

УДК 004.94

DOI: [10.26102/2310-6018/2020.28.1.005](https://doi.org/10.26102/2310-6018/2020.28.1.005)

Моделирование и оптимизация адаптивных многокомпонентных систем на базе алгебраических структур

Д.В. Жевнерчук, П.С. Кулясов

*Нижегородский государственный технический университет им. Р.Е. Алексева,
Нижний Новгород, Россия*

Резюме: В работе рассматриваются вопросы моделирования и оптимизации многокомпонентных систем. Построена классификация интерфейсов по операциям, осуществляемым над ними, и введены соответствующие обозначения для каждого из них. Предложена трехмерная структура, описывающая и систематизирующая операции над интерфейсами компонентов моделируемой системы («интеграция», «сопряжение» и «фильтрация»), представляющая собой невзвешенный ориентированный граф, в каждом из трех измерений обладающий свойствами алгебраической решетки. Показано, что «интеграция», «сопряжение» и «фильтрация» задают отношение частичного порядка на множестве интерфейсов, также приводится их представление в виде алгебраической решетки, являющейся индексирующей структурой. Такая структура позволит строить эффективные информационно-поисковые алгоритмы, ускоряющие решение задач сборки и реконfigurирования многокомпонентных систем. В частности, была предложена модификация алгоритма поиска в глубину, в которой учтены конструктивные особенности решетки, которая обладает меньшей вычислительной сложностью, чем классический, и не обладает его основными недостатками – неполнотой и неоптимальностью при поиске с ограничением глубины. Приводится пример использования полученных в работе результатов для моделирования образовательных программ высшего образования как многокомпонентных структур, представленных в виде набора компонентов, взаимодействующих посредством стандартизированных интерфейсов.

Ключевые слова: многокомпонентная система, алгебраическая модель, стандартизированный интерфейс, решетка.

Для цитирования: Жевнерчук Д.В., Кулясов П.С. Моделирование и оптимизация адаптивных многокомпонентных систем на базе алгебраических структур. *Моделирование, оптимизация и информационные технологии.* 2020;8(1). Доступно по: https://moit.vivt.ru/wp-content/uploads/2020/02/ZhevnerchukKulyasov_1_20_1.pdf DOI: 10.26102/2310-6018/2020.28.1.005

Modeling and optimization of adaptive multicomponent systems based on algebraic structures

D.V. Zhevnerchuk, P.S. Kulyasov

*Nizhny Novgorod State Technical University n.a. R.E. Alekseev,
Nizhny Novgorod, Russia*

Abstract: The paper describes the issues of modeling and optimization of multicomponent systems. The classification of interfaces by types is presented and the corresponding notation is introduced for each of them. A three-dimensional structure is proposed that describes and systematizes operations on the interfaces of the components of the simulated system (integration, conjugation, and filtering), which is an unweighted directed graph that has the properties of an algebraic lattice in each of the three dimensions. For each type of operation, a partial order relation is substantiated on a set of interfaces and an algebraic lattice representation is presented with justification. The proposed structure can be used as an index. With its help, a quick search for the desired interface can be carried out, for optimization of which a depth search algorithm is proposed, which is modified taking into account the design features

of the graph. The proposed algorithm has less computational complexity than the classical one, and does not have its main drawbacks - incompleteness and inoptimality when searching with depth restriction. It also shows the application of the proposed approach to the modeling of multicomponent systems based on algebraic structures using the example of a higher education educational program presented as a set of components interacting via standardized interfaces.

Keywords: multicomponent system, algebraic model, standardized interface, lattice.

For citation: Zhevnerchuk D.V., Kulyasov P.S. Modeling and optimization of adaptive multicomponent systems based on algebraic structures. *Modeling, Optimization and Information Technology*. 2020;8(1). Available from: https://moit.vivt.ru/wp-content/uploads/2020/02/ZhevnerchukKulyasov_1_20_1.pdf DOI: 10.26102/2310-6018/2020.28.1.005 (In Russ).

Введение

В настоящее время одной из наиболее острых проблем в сфере информационных технологий является так называемая «проблема интероперабельности», заключающаяся в необходимости обеспечения взаимодействия разнородных элементов (систем) при формировании гетерогенной информационно-коммуникационной среды, что является ключевой особенностью развития современных информационно-коммуникационных систем [1]. Также необходимо принять во внимание, что базовыми элементами открытых информационных систем (ОИС) являются компоненты, представляющие собой блоки-преобразователи информации, обладающие входными и выходными интерфейсами, которые должны быть стандартизированы и поддерживаться системой спецификаций [2]. Таким образом, можно утверждать, что проблема интероперабельности существует для информационных систем всех классов, и для ее разрешения применяют методы функциональной стандартизации, основанные в том числе на принципах открытых систем [1]. При этом для моделирования информационных систем применяется компонентный подход.

В настоящее время существуют следующие направления исследования и проектирования ОИС [3,4]: технология открытых систем и обеспечение интероперабельности информационных систем (Гуляев Ю.В., Олейников А.Я., Журавлев Е.Е., Батоврин В.К., Кочуков А.Н.), особенности интероперабельности в различных сферах применения (Chen D., Doumeingts G., Vernadat F., Батоврин В.К., Олейников А.Я., Журавлев Е.Е., Каменщиков А.А., Riedel M., Streit A., Lippert T., Wolf F., Kranzlmüller D., Петров А.Б.), применение принципов и технологий открытых систем в промышленности и образовании (Батоврин В.К., Разинкина Е.М., Меркулова А.В., Кочуков А.Н., Ипатов Ю.В.).

