

22. Veselovskaya V.V. Servis-orientirovannaya arkhitektura v bankovskoy sfere [Service-oriented architecture in the banking industry], *Nauchnye zapiski molodykh issledovateley* [Scientific notes of young researchers], 2014, No. 2, pp. 63-69.
23. Kadochnikov A.A., Yakubaylik O. Je. Servis-orientirovannye veb-sistemy dlya obrabotki geoprostranstvennykh dannykh [Service-oriented web system for spatial data processing], *Vestnik NGU. Seriya: Informatsionnye tekhnologii* [Vestnik NSU. Series: Information technologies], 2015, No. 1, pp. 37-45.

Статью рекомендовал к опубликованию д.т.н., профессор Н.Н. Бахтадзе.

Анисимов Владимир Иванович – Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» имени В.И. Ульянова (Ленина); e-mail: vianisimov@inbox.ru; 197376, Санкт-Петербург, ул. Профессора Попова, 5, корп. I, помещение 11-53; тел.: +78122343675; кафедра САПР; д.т.н.; профессор.

Васильев Сергей Алексеевич – e-mail: venom-gt@list.ru; кафедра САПР; аспирант.

Гридин Владимир Николаевич – Центр информационных технологий в проектировании РАН; e-mail: info2@ditc.ras.ru; г. Одинцово, ул. Маршала Бирюзова, 7а; тел.: 84955960219; научный руководитель; д.т.н.; профессор.

Anisimov Vladimir Ivanovich – Saint-Petersburg State Electrotechnical University; e-mail: vianisimov@inbox.ru; Professora Popova street, 5-1 room 53, 197376 St. Petersburg, Russia, 197376; phone: +78122343675; the department of CAD; dr. of eng. sc.; professor.

Vasylev Sergey Alexeyevich – e-mail: venom-gt@list.ru; the department of CAD; postgraduate student.

Gridin Vladimir Nikolaevich – Design information technologies Center Russian Academy of Sciences; e-mail: info2@ditc.ras.ru; Odintsovo, Marshala Biryuzova, 7a; phone: +74955960219; scientific director; dr. of eng. sc.; professor.

УДК 519.17

DOI 10.23683/2311-3103-2018-4-47-58

**Л.А. Зинченко, В.А. Верстов, Б.С. Сорокин, И.В. Никитин, А.С. Бачурин,
М.В. Гусев, В.Е. Дмитриев**

АЛГОРИТМ ПОИСКА КРАТЧАЙШЕГО ПУТИ В РАСПРЕДЕЛЕННЫХ МИКРОСИСТЕМАХ С ПРЕДФРАКТАЛЬНОЙ ТОПОЛОГИЕЙ*

Рассмотрены особенности проектирования распределенных микросистем, которые могут быть использованы при создании Интернета вещей. Отмечено, что при решении практически важных задач задание веса ребра графа зависит от конкретного приложения. Показано, что в иерархических распределенных микросистемах вопросы обеспечения работоспособности отдельных узлов, являющихся ретрансляторами информации для отдельных узлов системы, являются крайне важными. Отмечено, что при использовании распределенных микросистем в лабиринтных структурах, например, в условиях городской застройки, при использовании естественных каналов передачи информации также необходимо учитывать многолучевой характер распространения радиоволн. Для повышения вероятности сохранения работоспособности проектируемых устройств при внешних помехах предложено при синтезе топологии распределенных микросистем использовать самоподобные множества, в частности фракталы. Для одного из возможных вариантов реализации предложенного подхода рассмотрено использование предфрактальных графов, являющихся конечными аналогами фрактальных графов. Были проведены вычислительные эксперименты по исследованию временной сложности наиболее эффективных алгоритмов поиска кратчайшего пути. Анализ полученных результатов позволяет сделать вывод, что

* Работа выполнена при частичной финансовой поддержке гранта РФФИ № 16-07-00676 а.

алгоритм сжатия иерархий в общем случае показывает лучшую по сравнению с другими алгоритмами производительность только для графов с небольшим количеством вершин, при этом алгоритм Дейкстры обладает лучшей асимптотической временной сложностью по сравнению с алгоритмом сжатия иерархий. Алгоритм A* показал свою асимптотическую эффективность, однако пространственная сложность алгоритма ограничивает его применение при исследовании распределенных микросистем. На основе анализа полученных экспериментальных результатов предложен алгоритм поиска кратчайшего пути, ориентированный на применение в распределенных микросистемах с предфрактальной топологией, учитывающий особенности выбранного проектного решения. Он является комбинацией алгоритма Дейкстры и алгоритма сжатия иерархий. Показано, что использование знаний позволяет снизить вычислительные затраты при поиске кратчайшего пути. Обсуждены особенности применения предложенного алгоритма при поиске пути в распределенных микросистемах с предфрактальной топологией в лабиринтных структурах.

