума GÉANT как глобального ИКТ-проекта, интегрирующего и связывающего между собой национальные научно-образовательные сети стран Европы, взаимодействующего с NREN стран других континентов и координирующего развитие региональных сетевых консорциумов. Представлены особенности опорной инфраструктуры GÉANT и организации сетевой связности, сведения о развиваемых сервисах, текущих и потенциальных направлениях взаимодействия Федеральной университетской сети России RUNNet и консорциума GÉANT.

Список литературы

1. Allocchio C., Balint L., Berkhout V., Bersee J., Izhvanov Y. et al. A History of international research networking: the people who made it happen. N. Y.: Wiley-VCH, 2010. 317 p.
2. Ryan J. A History of the Internet and the digital future. London: Breaction Books Ltd, 2015. 248 p.
3. Lehtisalo K. The History of NORDUnet: Twenty-five years of networking cooperation in the Nordic countries (2005). URL: <http://www.nordu.net/history/book.html>.
4. Отчет о выполнении Государственного задания ФГАУ ГНИИ ИТТ "Информика" на 2017 г. по теме "Обеспечение функционирования инфраструктуры RUNNet и доступа Минобрнауки России к сети Интернет. Проектирование и подготовка к запуску научных сервисов для обеспечения до-

ступа пользователей к данным и результатам исследований в приоритетных научных областях". М.: ФГАУ ГНИИ ИТТ "Информика", 2017. 514 с.

5. **Концепция** функционирования и развития федеральной университетской компьютерной сети RUNNet на 2017–2019 годы (утверждена Минобрнауки России 03 октября 2017 г. № ГТ-90/вн). М.: ФГАУ ГНИИ ИТТ "Информика", 2017. 33 с.

6. **Ижванов Ю. Л.** Научно-образовательные компьютерные сети. Прошлое, настоящее и тенденции развития // Образовательные ресурсы и технологии. 2017. № 2 (19). С. 17–25.

7. **Официальный** сайт консорциума GÉANT. URL: <http://www.geant.net>.

8. **GÉANT Association Compendium of national research and education networking organisations in Europe — 2015 Edition.** URL: <https://compendium.geant.org>.

9. **Абрамов А. Г., Евсеев А. В.** Сеть RUNNet: навстречу современным вызовам сферы телекоммуникаций в науке и образовании // Информатизация образования и науки. 2017. № 1 (33). С. 100–115.

10. **Абрамов А. Г., Евсеев А. В.** Анализ потребностей организаций высшего образования России в телекоммуникационных сервисах научно-образовательной направленности // Информатизация образования и науки. 2018. № 2 (38). С. 9–18.

11. **Официальный** сайт проекта WLCG. URL: <http://wlcg.web.cern.ch>.

12. **Martelli E., Stancu S.** LHCOPN and LHCONE: status and future evolution // Journal of Physics: Conference Series. 2015. Vol. 664. URL: <http://iopscience.iop.org/article/10.1088/1742-6596/664/5/052025>.

13. **GÉANT Annual Report 2016.** URL: <https://ar2016.geant.org>.

14. **Абрамов А. Г., Васильев И. В., Порхачев В. А.** Развитие инфраструктуры аутентификации и авторизации для удостоверяющей федерации в рамках проектов eduGAIN и eduoam на базе сети RUNNet // ИТНОУ: Информационные технологии в науке, образовании и управлении. 2017. № 4. С. 56–64.

15. **Абрамов А. Г.** Облачные технологии на основе открытых программных решений в университетской сети RUNNet: опыт развертывания и эксплуатации // Информатизация образования и науки. 2016. № 1 (29). С. 106–117.

A. G. Abramov, e-mail: abramov@runnet.ru,

St.-Petersburg branch of FPAI SIIT&T "Informika" (Deputy director),
Peter the Great St.-Petersburg Polytechnic University (Assistant professor)

Pan-European Research and Education Network Consortium GÉANT: Features of Infrastructure, Key Projects and Services

This review paper is dedicated to the world's largest telecommunication consortium GÉANT (Gigabit European Academic Network Technology), which is the pan-European high-speed data transfer network operated and developed in the needs of the research and education sphere. General information about current status of National Research and Education Networks (NRENs) developing around the world and Federal university computer network of Russia RUNNet (Russian UNiversity Network) is given. The main principles of functioning, the features of the terabit-ready backbone infrastructure, global, regional and intercontinental network connectivity of the GÉANT are considered. The innovative GÉANT network technology which is built on two layers (transmission and packet) is presented and discussed in brief. Some quantitative and qualitative characteristics of the network, GÉANT topology and connectivity maps are given. The main tasks are being performed by consortium in order to support and manage global research projects in priority scientific areas and regional telecommunications initiatives (such as RedCLARA, AfricaConnect, Asi@Connect, EaPConnect, CAREN) are mentioned and discussed. The information about the network and collaboration services developed by consortium in cooperation with local NRENs is provided. Among these services are connectivity and network management services, trust, identity and security services, the wide range of cloud services, real-time communications services, etc. The current state and potentially directions in-demand of the cooperation between the RUNNet network and the consortium GÉANT are considered.

Keywords: GÉANT, network consortium, national research and education network, NREN, RUNNet, telecommunication infrastructure, network connectivity, network services, research communities

DOI: 10.17587/it.24.546-552

References

1. **Allocchio C., Balint L., Berkhout V., Bersee J., Izhvanov Y. et al.** *A History of international research networking: the people who made it happen*, N. Y., Wiley-VCH, 2010, 317 p.