При моделировании ОИС, в основе которых лежат многокомпонентные структуры большой размерности, приходится решать вопросы разрешения вычислительной сложности алгоритмов их сборки и реконfigurирования. В [5] предлагается обобщенный метод синтеза интероперабельных структур, включающий концептуальные каркасы онтологий, формализующие иерархии многокомпонентных систем. Однако при этом дальнейшего развития требует подход к обработке сопряжения компонентов системы посредством интерфейсов, на которые происходит отображение пары доменов свойств, определяющихся общим доменом спецификации и через связанную с ним точку доступа существует канал обмена данными между компонентами.

Материалы и методы

В ОИС интерфейсы описывают способность к взаимодействию между компонентами системы. Набор интерфейсов (количество, типы и структура) влияет на

сборку многокомпонентных систем и определяет их конфигурацию. На Рисунке 1 представлена классификация основных типов интерфейсов и их обозначения.

Можно выделить три основных вида операций над интерфейсами (Рисунок 2).



Рисунок 1 – Типы интерфейсов и их обозначения
Figure 1 – Types of interfaces and their designations



Рисунок 2 – Операции над интерфейсами и их обозначения
Figure 2 – Operations on interfaces and their designations

1. Интеграция интерфейсов

Операция интеграции представляет собой объединение нескольких интерфейсов одного типа в один сложный (составной). Он может включать в себя несколько простых интерфейсов, причем уровней вложенности может быть несколько (Рисунок 3).

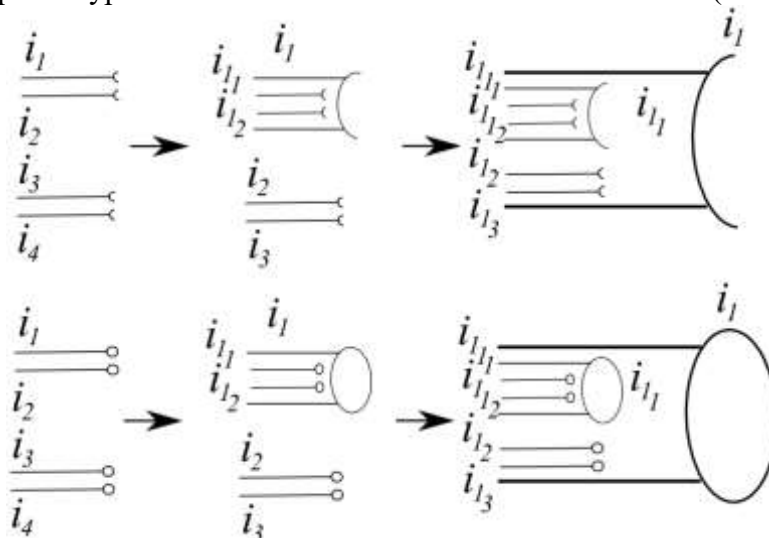


Рисунок 3 – Интеграция интерфейсов

Figure 3 – Interfaces Integration

Множество интерфейсов I является упорядоченным [6], поскольку для любых элементов этого множества определено бинарное отношение $i_{k_m} \subseteq i_k$ (вложенность интерфейсов) и для любого интерфейса $i_k \in I$ будут выполняться следующие условия:

- рефлексивности ($i_k \subseteq i_k$); (1)

- антисимметричности (если $i_{k_m} \subseteq i_{k_n}$ и $i_{k_n} \subseteq i_{k_m}$, то $i_{k_n} = i_{k_m}$); (2)

- транзитивности (если $i_{k_m} \subseteq i_k$ и $i_{k_{m_l}} \subseteq i_{k_m}$, то $i_{k_{m_l}} \subseteq i_k$). (3)

Согласно [7], частично упорядоченное множество является решеткой, если для любой пары подмножеств этого множества существует один и только один элемент - нижняя граница и аналогично существует один и только один элемент - верхняя граница (4). Как можно заметить на Рисунке 4, структура, описывающая интеграцию интерфейсов удовлетворяет этому определению. Нулевой элемент вводится искусственно, однако он не вносит никаких противоречий.

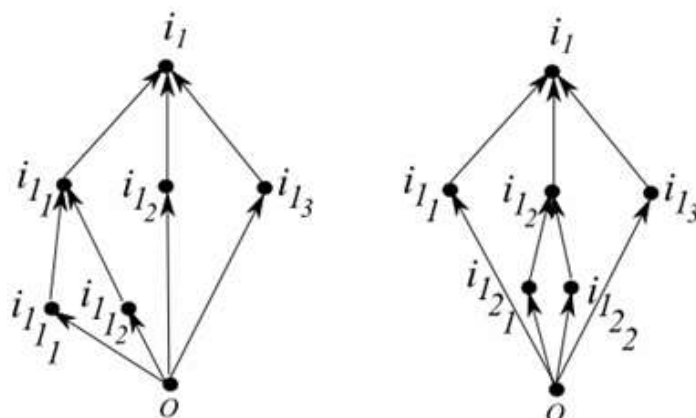


Рисунок 4 – Интеграция интерфейсов (представление в виде решетки)
 Figure 4 – Interfaces Integration (lattice representation)

2. Сопряжение интерфейсов

Все интерфейсы могут быть разделены на две категории: интерфейсы типа in (входные) и типа out (выходные). Для любого интерфейса i типа in на множестве всех интерфейсов I может быть определено подмножество $I_1 = \{i_1, i_2, \dots, i_n\}$ интерфейсов типа out , с которыми он может быть сопряжен (Рисунок 5, а). В свою очередь для интерфейсов типа out , входящих в подмножество I_1 , может быть определено подмножество $I_2 = \{i_2, i_2, \dots, i_2\}$ интерфейсов типа in , с которыми они могут быть сопряжены (Рисунок 5, б). И так далее (Рисунок 5, в). Количество таких подмножеств n будет зависеть от рассматриваемой системы и от сложности ее структуры. Аналогично для интерфейса i типа out (Рисунок 6, а, б, в).