Распределенные микросистемы; поиск кратчайшего пути; предфрактальные графы; лабиринтные структуры.

**L.A. Zinchenko, V.A. Verstov, B.S. Sorokin, I.V. Nikitin, A.S. Bachurin,
M.V. Gusev, V.E. Dmitriev**

A SHORTEST PATH ALGORITHM FOR DISTRIBUTED MICROSYSTEMS WITH PREFRACTAL TOPOLOGY

The article presents design features of distributed microsystems, which can be used to implement the Internet of Things. It should be mentioned that for solving practically important problems the weight of a graph edge depends on an application. It is shown that in hierarchical distributed microsystems the issues of reliability of individual nodes, which are information transmitters for individual nodes of the system, are extremely important. It is noted that the use of distributed microsystems in labyrinth structures, for example, in urban environments, using the natural channels of information transmission, requires calculations of multipath radio wave propagation. To increase the reliability of a system under external damage it is suggested to use self-similar sets, in particular fractals, to synthesize the topology of distributed microsystems. The use of prefractal graphs, which are finite analogs of fractal graphs, is discussed. Results of computational experiments are presented to compare the time complexity of the most effective shortest path search algorithms. A shortest path search algorithm is proposed that is oriented to application in distributed microsystems with prefractal topology, taking into account the features of the selected design solution. It is a combination of the Dijkstra's algorithm and the hierarchy contraction algorithm. It is shown that the use of knowledge allows reducing the computational costs in the search for the shortest path. The features of the proposed algorithm application in the search for a path in distributed microsystems with prefractal topology in labyrinth structures are discussed.

Distributed microsystems; prefractal graph; shortest path.

Введение. Распределенные микросистемы (распределенные МЭМС, РМЭМС) находят широкое применение в промышленности, в быту (Интернет вещей), в медицине и т.д. [1, 2, 15]. Интернет вещей является одним из перспективных концепций дальнейшего развития инфокоммуникационных технологий. Для его практической реализации необходимо обеспечение взаимодействия микросистем, в том числе мобильных. При этом часть микросистем, помимо основных функций, дополнительно используются как ретрансляторы для приема и передачи информации микросистемами, не оснащенных приемо-передающими устройствами.

На рис. 1 показан пример возможных взаимодействий между различными микросистемами в иерархической РМЭМС. Микросистемы могут обмениваться информацией как напрямую с головным устройством, так и передавать ее другим микросистемам для дальнейшей передачи и/или дополнительной обработки данных. Для обмена информацией могут быть использованы как искусственные, так и естественные каналы передачи информации (рис. 1).

В ряде практически важных применений как отдельные узлы, так и все микросистемы могут быть подвержены внешнему воздействию, при котором возможно ухудшение функционирования отдельных узлов, а в некоторых случаях и полный отказ, как отдельного узла, так и всех микросистем [3, 4, 19]. Необходимо отметить, что отказ микросистемы, выполняющей функции ретранслирующего устройства для отдельной группы микросистем в сети Интернета вещей, приведет к потере связи с этой группой микросистем.

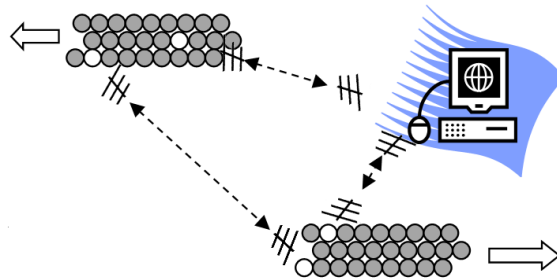


Рис. 1. Пример иерархической сети распределенных микросистем

Одним из возможных подходов к решению этой проблемы является использование при структурном синтезе распределенных микросистем самоподобных множеств, в частности фракталов [5, 18].

Самоподобные графы [6] являются комбинацией графов и самоподобных множеств, что позволяет получить принципиально новые свойства проектируемых на их основе систем. Конечными аналогами фрактальных графов, получаемых на основе нерегулярных самоподобных множеств, являются предфрактальные графы [7].