2. **Ryan J.** *A History of the Internet and the digital future*, London, Breaktion Books Ltd., 2015, 248 p.

3. **Lehtisalo K.** *The History of NORDUnet: Twenty-five years of networking cooperation in the Nordic countries*, 2005, available at: <http://www.nordu.net/history/book.html>.

4. **Otchet** o vypolnenii Gosudarstvennogo zadaniya FGAU GNII ITT "Informika" na 2017 g. po teme "Obespechenie

funktsionirovaniya infrastruktury RUNNet i dostupa Minobrnauki Rossii k seti Internet. Proektirovanie i podgotovka k zapusku nauchnykh servisov dlya obespecheniya dostupa pol'zovateley k dannym i rezul'tatam issledovaniy v prioritnykh nauchnykh oblastiakh", Moscow, FGAU GNII ITT "Informika", 2017, 514 p. (in Russian).

5. **Kontseptsiya** funktsionirovaniya i razvitiya federal'noy universitetskoj komp'yuternoy seti RUNNet na 2017–2019 gody (utverzhdena Minobrnauki Rossii 03 oktyabrya 2017 g. № GT-90/vn), Moscow, FGAU GNII ITT "Informika", 2017, 33 p. (in Russian).

6. **Izhvanov Yu. L.** Nauchno-obrazovatel'nye komp'yuternye seti. Proshloe, nastoyashchee i tendentsii razvitiya (Research and

education computer networks. Past, present and development trends), *Obrazovatel'nye resursy i tekhnologii*, 2017, vol. 2 (19), pp. 17–25 (in Russian).

7. **Official** website of the consortium GÉANT, available at: <http://www.geant.net>.

8. GÉANT Association Compendium of national research and education networking organisations in Europe — 2015 Edition, available at: <https://compendium.geant.org>.

9. **Abramov A. G., Evseev A. V.** Set' RUNNet: navstrechu sovremennym vyzovam sfery telekommunikatsiy v nauke i obrazovanii (Network RUNNet: towards the state-of-the-art challenges in the field of telecommunications in science and education), *Informatizatsiya Obrazovaniya i Nauki*, 2017, vol. 1 (33), pp. 100–115 (in Russian).

10. **Abramov A. G., Evseev A. V.** Analiz potrebnostej organizacij vysshego obrazovaniya Rossii v telekommunikatsionnykh servisakh nauchno-obrazovatel'noj napravlenosti (Analysis of demands of russian organizations of higher education in telecommunication services of scientific and educational direction), *Informatizatsiya Obrazovaniya i Nauki*, 2018, vol. 2 (38), pp. 9–18 (in Russian).

11. **Official** website of the project WLCG, available at: <http://wlcg.web.cern.ch>.

12. **Martelli E., Stancu S.** LHCOPN and LHCONE: status and future evolution, *Journal of Physics: Conference Series*, 2015, vol. 664, available at: <http://iopscience.iop.org/article/10.1088/1742-6596/664/5/052025>.

13. GÉANT Annual Report 2016, available at: <https://ar2016.geant.org>.

14. **Abramov A. G., Vasilyev I. V., Porhachev V. A.** Razvitie infrastruktury autentifikatsii i avtorizatsii dlya udostoverayushchey federatsii v ramkakh proektov eduGAIN i eduroam na baze seti RUNNet (Development of the authentication and authorization infrastructure for the identity federation within the eduGAIN and eduroam projects based on the RUNNet network), *ITNOU: Informatsionnye Tekhnologii v Nauke, Obrazovanii i Upravlenii*, 2017, vol. 4, pp. 56–64 (in Russian).

15. **Abramov A. G.** Oblachnye tekhnologii na osnove otkrytykh programmnykh reshenij v universitetskoj seti RUNNet: opyt razvertyvaniya i jekspluatatsii (Cloud technologies based on open source solutions in the university network RUNNet: experience of deployment and operation), *Informatizatsiya Obrazovaniya i Nauki*, 2016, vol. 1 (29), pp. 106–117 (in Russian).

УДК 004.422

DOI: 10.17587/it.24.553-560

А. А. Романов, преподаватель, e-mail: gloomsfb@gmail.com,

Д. Г. Волчек, преподаватель, e-mail: dvolchek@yandex.ru,

Д. И. Муромцев, канд. техн. наук, доц., зав. каф., e-mail: mouromtsev@corp.ifmo.ru,
Университет ИТМО, Санкт-Петербург

Семантические технологии в массовых открытых онлайн-курсах: экстракция, обогащение и связывание данных электронных учебных курсов для образовательной платформы Open edX

Рассматривается проблема, связанная с увеличением количества учебных материалов, которые доступны в открытых онлайн-курсах. Такой рост приводит к снижению качества отдельных ресурсов, усложнению классификации и поиска, что в целом негативно отражается на конечных пользователях. Предлагается решение для платформы Open edX, основанное на принципах семантических технологий и объединяющее в себе экстракцию, обогащение и взаимосвязь учебных данных. Для исследования эффективности метода и внедрения его в учебный процесс были разработаны и протестированы специальные испытательные компоненты.