С точки зрения операции сопряжения интерфейсов, множество I является частично упорядоченным [4], поскольку для любых элементов этого множества определено бинарное отношение $i_k \subseteq i_m$ (сопряжение интерфейсов) и для любого интерфейса $i_k \in I$ будут выполняться условия 1-3. Свойство 4 так же выполняется.

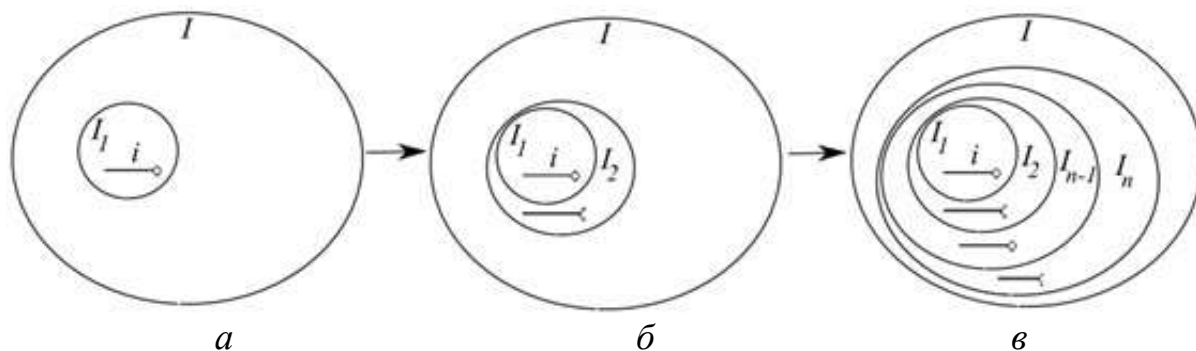


Рисунок 5 – Сопряжение интерфейсов для интерфейса типа out (представление в виде подмножеств)

Figure 5 – Conjugation of interfaces for an interface of type out (representation as subsets)

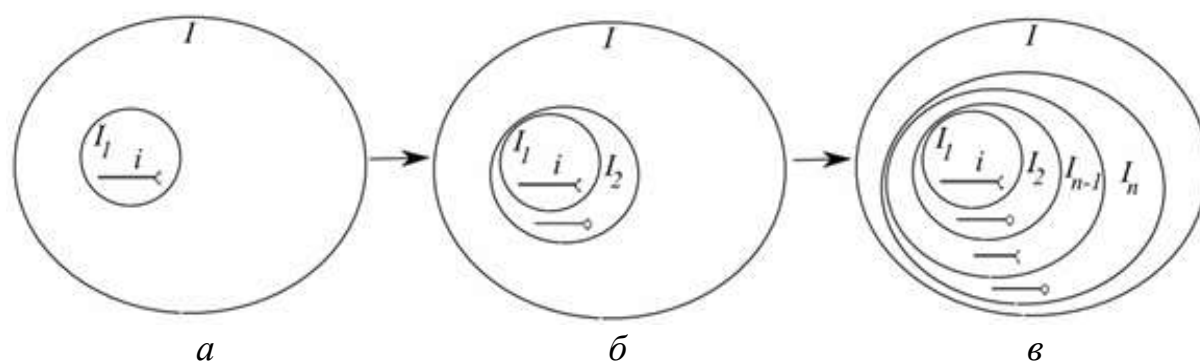


Рисунок 6 – Сопряжение интерфейсов для интерфейса типа in (представление в виде подмножеств)

Figure 6 – Conjugation of interfaces for an interface of type in (representation as subsets)

В результате операция сопряжения интерфейсов может быть представлена структурой, имеющей вид решетки (Рисунок 7).

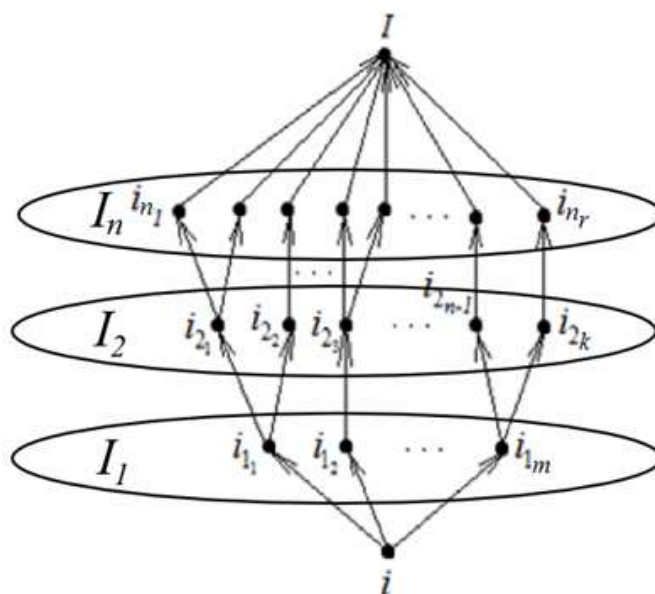


Рисунок 7 – Сопряжение интерфейсов (представление в виде решетки)
 Figure 7 – Conjugation of interfaces (representation as lattice)

3. Фильтрация интерфейсов

С точки зрения данной операции, интерфейсы делятся на две категории: фильтруемые и фильтрующие. Следует отметить, что набор интерфейсов, являющихся фильтрующими, может зависеть от рассматриваемого подмножества компонентов системы. Для ее выполнения должен быть задан критерий и из всех интерфейсов, которые сопрягаются с фильтрующим, будут выбраны только те, которые ему соответствуют (Рисунок 8).