Предфрактальный граф обозначается следующим образом [7]:

$$G_L = (V_L, E_L),$$

где V_L – множество вершин графа; E_L – множество его ребер; $l = 1, 2, \dots, L$ – номер этапа порождения предфрактального графа.

Предфрактальные графы порождаются связным графом (затравкой) H путем замены каждой его вершины графом H . При этом

$$G_l = H.$$

Необходимо отметить, что большая часть структур естественного происхождения обладает свойством самоподобия [8].

В статье предложен алгоритм поиска кратчайшего пути, ориентированный на использование в распределенных микросистемах с предфрактальной топологией. Необходимо отметить, что вес ребра графа в рассматриваемых приложениях зависит от конкретного приложения. Для мониторинговых устройств критически важным параметром является низкая задержка передаваемой информации, поэтому для них предпочтительно выбирать в качестве веса ребра время задержки. Также предпочтительно использование наименьшего количества микросистем как ретранслирующих устройств, однако, из-за работы при работе на больших расстояниях пропускная способность такой системы является низкой. При необходимости передачи большого объема информации, например, видеосигнала в режиме реального времени, в качестве веса ребра эффективнее использовать пропускную способность канала. При этом предпочтительно использование большего количества микросистем как ретрансляторов, что приведет к увеличению временной задержки, однако пропускная способность такой системы будет выше, чем в первом случае.

При использовании распределенных микросистем в лабиринтных структурах, например, в условиях городской застройки, при использовании естественных каналов передачи информации также необходимо учитывать многолучевой характер распространения радиоволн [9], что приводит к нежелательным помехам в работе распределенных микросистем.

Алгоритмы для решения задачи поиска кратчайшего пути. Для работы с графами было разработано множество алгоритмов, среди которых можно выделить следующие группы алгоритмов [10]:

- ◆ Алгоритмы обхода графа.
- ◆ Алгоритмы нахождения кратчайшего пути.
- ◆ Алгоритмы нахождения остового дерева
- ◆ Алгоритмы максимального потока

Наиболее популярными алгоритмами нахождения кратчайшего пути [10–14] являются алгоритм Дейкстры, A* алгоритм и т.д.

Алгоритм Дейкстры. Алгоритм Дейкстры был предложен в 1959 г. Этот алгоритм позволяет работать только со связанными графами без ребер с отрицательным весом [10]. В процессе работы алгоритм присваивает каждой вершине некое числовое значение, которое соответствует текущему кратчайшему расстоянию до этой вершины от начальной (вес вершины). На каждом шаге алгоритма выбирается ребро с наименьшим весом из текущего множества смежных вершин и определяется вес всех вершин, которые смежны с текущей выбранной вершиной. Алгоритм продолжает свою работу до тех пор, пока все вершины не будут обработаны.

Сложность алгоритма Дейкстры зависит от выбранного представления графа [10]. При задании графа в виде массива временная сложность алгоритма $O(n^2)$, где n – число вершин графа. Для частных случаев, например, для разреженных графов время работы может быть уменьшено. Необходимо отметить, что в алгоритме Дейкстры память может использоваться неэффективно.

Алгоритм A*. Алгоритм A* был предложен в 1968 как расширение алгоритма Дейкстры [11]. Рассматриваемый алгоритм использует координатную плоскость, на которой расположен граф, выбранный для исследования. В качестве веса используется расстояние между вершинами. В процессе работы алгоритм присваивает каждой вершине числовое значение, которое соответствует сумме наиболее короткого расстояния до данной точки от начальной (вес вершины) и расстоянию от данной точки до конечной, вычисленное как кратчайшее расстояние между ними на координатной плоскости (по теореме Пифагора). На каждом этапе алгоритма выбирается вершина с наименьшим весом и определяется вес всех вершин, которые смежны с выбранной вершиной. Алгоритм продолжает свою работу до тех пор, пока конечная вершина не окажется выбранной для исследования как наименьшая по весу в данный момент.

Алгоритм поиска A* [12] является эвристическим алгоритмом, т.е. не имеет строго обоснования своего решения. Из-за своей эвристической природы сложность алгоритма зависит от эвристической функции. Также алгоритм A* зависит от использования эвристики, которая, в свою очередь, зависит от поставленной задачи. По оценкам [12], временная сложность алгоритма является полиномиальной. Необходимо отметить, что пространственная сложность этого алгоритма в худшем случае может расти экспоненциально.