Ключевые слова: семантическая сеть, образование, связанные данные в образовании, образовательная онтология, электронное образование, семантические технологии в образовании

Введение

Идея дистанционного образования рассматривается уже достаточно давно [1, 2], и по мере развития информационных технологий эта идея воплощается в жизнь все более успешно. Спектр различных форм, используемых для реализации концепции дистанционного обучения, широк и включает в себя как достаточно примитивные способы (напри-

мер, бумажную переписку, радиотрансляции, телепрограммы), так и более успешные виды (такие как онлайн-обучение), которые обеспечивают быструю и удобную коммуникацию между субъектами образовательного процесса. Онлайн-обучение стало возможным благодаря развитию сети Интернет, росту ее популярности и, соответственно, увеличению числа пользователей. Начиная с середины 90-х годов XX века многие университеты мира продви-

гали свои образовательные программы в сети Интернет. В основной своей массе это были частные инициативы самих учебных заведений, которые университеты реализовывали собственными силами. При этом наблюдалось многообразие организации такого вида обучения, что привело к ряду позитивных и негативных последствий. К позитивным факторам можно отнести тот факт, что благодаря независимости университетов на разработчиков не накладывались какие бы то ни было ограничения. При таком подходе они могли изобретать и исследовать различные методы, средства, формы и стратегии. Однако обучающемуся было неудобно, а порой и невозможно, участвовать в образовательном процессе из-за различий вузовских систем онлайн-образования по своей структуре и форме, а также из-за направленности онлайн-курсов вуза на обучение студентов только своего вуза.

Следующим этапом эволюции дистанционных образовательных технологий стали массовые открытые онлайн-курсы (МООК). Эта концепция была представлена в 2008 г. и через четыре года превратилась в по-настоящему влиятельный тренд [3]. В данный момент существуют несколько крупных платформ, которые на своих ресурсах размещают курсы различных учебных заведений. Одна из них — проект edX.org, в основе которого лежит свободно распространяемая система управления обучением (Learning Management System, LMS) Open edX [4]. В проекте edX.org участвуют около 100 ведущих мировых университетов, которые предоставляют свыше 1300 курсов. Обучение проходят более 10 миллионов студентов со всего мира. Сама платформа Open edX используется многими другими образовательными проектами, например, проектом "Открытое образование" (<http://openedu.ru>) или площадкой курсов Университета ИТМО (<http://courses.ifmo.ru>). С одной стороны, совокупность всех образовательных курсов представляет собой большой объем данных, которые уже невозможно обрабатывать традиционными способами, причем семантические технологии решают эту проблему, определяя общие правила для осуществления обмена данными и их многократного использования в различных приложениях, корпорациях и сообществах [5]. С другой стороны, это эффективный способ представления данных в сети Интернет, причем такой формат понятен как человеку, так и компьютеру. Семантические технологии представляют значения с помощью онтологии и обеспечивают

аргументацию, используя связи, правила, логику и условия, оговоренные в онтологии [6]. Семантические технологии помогают извлекать полезную информацию из данных, содержания документов или кодов приложений, опираясь на открытые стандарты, что позволяет говорить о том, что компьютер понимает смысл документа. Процесс построения онтологий получил название "онтологический инжиниринг" [7], и его можно считать развитием методологии инженерии знаний с помощью искусственного интеллекта.

В данной статье описан онтологический инжиниринг и основные этапы разработки по внедрению семантических технологий в МООК и их необходимость. Стоит отметить факт научной новизны, а именно — отсутствие схожих по реализации проектов в сфере образования, хотя семантические технологии широко применяются в сфере бизнеса и банковского сектора.

Постановка проблемы и ее решение

Рост числа образовательных материалов [8], безусловно, является позитивным аспектом индустрии образования. Используя существующие ресурсы, можно удобно и быстро получить высококачественные, а иногда и уникальные знания во многих предметных областях. Однако такой объем информации приносит и определенные проблемы, такие как:

- необходимость структурирования информации;
- сложность поиска необходимых знаний;
- возникновение дублирования и наложения учебных материалов;
- необходимость построения междисциплинарных связей;
- сложность построения образовательных траекторий;
- соответствие информации учебным планам и связь с компетенциями.

Таким образом, онтологический инжиниринг и разработка ряда модулей для образовательной платформы Open edX нацелена решить указанные проблемы.

МООК состоят из нескольких базовых частей, а именно, из теоретических материалов, практических заданий, оценки знаний и финального экзамена. В различных курсах возможны вариации (взаимное оценивание, проектная работа, творческое задание и т. п.), но их число невелико. Если автоматизировать

извлечение из теоретических материалов ключевых слов или так называемых концептов предметной области, то можно рассматривать их как "единицы знаний" [9, 10]. По данному набору концептов можно определить, к какой предметной области относится теоретический материал, что он описывает, какие цели преследует. Если обогатить концептами достаточный объем курсов, станет возможным построение иерархии предметных областей и, как следствие, междисциплинарных связей. Таким образом достигается структурирование информации. Если обучающийся встречается в лекции незнакомые термины и понятия, то, используя список концептов, он может посмотреть, в каких курсах и каких лекциях они также используются и объясняются. Все это позволяет повторно использовать теоретические материалы, облегчает поиск необходимых знаний, обеспечивает построение междисциплинарных связей.

Кроме теоретических материалов концептами можно обогащать и контент, так как он служит источником знаний, которые должен усвоить студент. Анализируя, какие предметные области изучены хорошо, а с какими обучающемуся еще нужно поработать, можно составить рекомендации для конкретного студента. Таким образом достигается построение индивидуальных образовательных траекторий. Оценивая степень того, насколько концепты теоретических материалов покрывают концепты тестов, можно оценить их соответствие друг другу, а значит, в случае несоответствия, сообщить создателям курса о необходимости корректировки. Таким образом, извлечение концептов поможет решить следующий ряд проблем.