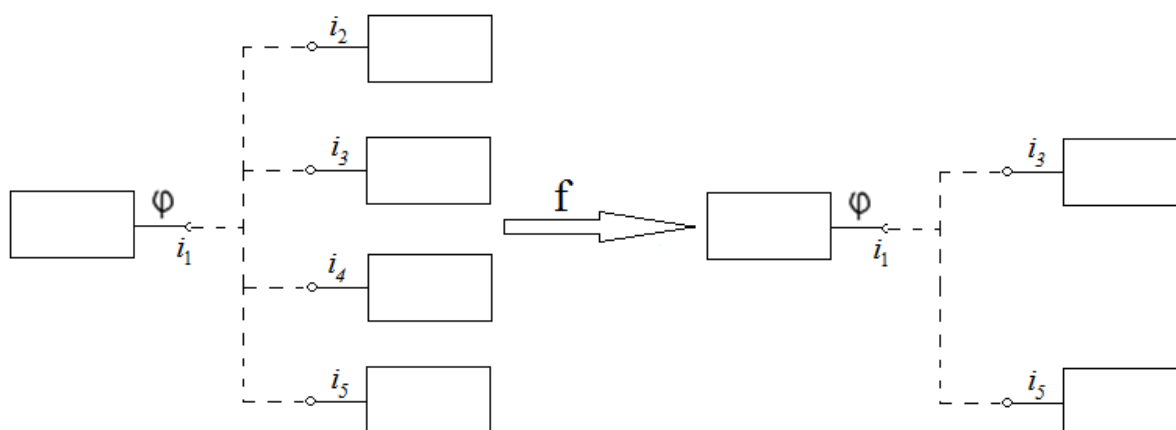


Рисунок 8 – Фильтрация интерфейсов
 Figure 8 – Filtering of interfaces

Поскольку операция фильтрации интерфейсов выполняется над сопряженными интерфейсами, для нее так же может быть использовано представление в виде решетки (Рисунок 9).

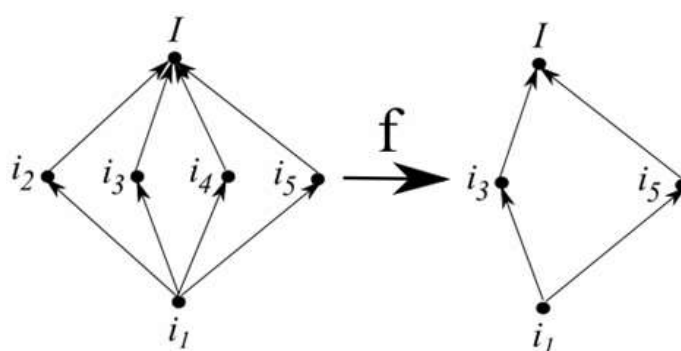


Рисунок 9 – Фильтрация интерфейсов (представление в виде решетки)
 Figure 9 – Filtering of interfaces (representation as lattice)

Алгоритм сопряжения и фильтрации компонентов на основе алгебраической решетки

Объединив структуры, описывающие эти три вида операций над интерфейсами в одну общую, получим трехмерную структуру, описывающую и систематизирующую

взаимодействие интерфейсов компонентов системы (на Рисунке 10 представлена такая структура, построенная для примера на Рисунке 11). В каждом из трех измерений часть этой структуры описывает один из трех видов операций над интерфейсами компонентов системы (интеграция, сопряжение и фильтрация) и представляет собой алгебраическую решетку. Поскольку все операции выполняются на общем множестве интерфейсов, то будет присутствовать избыточность за счет дублирования интерфейсов в двух или во всех трех фрагментах решетки.

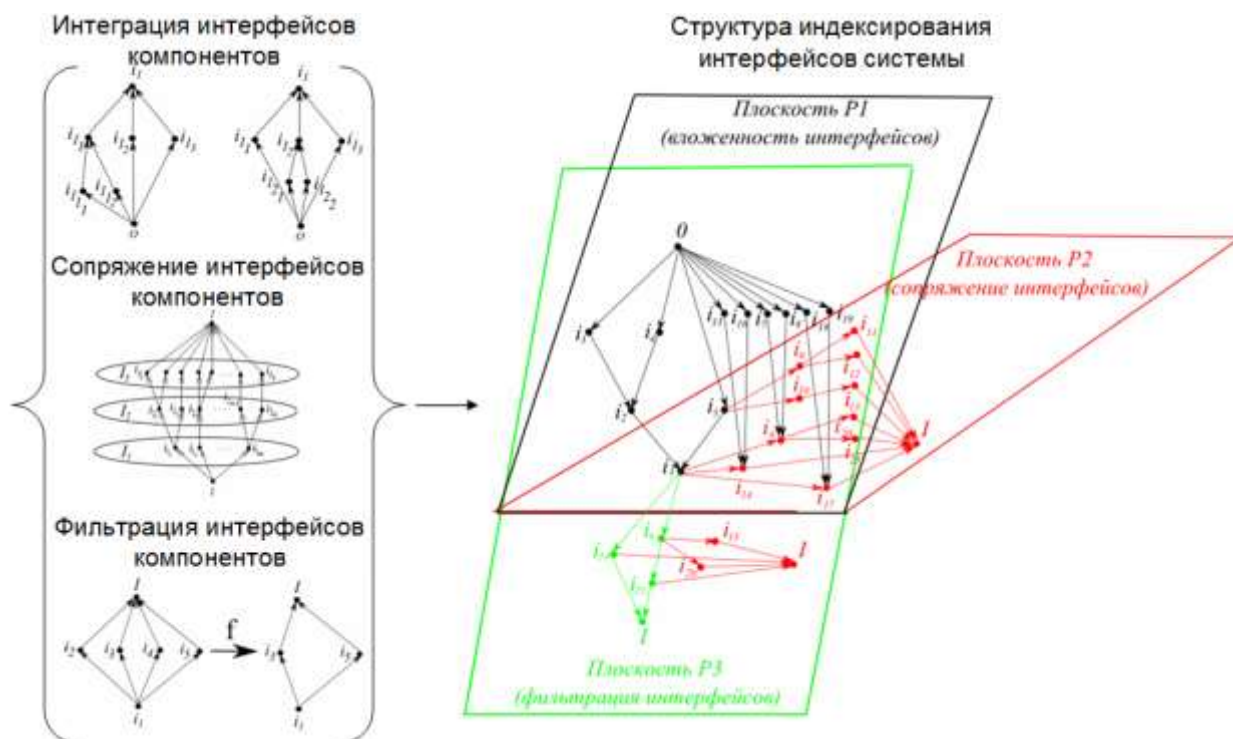


Рисунок 10 – Структура, описывающая взаимодействие интерфейсов компонентов системы