Алгоритм сжатия иерархий. Алгоритм сжатия иерархий (Contraction Hierarchies) [13] был предложен в 2008 г. В [13] приведены результаты экспериментального сравнения его эффективности с другими алгоритмами поиска кратчайшего пути и показано, что он является наиболее эффективным по сравнению с другими алгоритмами поиска кратчайшего пути.

В отличие от большинства алгоритмов поиска кратчайшего пути большая часть времени работы алгоритма уходит не на вычисление самого пути, а на обработку и упрощение самого графа. В связи с этим возможно более эффективное распределение памяти и уменьшение времени работы самого алгоритма.

Алгоритм сжатия иерархий базируется на уменьшении размерности графа за один шаг на одну вершину. Для этого определяется кратчайший путь между исследуемой вершиной и всеми смежными ей вершинами, а исследуемая вершина помечается как пройденная.

На рис. 2 приведен пример работы алгоритма сжатия иерархий. На рис. 2,а показан исходный граф H . Для обработки вершины c ребру между вершинами a и d присваивается вес 2, т.к. путь $a \rightarrow c \rightarrow d$ является кратчайшим. При этом ребро между вершинами a и b с весом 1 остается неизменным, т.к. путь $a \rightarrow c \rightarrow b$ не является кратчайшим путем. На рис. 2,б показан СН-граф H' . Необходимо отметить, что вершина c не удаляется, а выполняется переход к более высокому уровню иерархии и вершина c помечается как пройденная.

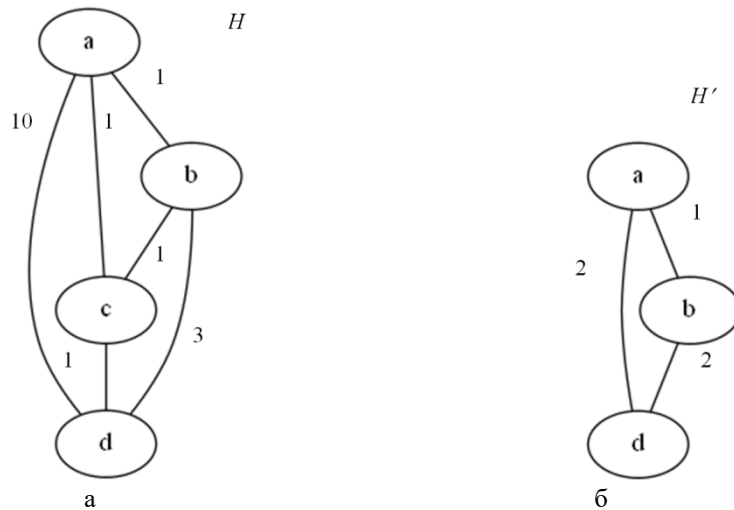


Рис. 2. Принцип работы алгоритма сжатия иерархий

На рис. 3 приведена структурная схема алгоритма. На первом шаге задается матрица смежности, при этом вес каждого ребра в графе хранится в соответствующем элементе матрицы смежности. Также необходимо задать вершины, между которыми будем искать кратчайший путь.

На следующем шаге в матрице смежности закрепляется столбец, равный номеру узла начала пути, уменьшенный на единицу, и счетчик i , соответствующий номеру строки, увеличивается. Если между вершинами существует ребро, то повторяется предыдущая операция, только теперь закрепляется столбец, равный строке, на которой алгоритм остановился, и ищется новое ребро. Затем сравнивается вес суммы двух ребер с весом ребра, смежного с рассматриваемыми вершинами. Если между вершинами не было пути или он оказался короче, то при необходимости добавляется ребро и обновляется вес суммы ребер, в противном случае ничего не меняется. Далее ребра помечаются как пройденные. Для этого вес ребер приравнивается нулю. Эти операции повторяются до тех пор, пока все ребра не будут пройдены. Алгоритм возвращает значение, являющееся кратчайшим путем между двумя исследуемыми вершинами.