1. *Построение областей знаний.* Путем анализа состава и порядка появления концептов, используемых в курсе, возможно семантическое построение предметных областей [11].

2. *Навигация по курсам.* Анализируя связи между концептами и пересечения различных предметных областей, можно осуществлять нелинейную навигацию, в том числе с учетом других курсов, доступных на платформе Open edX.

3. *Связывание курсов.* Если обучающемуся встретился незнакомый концепт, можно узнать, к какой предметной области относится концепт, где еще он используется или описывается, какие другие концепты с ним связаны, какие концепты ответственные за его появление или, наоборот, к появлению каких концептов он приводит.

4. *Индивидуализация.* Концепты, представленные в семантическом виде и формирующие предметные области, позволяют учащемуся

иметь в своем распоряжении не один конкретный курс, а все курсы, связанные между собой. Обучающийся может удобно и просто "путешествовать" по различным разделам многочисленных курсов, получая информацию в той последовательности и том объеме, которые удобны конкретно ему.

5. *Рекомендации.* Концепты можно получать не только из образовательного контента, но и из тестов. Таким образом, можно предоставлять рекомендации учащимся о том, что предметная область изучена недостаточно хорошо и требуется повторить материал. Рекомендации могут быть даны и создателям курсов, если, например, тест и образовательные материалы недостаточно коррелируют между собой с точки зрения использования концептов.

Модель онтологии MOOK

Представление данных MOOK в формальном, семантическом виде в целях их адекватной трактовки человеком и надлежащей машинной обработки — это достаточно сложная и распространенная задача [12]. Данная проблема может быть решена, в том числе благодаря онтологическому моделированию и семантическим технологиям [13]. Разработанная авторами онтология онлайн-курсов состоит из 25 классов, 13 свойств-объектов и 19 свойств-значений. Разработанная онтология (рис. 1) основана на онтологиях высшего уровня:

- AIISO (<http://purl.org/vocab/aiiso/schema>), которая предоставляет классы и свойства-объекты для описания внешней организационной структуры учреждения, занимающегося образовательной деятельностью;
- BIBO (<http://purl.org/ontology/bibo/>), которая предоставляет основные концепты и свойства для описания цитат и библиографических ссылок;
- FOAF (<http://xmlns.com/foaf/spec/>), которая описывает участников какого-либо процесса, виды их деятельности и виды отношений между собой или другими объектами;
- TEACH (<http://linkedscience.org/teach/ns/teach.rdf>), которая является словарем, содержащим термины, относящиеся к образовательному процессу.

Онтология состоит из классов, описывающих структуру онлайн-курса, и классов, которые содержат описание концептов и предметных областей. Определения этих классов приведены ниже.