Figure 10 – The structure that describes the interaction of the interfaces of the system components

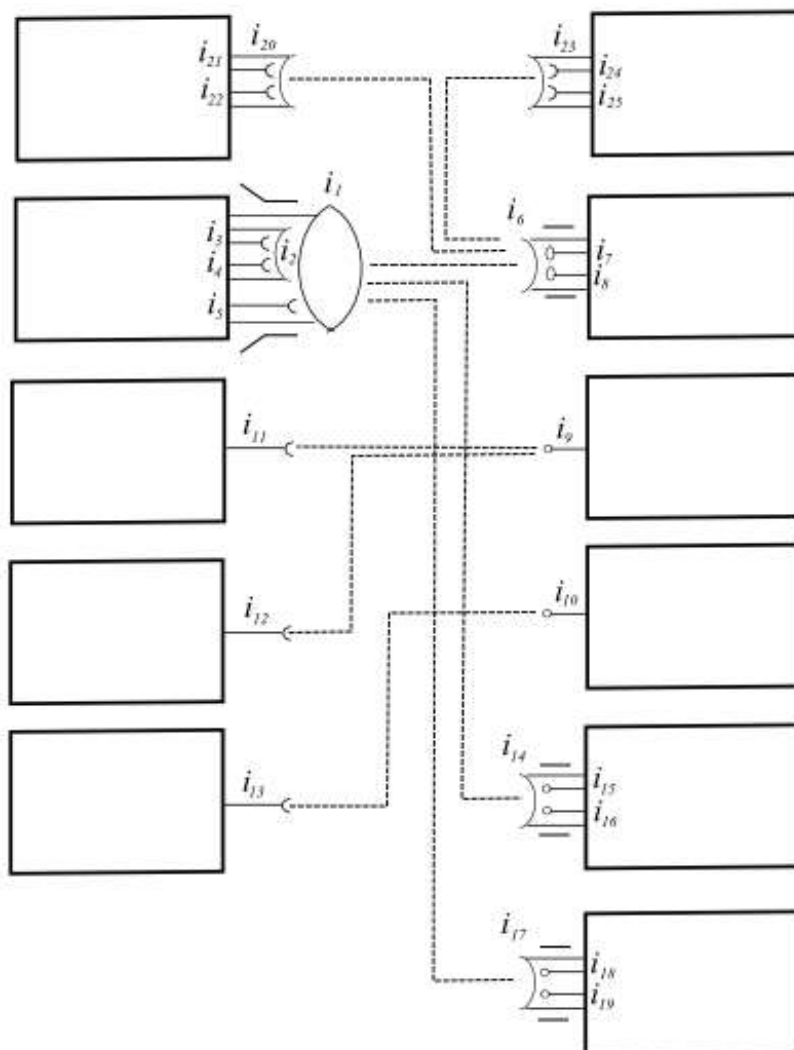


Рисунок 11 – Пример фрагмента моделируемой системы
 Figure 11 – An example of a fragment of a modelled system

Полученная структура может быть использована в качестве поискового индекса [8], назначение которого заключается в повышении скорости поиска нужных компонентов для сборки и реконфигурирования целевой системы. Дополнительная память, выделяемая для хранения индекса, и увеличение времени, требуемого для его обновления, компенсируются уменьшением времени на поиск информации.

Поскольку полученная структура представляет собой невзвешенный ориентированный граф [9], в качестве алгоритма поиска целесообразно использовать алгоритм поиска в глубину. Его основным недостатком является значительная нагрузка на стек вызовов, которая может привести к переполнению стека (stack overflow). Однако для данного случая алгоритм может быть модифицирован с учетом некоторых конструктивных особенностей графа (на Рисунке 12 изображена блок-схема предложенного алгоритма):

1. Поскольку в каждом из измерений граф имеет форму алгебраической решетки, обладающей максимальным и минимальным элементом, можно точно определить необходимый предел глубины поиска I , что позволит, решить проблему бесконечного пути, а также обойти основные недостатки поиска с ограничением глубины: неполнота

(при $I < d$, где d - глубина поиска) и неоптимальность (при $I > d$). Благодаря этому временная сложность алгоритма поиска будет снижена с $O(b^{d+1})$ до $O(b^I)$, а пространственная с $O(b^{d+1})$ до $O(b \cdot I)$, где b - коэффициент ветвления графа.