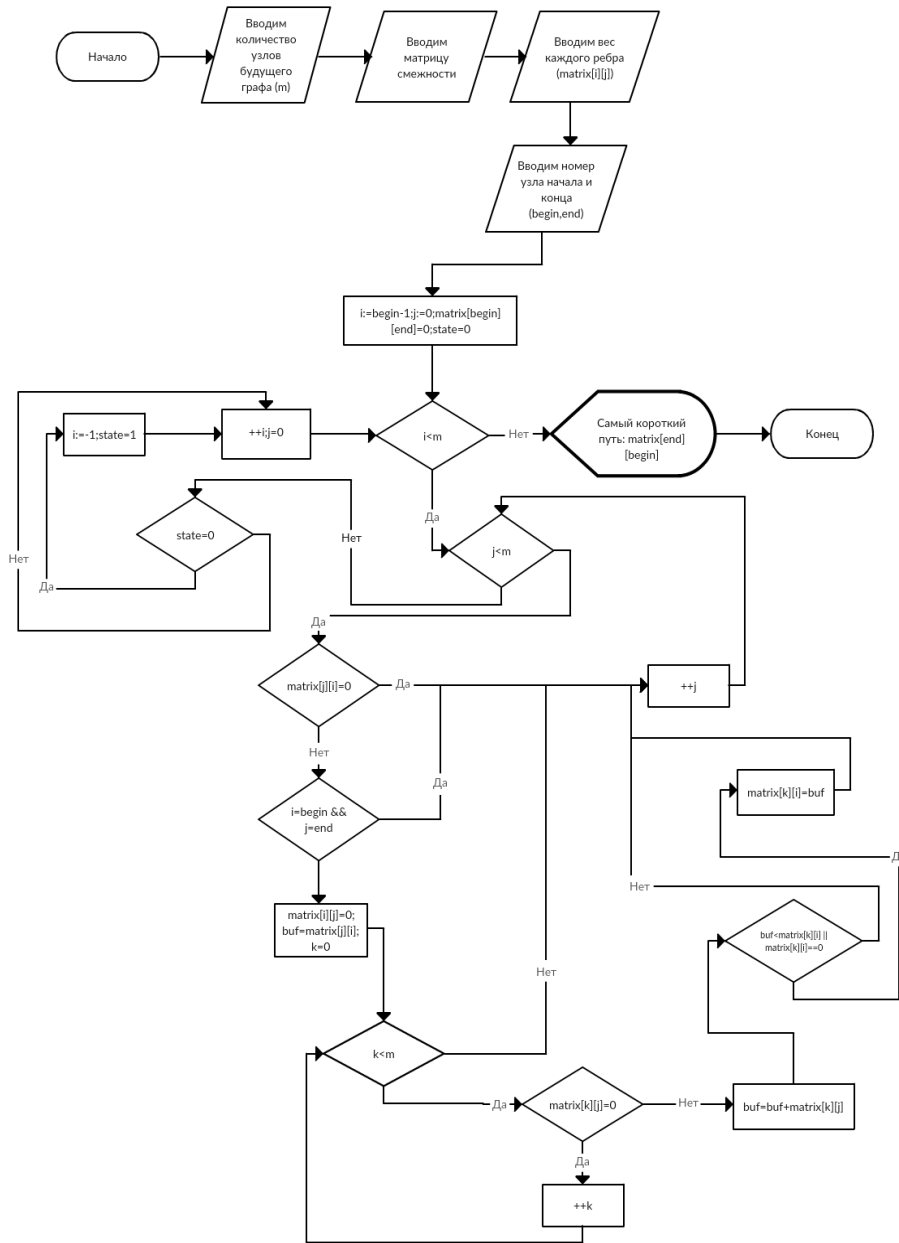


Рис. 3. Структурная схема алгоритма сжатия иерархий

Теоретические оценки поведения алгоритма сжатия иерархий в настоящее время выполнены только для частных случаев [13, 14].

В худшем случае сложность алгоритма сжатия иерархий $O(n^2)$. Для различных частных случаев, в частности, для разреженных графов, возможно уменьшение временной сложности до $O(n \log n)$.

Экспериментальные исследования. Для экспериментального исследования временной сложности рассмотренных выше алгоритмов были выбраны графы с количеством вершин n 15, 50 и 100. В табл. 1 систематизированы результаты экс-

периментальных исследований. На рис. 4 представлены графики временных затрат для рассмотренных выше алгоритмов. Кривая D получена при использовании алгоритма Дейкстры, кривая A* – при использовании алгоритма A*, а кривая CH – при использовании алгоритма сжатия иерархий.