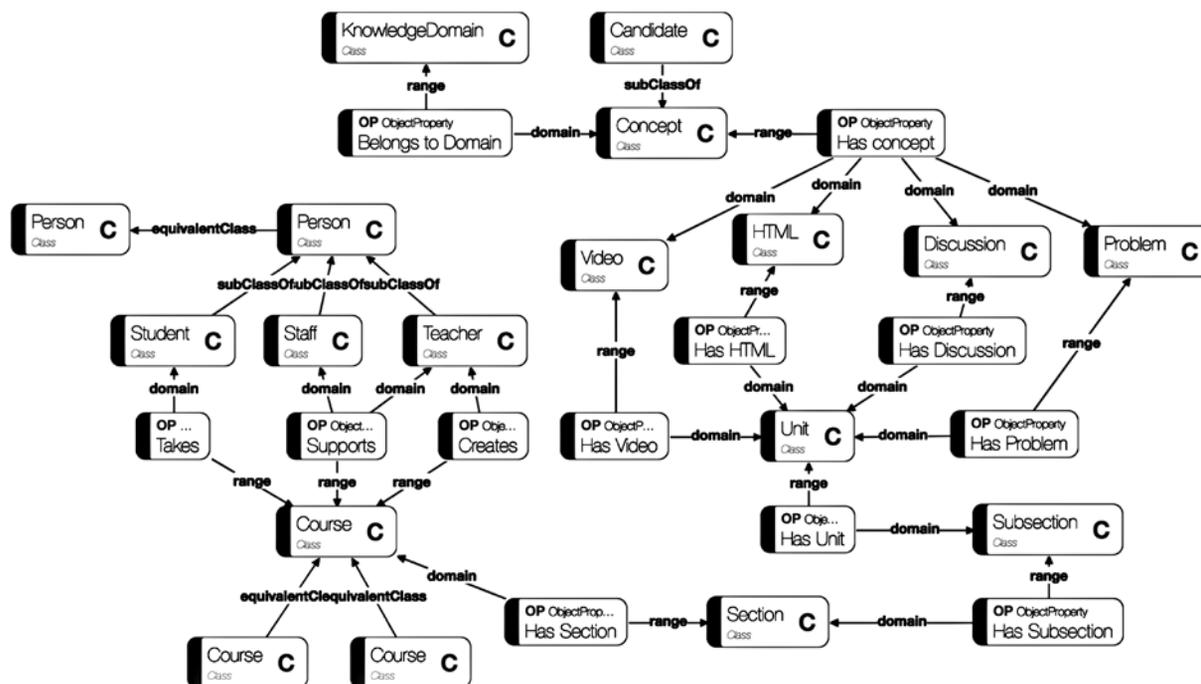


Рис. 1. Схема онтологии курсов платформы Open edX

- **Course (Курс)** — главный класс рассматриваемой онтологии. Он соответствует онтологиям AIIISO:Course и TEACH:Course. Атрибуты указанного класса включают в себя: даты начала и окончания курса; число часов в неделю, необходимое для успешного прохождения курса; название, описание и изображение для обозначения курса.
- **Section (Секция)** — класс, описывающий верхний уровень структуры курса. Он содержит следующие атрибуты: заголовок, дату начала и видимость пользователям (секции — это учебные разделы, отображающиеся поэтапно, обычно еженедельно).
- **Subsection (Подсекция)** — класс, описывающий элементы, входящие в секции. Данный класс включает следующие атрибуты: заголовок, дату начала, видимость и сроки выполнения (например, дата выполнения тестового задания, однако сроки могут и отсутствовать).
- **Unit (Юнит)** — класс, описывающий элементы, входящие в подсекции. В него входят следующие атрибуты: заголовок и видимость.
- **HTML** — класс, содержащий основной образовательный материал (текстовые лекции) в формате HTML.
- **Video (Видео)** — класс, необходимый для описания учебных материалов в формате видео и содержащий следующие атрибуты: ссылку на видео, хостинг (ресурс), субтитры.
- **Problem (Тест)** — класс для различных форм тестовых заданий. Атрибуты: число баллов, число попыток и тип заданий.
- **Person (Пользователь)** — класс для описания информации обо всех пользователях, вовлеченных в образовательный процесс MOOK платформы. Он соответствует онтологии FOAF:Person и содержит следующие подклассы:
 - **Teacher (Преподаватель)** — класс, описывающий создателей и разработчиков курса;
 - **Staff (Персонал)** — класс, описывающий участников, вовлеченных в поддержку курса (отвечающих на вопросы и предложения и решающих проблемы пользователей);
 - **Student (Студент)** — класс, описывающий участников, изучающих курс.
- **Concept (Концепт)** — класс, описывающий термины, входящие в учебные материалы:
 - **Concept Candidate** (Кандидат в концепты) — класс для хранения информации об автоматически полученных в результате обработки естественного языка концептах. Класс включает следующие атрибуты: вес концепта (его значимость), число вхождений в различные разделы и рейтинг;
 - **Concept** (Концепт) — класс, содержащий описание концептов и их связи с внешними источниками данных (такие как DBpedia, Wikidata).

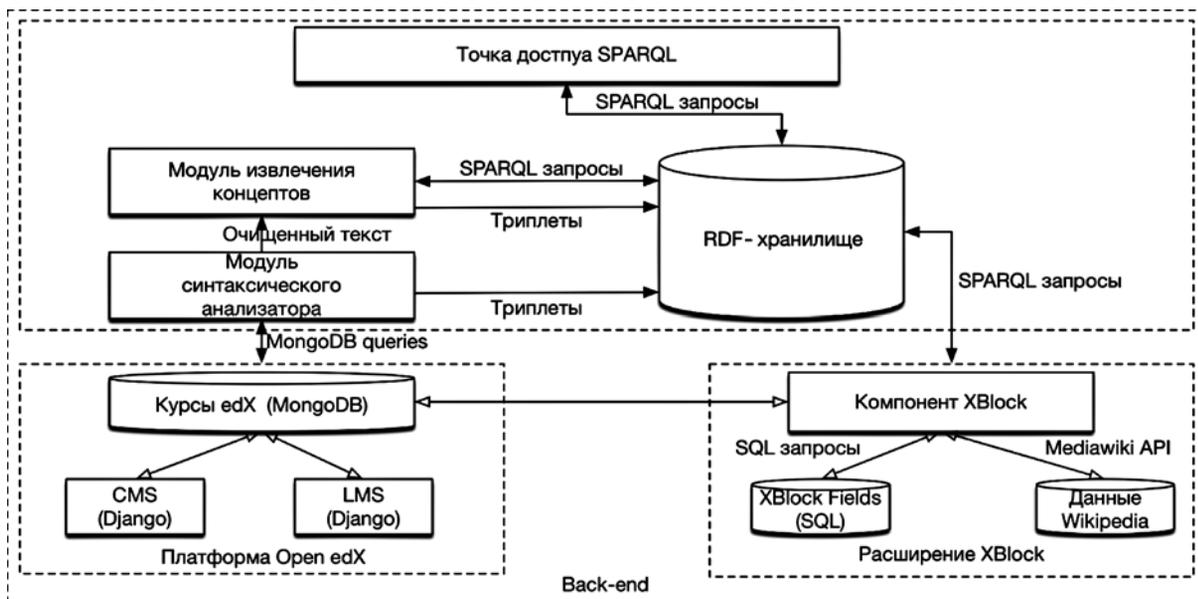


Рис. 2. Общая архитектура разработки

Программная реализация

Как показано на общей схеме архитектуры (рис. 2), компонент XBlock взаимодействует с хранилищем платформы Open edX посредством MongoDB-запросов. Таким образом, с помощью разработанного расширения происходит обработка данных, связанных со структурой курсов, лекциями, учебными материалами в формате HTML, концептами, полученными за счет алгоритма экстракции ключевых слов, и концептами, связанными с лекциями вручную. Расширение XBlock, разработанное с использованием языка Python, преобразует данные в триплеты (формат N-Triples <http://www.w3.org/TR/n-triples/>) и экспортирует их в семантическое ядро системы на основе RDF4J (<http://rdf4j.org/>).