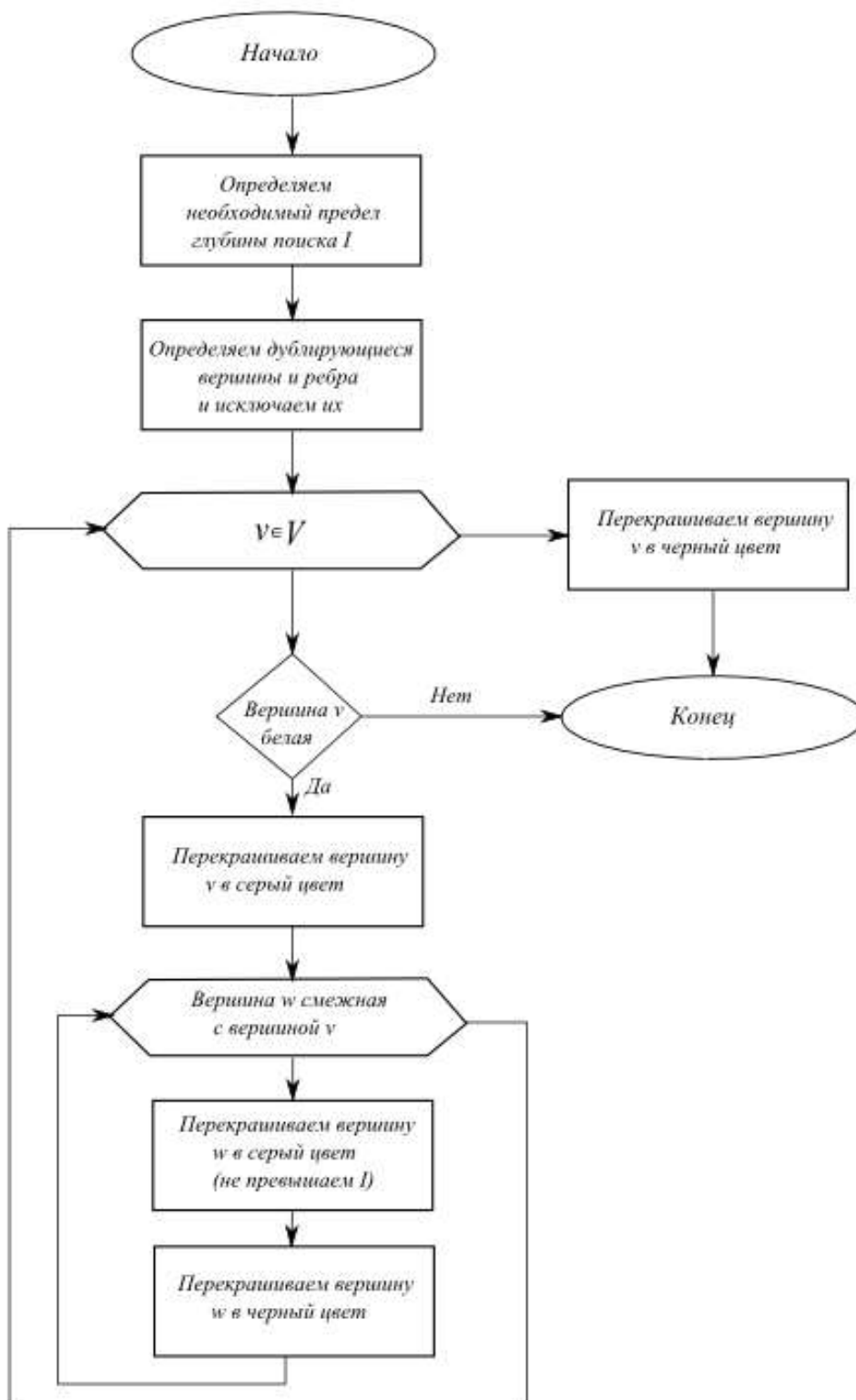


Рисунок 12 – Блок-схема алгоритма
 Figure 12 – Flowchart of algorithm

2. Поскольку граф обладает избыточностью в плоскости сопряжения интерфейсов P_2 , необходимо исключить из рассмотрения дублирующиеся ребра и вершины, для определения которых можно использовать матрицу смежности. Так как граф является невзвешенным и в нем отсутствуют петли, соответствующие элементы матрицы будут иметь значения больше единицы.

С учетом модификации алгоритм поиска будет выглядеть следующим образом:

Пусть задан граф $G\{V, E\}$, где V - множество вершин графа, E - множество ребер графа. Предположим, что в начальный момент времени все вершины окрашены в белый цвет.

Шаг 1. Определяем необходимый предел глубины поиска I .

Шаг 2. Определяем дублирующиеся ребра и вершины и исключаем их из рассмотрения.

Шаг 3. Пройдем по всем вершинам $v \in V$. Если вершина v белая, выполняем шаги 4-7. В противном случае работа алгоритма завершается.

Шаг 4. Перекрашиваем вершину v в серый цвет.

Шаг 5. Для всякой вершины w , смежной с вершиной v и окрашенной в белый цвет, рекурсивно выполняем шаги 4-7. При выполнении данного шага не превышаем предел глубины поиска I .

Шаг 6. Перекрашиваем вершину v в черный цвет.

Предложенный алгоритм позволяет оптимизировать процесс сборки и реконfigurирования многокомпонентной системы за счет сокращения времени поиска нужных компонентов и уменьшения объема требуемых вычислительных ресурсов, поскольку обладает меньшей вычислительной сложностью, чем классический алгоритм поиска на графах.

Применение полученных результатов к моделированию и оптимизации образовательных программ как многокомпонентных систем

Рассмотрим особенности применения полученных результатов для моделирования образовательных программ высшего образования (ОПВО), в основе которых лежит блочное представление учебных планов, дисциплин и обеспечивающих их ресурсов, а также требований и ограничений внешней среды, как элементов информационных систем, обладающих стандартизированными интерфейсами [10]. Использование предлагаемой структуры (Рисунок 8), описывающей взаимодействие интерфейсов компонент моделируемой ОПВО позволит осуществлять оптимизацию за счет быстрого поиска нужных компонентов, а поддержка операции фильтрации позволит достичь адаптивности моделируемой системы к изменениям, диктуемым внешним окружением (выход новых образовательных стандартов, изменения требований работодателей), а так же внутренней необходимостью (усовершенствование и актуализация учебных курсов, кадрового и материально-технического обеспечения, оптимизация за счет удаления "избыточных" учебных курсов).

В качестве примера приведем фрагмент ОПВО по направлению 09.03.01 "Информатика и вычислительная техника" (Рисунок 13) и построенную для него структуру (Рисунок 14, критерий фильтрации - "все дисциплины, которые могут проводиться в выбранной аудитории").

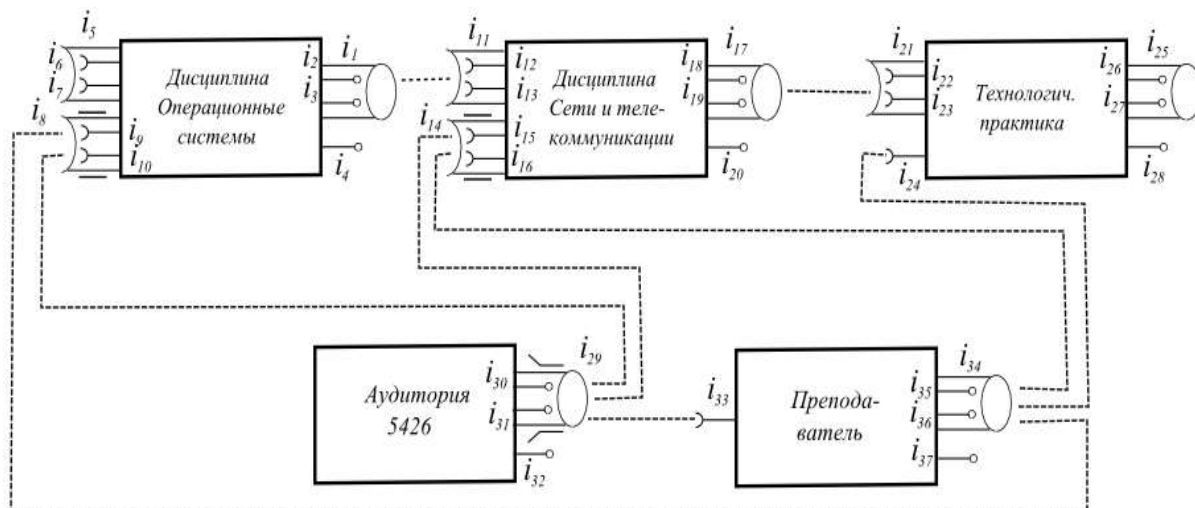


Рисунок 13 – Пример фрагмента моделируемой ОПВО
 Figure 13 – An example of a fragment of a modelled OPVO

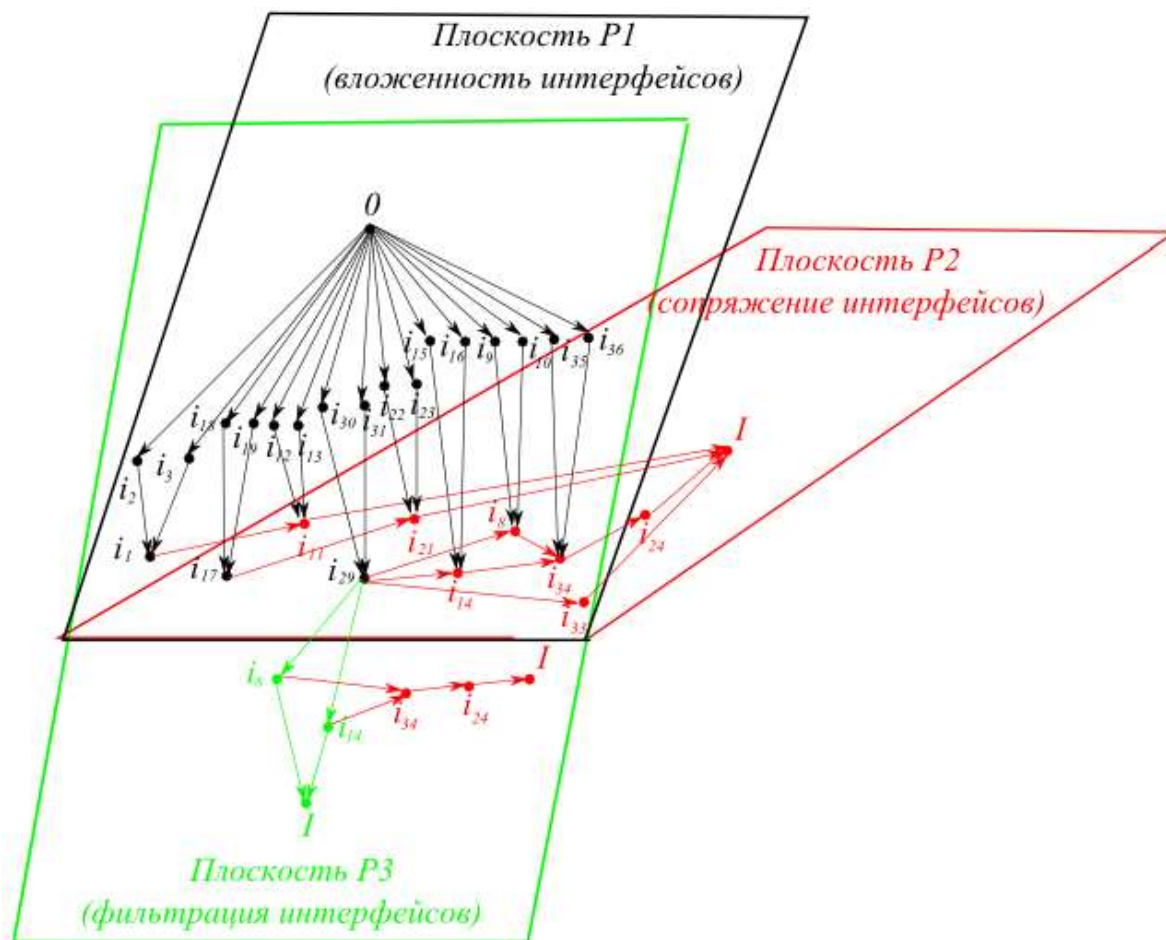


Рисунок 14 – Структура, описывающая взаимодействие интерфейсов
 компонентов ОПВО

Figure 14 – Structure describing the interaction of the interfaces
 of the components of OPVO