Компонент XBlock. XBlock является стандартной точкой расширения платформы Open edX со своим API. Для хранения данных, по-

лученных от пользователя, используются формы, которые по умолчанию сохраняют информацию в предустановленной базе данных SQL. Указанная БД была расширена в связи с необходимостью динамического поиска данных по уже существующим концептам, "привязанным" к другим лекциям. Кроме того, динамический поиск связан с терминами из Википедии посредством MediaWiki API. Компонент XBlock (рис. 3) решает задачу связывания лекций с концептами, чтобы затем установить соответствие данных с онтологией.

Модуль синтаксического анализатора. На рис. 4 представлена структура модуля синтаксического анализатора базы данных MongoDB Open edX. После обработки анализатором данные преобразуются в зависимости от одного из нижеуказанных типов.

- *Структура.* В отсутствие документации о структуре хранилища и с учетом того факта, что используется документно-ориентированная СУБД MongoDB, основной задачей было выявить все связи между учебными материалами курсов, распределенные по трем следующим коллекциям.

1. В коллекции `Active_versions` хранится краткая информация и текущая опубликованная версия (`published-branch`) курса, причем версия имеет идентификатор `ObjectId`, используемый для связи со второй коллекцией.

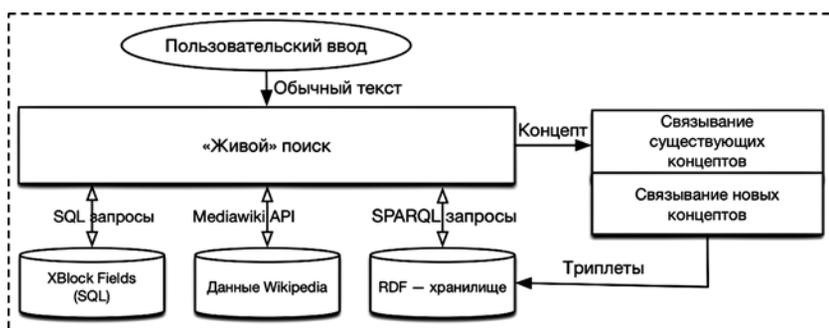


Рис. 3. Структура компонента XBlock

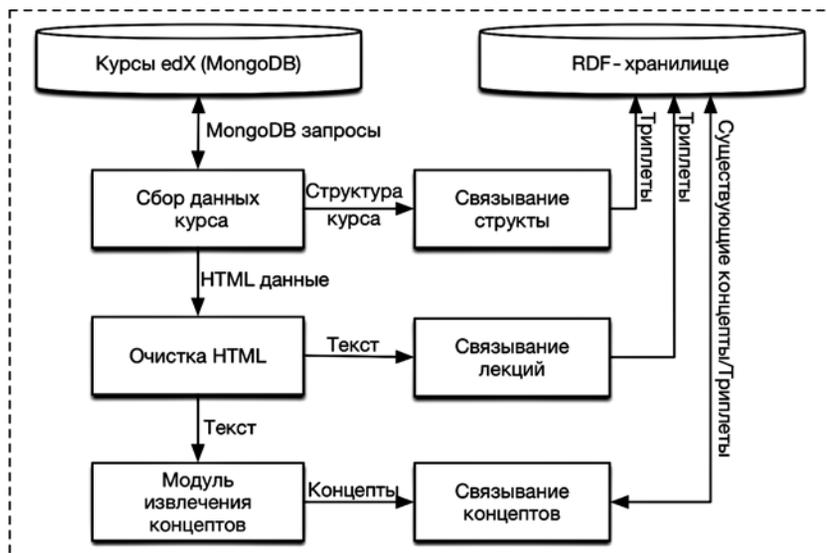


Рис. 4. Структура синтаксического анализатора

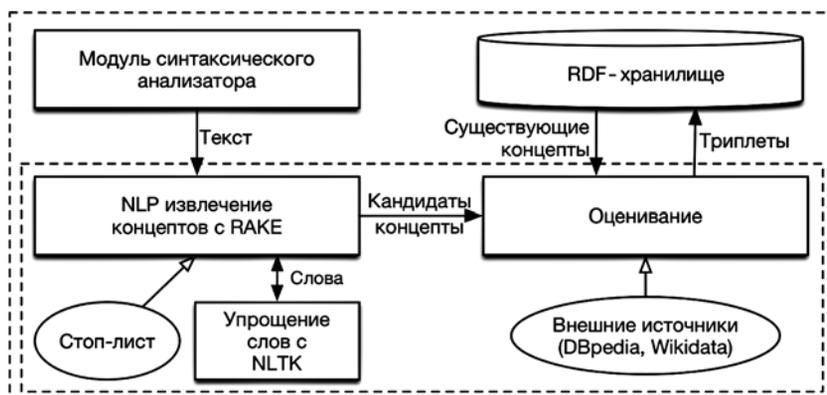


Рис. 5. Структура модуля извлечения концептов

2. В коллекции Structures находится вся история изменений и структура всех курсов, при этом один набор данных содержит сразу все данные о курсе. Следовательно, определить, к чему относится информация (лекция, видео или тест), возможно лишь по параметру типа данных (*block_type*). Кроме того, необходимо учитывать иерархию курса в системе Open edX, т. е. вложенность всех учебных материалов (называемых юнитами) и полный путь от родителей до потомков. Идентификатор потомков позволяет выявить связь с третьей коллекцией.

3. В Definitions хранятся все данные курса, такие как текст лекций, учебные тесты, обсуждения, видео и их субтитры и т. д.

- *HTML-данные.* Все данные лекций хранятся в формате HTML. Для создания читаемого текста модуль определяет все HTML-теги, прочие нечитаемые последовательности и запрещенные символы, чтобы установить

корректные связи с онтологией и затем выполнить импорт в RDF-хранилище.

- *Концепты.* Концепты разделяются на два типа. К первому типу относятся концепты, добавленные "вручную" и связанные с лекциями с помощью XBlock компонента, а ко второму — концепты, автоматически выявленные из лекционного материала посредством модуля извлечения концептов (рис. 5).

Модуль извлечения концептов. Связывание лекций и концептов "вручную" создателем курса — длительный процесс, в особенности для уже полностью готового курса. В связи с этим был применен метод автоматического извлечения концептов на основе нейролингвистического программирования. В основу соответствующего модуля были положены готовые способы извлечения ключевых слов (RAKE — Rapid Automatic Keyword Extraction, инструментарий Быстрого Автоматического Извлечения Ключевых Слов) и NLTK (Natural Language Toolkit, Инструментарий Естественного Языка), а также дополнительные преобразования, связанные с более точным выделением ключевых элементов текста и его обработкой, в частности, с определением падежей, рода, числа и т. д. В результате были получены кандидаты в концепты, оцененные по открытым источникам данных (DBpedia, Wikidata), по уже связанным ранее концептам, и по добавленным "вручную" авторами курсов.

Результаты внедрения компонента

Разработанные модули и компонент XBlock были успешно интегрированы в платформу Open edX и протестированы на уже размещенных онлайн-курсах. В результате работы ошибок не выявлено, а экстракция всех данных и выявление концептов прошли успешно, что позволило получить следующие количественные метрики: курсов — 6, секций — 64, подсекций — 315, юнитов — 941, страниц лекций — 1005; время работы (выполнение MongoDB-запросов и создание локального RDF-хранилища) составило в среднем 0,72 с после 1000 запусков; пол-

Статистические данные по структуре онлайн-курсов

Курс	Секции	Подсекции	Юниты	Концепты	Уникальные концепты
Applications of Linear Algebra Part 1	8	49	50	1072	559
Applications of Linear Algebra Part 2	9	58	86	1610	811
Introduction to Differential Equations	18	45	106	1890	639
Introduction to Differential Equations	5	28	181	3346	1078
LAFF On Programming for Correctness	8	39	181	1871	654
Pre-University Calculus	9	71	238	1950	770

ное время работы (MongoDB-запросы, SPARQL-запросы к DBpedia, выявление концептов и их анализ) заняло около 2 ч; общее число триплетов — 32 590. Подробная информация по полученным данным представлена в таблице.

Заключение

Задачи по разработке и документированию онтологии выполнены в полной мере, экстракция данных из онлайн-курсов платформы Open edX осуществлена. Автоматическое выявление концептов также реализовано, при этом в дальнейшем следует улучшить качество выявления терминов с точки зрения лексики языка. Однако уже полученный набор данных, представленный в семантическом виде, позволяет использовать их для формирования:

- междисциплинарных связей;
- рекомендаций при прохождении курсов;
- нелинейной траектории изучения материала по одному курсу с учетом всех курсов, представленных на учебной платформе;
- рекомендаций для создателей курса на основе анализа корреляции между концептами учебных материалов и тестов.

Прделанная работа является основой для создания интерфейсов визуализации динамических переходов между учебными материалами, рекомендательных сервисов для студентов и разработчиков курсов. Все материалы представлены в открытом репозитории лаборатории "Интеллектуальные методы обработки информации и семантические технологии" на сайте <https://github.com/ailabito/edX-ontology>.

Список литературы

1. Андреев А. А. Введение в дистанционное обучение. 1997. С. 9—13.

2. Полат Е. С. Дистанционное обучение: организационные и педагогические аспекты // Информатика и образование. 1996. № 3. С. 87—91.

3. Бадарч Д., Токарева Н., Цветкова М. МООК: реконструкция высшего образования // Высшее образование в России. 2014. № 10. С. 135—145.

4. Wilson L., Gruz A. MOOCs — international information and education phenomenon? // Bulletin of the Association for Information Science and Technology. 2014. Т. 40. № 5. С. 35—40.

5. Харабет Я. К. Автоматическое выделение количественных конструкций в русскоязычных научно-популярных текстах // XVIII Объединенная научная конференция "Интернет и современное общество" (IMS—2015). Санкт-Петербург. 2015. С. 23—25.

6. Гаврилова Т. А. Онтологический подход к управлению знаниями при разработке корпоративных информационных систем // Новости искусственного интеллекта. 2003. № 2. С. 24—30.

7. Гаврилова Т. А. Онтологический инжиниринг // Докл. Восьмой науч.-практ. конф. "Реинжиниринг бизнес-процессов на основе современных информационных технологий. Системы управления знаниями" (РБП-СУЗ-2005). 2005. С. 79—82.

8. Хорошевский В. Ф. Семантические технологии: ожидания и тренды // Открытые семантические технологии проектирования интеллектуальных систем: Тр. II Междунар. науч.-техн. конф. Беларусь, Минск: БГУИР. 2012. С. 143—158.

9. Ховлова М. В. Исследование лексико-синтаксической сочетаемости в русском языке с помощью статистических методов (на базе корпусов текстов) // АКД. СПб. 2010. С. 26—32.