Заключение

В работе показано применение алгебраических структур для моделирования систем на базе компонентного подхода. Предложена трехмерная структура, описывающая и систематизирующая набор интерфейсов, которыми обладают компоненты открытой информационной системы, на базе решеток, позволяющая индексировать интерфейсы компонентов системы и осуществлять их поиск. Предложена модификация алгоритма поиска в глубину для частного случая невзвешенного ориентированного графа, обладающая меньшей вычислительной сложностью и позволяющая осуществлять быстрый поиск, что, в свою очередь, позволяет осуществлять оптимизацию моделируемой системы. Показано применение предложенной структуры и алгоритма на примере моделирования ОПВО как многокомпонентной системы.

ЛИТЕРАТУРА

1. Гуляев Ю.В., Журавлев Е.Е., Олейников А.Я. Методология стандартизации для обеспечения интероперабельности информационных систем широкого класса. Аналитический обзор. *Журнал радиоэлектроники*. 2012;3. Доступно по: <http://jre.cplire.ru/jre/mar12/2/text.html> (дата обращения: 11.03.2020).
2. Жевнерчук Д.В. Принципы блочно-иерархической организации открытых информационных систем. Методика. Н.Новгород: НГТУ им. Р.Е. Алексеева, 2017.
3. Батоврин В.К., Гуляев Ю.В., Олейников А.Я. Обеспечение интероперабельности – основная тенденция в развитии открытых систем. *Информационные технологии и вычислительные системы*. 2009;5:7-15.
4. Гуляев Ю.В., Олейников А.Я. Открытые системы: от принципов к технологии. *Информационные технологии и вычислительные системы*. 2003;3:4-12.
5. Жевнерчук Д.В. Обобщенный метод синтеза многокомпонентных интероперабельных структур на основе онтологии и недетерминированного конечного автомата. *Информационные технологии*. 2019;2(25):67-74.
6. Биркгоф Г. Теория решеток. М.: Наука, 1984.
7. Кофман А. Введение в теорию нечетких множеств. М.: Радио и связь, 1982.
8. Дейт К.Дж. Введение в системы баз данных. 8-е издание. М.: Вильямс, 2006.
9. Уилсон Р. Введение в теорию графов. Пятое издание. СПб.: Диалектика, 2019.
10. Кулясов П.С. Обобщенный алгоритм компонентной сборки образовательных программ высшего образования. *Научно-технический вестник Поволжья*. 2019;11:24-27.

REFERENCES

1. U.V. Gulyaev. Standardization methodology to ensure the interoperability of wide-class information systems. Analytical Review / U.V. Gulyaev, E.E. Zhuravlev, A.J. Oleinikov. *Zhurnal radioelektroniki*. 2012;3. Available at: <http://jre.cplire.ru/jre/mar12/2/text.html> (In Russ) (accessed 11.03.2020).
2. D.V. Zhevnerchuk. The principles of block-hierarchical organization of open information systems. Methodology. N.Novgorod: NSTU n.a. R.E. Alekseev. 2017.
3. V.K. Batovrin, U.V. Gulyaev, A. Y. Oleinikov. Ensuring interoperability is a major trend in the development of open systems. *Informacionnye tekhnologii I vychislitel'nye sistemy*. 2009;5:7-15. (In Russ)

4. U.V. Gulyaev, A. Y. Oleinikov. Open systems: from principles to technology. *Informacionnye tekhnologii I vichslitel'nye sistemy*. 2003;3:4-12. (In Russ)
5. D.V. Zhevnerchuk. A generalized method for the synthesis of multicomponent interoperable structures based on ontology and a nondeterministic finite state machine. *Information Technology*. 2019;2(25):67-74. (In Russ)
6. G. Birkhoff. Lattice theory. M: Science, main edition of physical and mathematical literature, 1984.
7. A. Kaufmann. Introduction to the theory of fuzzy sets. M: Radio and communication, 1982.
8. C.J. Date. An introduction to database systems. M.: Vilyams, 2006.
9. R.J. Wilson. Introduction to graph theory. Fifth edition. SPb.: Dialektika, 2019.
10. P.S. Kulyasov. Generalized algorithm for component assembly of higher education educational programs. *Scientific and Technical Bulletin of the Volga Region*. 2019;11:24-27. (In Russ)

ИНФОРМАЦИЯ ОБ АВТОРЕ / INFORMATION ABOUT THE AUTHOR

Жевнерчук Дмитрий Валерьевич, д.т.н., доцент, зав. кафедрой "Вычислительные системы и технологии", ФГБОУ ВО Нижегородский государственный технический университет им. Р.Е. Алексеева, Нижний Новгород, Российская Федерация.

e-mail: zhevnerchuk@yandex.ru

ORCID: [0000-0001-6306-0893](https://orcid.org/0000-0001-6306-0893)

Dmitriy V. Zhevnerchuk, Doctor of Technical Sciences, Associate Professor, Head Department of Computing Systems and Technologies Dept., Federal State Budgetary Educational Establishment of Higher Education Nizhny Novgorod State Technical University n.a. R.E. Alekseev, Nizhny Novgorod, Russian Federation.

Кулясов Павел Сергеевич, старший преподаватель, ФГБОУ ВО Нижегородский государственный технический университет им. Р.Е. Алексеева, Нижний Новгород, Российская Федерация.

e-mail: p.kulyasov@gmail.com

Pavel S. Kulyasov, Senior Lecturer of Computing Systems and Technologies Dept, Nizhny Novgorod State Technical University n.a. R.E. Alekseev, Nizhny Novgorod, Russian Federation.