10. Большакова Е. И., Васильева Н. Э. Формализация лексико-синтаксической информации для распознавания регулярных конструкций естественного языка // Программные продукты и системы. 2008. № 4. С. 103—106.

11. Палагин А. В., Петренко Н. Г. К вопросу системно-онтологической интеграции знаний предметной области // Математические машины и системы. 2007. Т. 1. № 3—4.

12. Vas R. Educational ontology and knowledge testing // The Electronic Journal of Knowledge Management. 2007. Т. 5. № 1. С. 123—130.

13. Berners-Lee T., Hendler J., Lassila O. The semantic web. Scientific American. 2001. № 5. P. 28—37.

A. A. Romanov, Teacher, e-mail: gloomspb@gmail.com,
D. G. Volchek, Teacher, e-mail: dvolchek@yandex.ru,
D. I. Mouromtsev, Professor, e-mail: mouromtsev@corp.ifmo.ru,
ITMO University, Laboratory of Information Science and Semantic Technologies,
St. Petersburg, Russian Federation

Towards Semantic MOOC: Extracting, Enriching and Interlinking E-learning Data in Open edX Platform

The number of educational materials available through open online courses gradually increases. Their growing number negatively affects quality of certain materials and makes it harder for end users to classify information and search for data. The offered solution is based on the principles of semantic technologies for Open edX platform. It includes extraction, enrichment and interlinking of educational data. For the purposes of method effectiveness study and method implementation, special components were developed and tested.

Keywords: semantic web, education, linked data in education, educational ontology, e-learning, educational semantic technologies

DOI: 10.17587/it.24.553-560

References

1. **Andreev A. A.** *Vvedenie v distancionnoe obuchenie* (Introduction to distance learning), 1997 (in Russian).
2. **Polat E. S.** Distancionnoe obuchenie: organizacionnye i pedagogicheskie aspekty (Distance learning: organizational and pedagogical aspects), *Informatika i Obrazovanie*, 1996, no. 3, pp. 87–91 (in Russian).
3. **Badarch D., Tokareva N., Cvetkova M.** MOOC: rekonstrukcija vysshego obrazovanija (MOOC: reconstruction of higher education), *Vysshee Obrazovanie v Rossii*, 2014, no. 10 (in Russian).
4. **Wilson L., Gruzd A.** MOOCs — international information and education phenomenon?, *Bulletin of the Association for Information Science and Technology*, 2014, vol. 40, no. 5, pp. 35–40.
5. **Harabet Ja. K.** Avtomaticheskoe vydelenie kolichestvennykh konstrukcij v russkojazychnykh nauchno-populjarnykh tekstah (Automatic selection of quantitative structures in the Russian popular science texts), *XVIII Ob#edinennaja nauchnaja konferencija "Internet i sovremennoe obshhestvo" (IMS—2015)*, Sankt-Peterburg, 2015, pp. 23–25 (in Russian).
6. **Gavrilova T. A.** Ontologicheskij podhod k upravleniju znanijami pri razrabotke korporativnykh informacionnykh sistem (Ontological approach to knowledge management in the development of corporate information systems), *Novosti Iskusstvennogo Intellekta*, 2003, no. 2 (in Russian).
7. **Gavrilova T. A.** Ontologicheskij inzhiniring (Ontology engineering), *Dokl. Vos'moj nauchno-prakticheskoy konferencii "Re-*
8. **Horoshevskij V. F.** Semanticheskie tehnologii: ozhidaniya i trendy (Semantic technologies: expectations and trends), *Otkrytye semanticheskie tehnologii proektirovanija intellektual'nykh sistem: Tr. II Mezhdunar. nauch.-tehn. konf. Belarus'*, Minsk, BGUIR, 2012, pp. 143–158 (in Russian).
9. **Hohlova M. V.** *Issledovanie leksiko-sintaksicheskoy sochetnosti v russkom jazyke s pomoshh'ju statisticheskikh metodov (na baze korpusov tekstov)* (Lexico-syntactic compatibility in the Russian language with the help of statistical methods (based on text corpora)), AKD, SPb., 2010 (in Russian).
10. **Bol'shakova E. I., Vasil'eva N. Je.** Formalizacija leksiko-sintaksicheskoy informacii dlja raspoznavanija reguljarnykh konstrukcij estestvennogo jazyka (The formalization of lexico-syntactic information to recognize regular structures in natural language), *Programmnye Produkty i Sistemy*, 2008, no. 4, pp. 103–106 (in Russian).
11. **Palagin A. V., Petrenko N. G.** K voprosu sistemno-ontologicheskoy integracii znanij predmetnoj oblasti (To the issue of system-ontological integration of knowledge of subject area), *Matematicheskie Mashiny i Sistemy*, 2007, vol. 1, no. 3–4 (in Russian).
12. **Vas R.** Educational ontology and knowledge testing, *The Electronic Journal of Knowledge Management of*, 2007, vol. 5, no. 1, pp. 123–130.
13. **Berners-Lee T., Hendler J., Lassila O.** The semantic web, *Scientific American*, 2001, vol. 284, no. 5, pp. 28–37.

Адрес редакции:

107076, Москва, Стромьинский пер., 4

Телефон редакции журнала (499) 269-5510

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова*.

Корректор *Е. В. Комиссарова*.

Сдано в набор 06.06.2018. Подписано в печать 23.07.2018. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ ИТ818. Цена договорная.

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

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

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

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.