Таблица 1

Результаты экспериментальных исследований временной сложности алгоритмов поиска кратчайшего пути

Алгоритм	$t, \text{мс}$		
	n		
	15	50	100
Дейкстры	18	29	48
A*	12,5	19,3	26
Сжатия иерархий	10,1	38	63

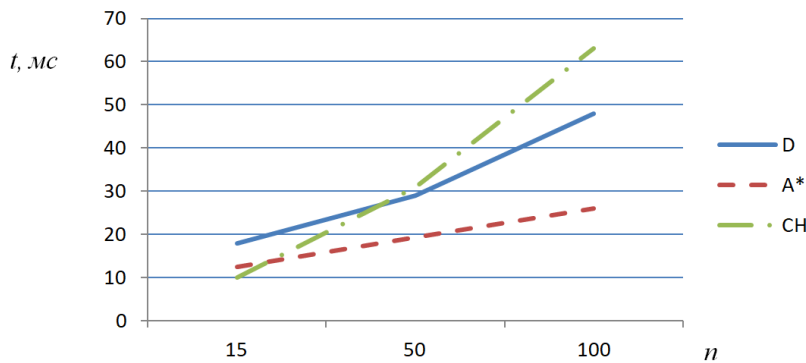


Рис. 4. Временные затраты алгоритмов поиска кратчайшего пути

n

Анализ полученных результатов позволяет сделать вывод, что алгоритм сжатия иерархий в общем случае показывает лучшую по сравнению с другими двумя алгоритмами производительность только для графов с небольшим количеством вершин, при этом алгоритм Дейкстры обладает лучшей асимптотической временной сложностью по сравнению с алгоритмом сжатия иерархий.

Алгоритм A* показал свою асимптотическую эффективность, однако пространственная сложность алгоритма ограничивает его применение при исследовании распределенных микросистем.

Алгоритм поиска кратчайшего пути в распределенных микросистемах предфрактальной топологии в лабиринтных системах. Для уменьшения времени поиска кратчайшего пути в статье предлагается следующий алгоритм поиска кратчайшего пути в распределенных микросистемах с предфрактальной топологией.

На первом этапе выполняется обработка графа H , являющегося затравкой, с использованием алгоритма сжатия иерархий. Затем выполняется преобразование исходного графа путем замены всех затравок H графом H' . На следующем шаге выполняется поиск кратчайшего пути в преобразованном графе G'_L на основе алгоритма Дейкстры.

Ниже представлен псевдокод предложенного алгоритма.

```

S_Path( $G_L$ )
{
   $H' = \text{Contraction\_Hierarchy}(H)$ ;
   $G'_L = \text{Update}(G_L, H')$ 
  Dijkstra( $G'_L$ );
}

```

На рис. 5 приведен пример предфрактального графа G_2 , порожденного за-
 травкой H (графом (рис. 2)). На рис. 6 приведен преобразованный граф G'_2 после
 сжатия иерархии. Дальнейший поиск кратчайшего пути выполняется в графе G'_2
 на основе алгоритмы Дейкстры.

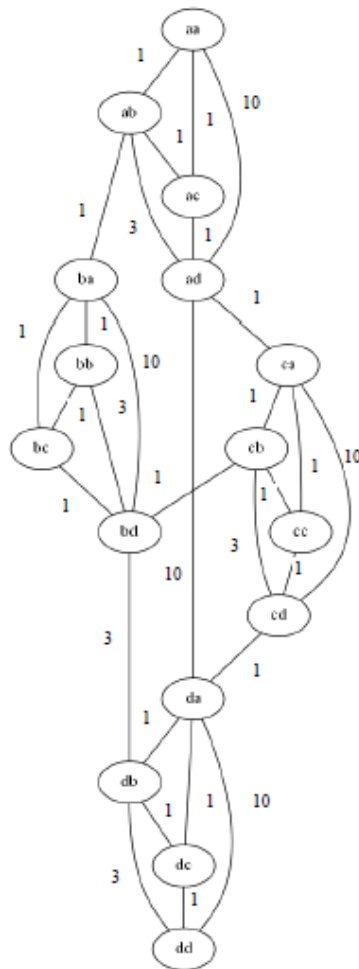


Рис. 5. Предфрактальный граф G_2

Заключение. Широкое использование распределенных микросистем требует
 разработки новых алгоритмов проектирования таких устройств [16, 20]. Задача
 усложняется при необходимости проектирования устройств, стойких к спецвоз-

действиям и предназначенных для использования в лабиринтных структурах. Одним из возможных путей повышения вероятности сохранения работоспособности сложных структур является использование свойств самоподобия, в частности, фракталов, аналогично принципам формирования структур естественного происхождения.

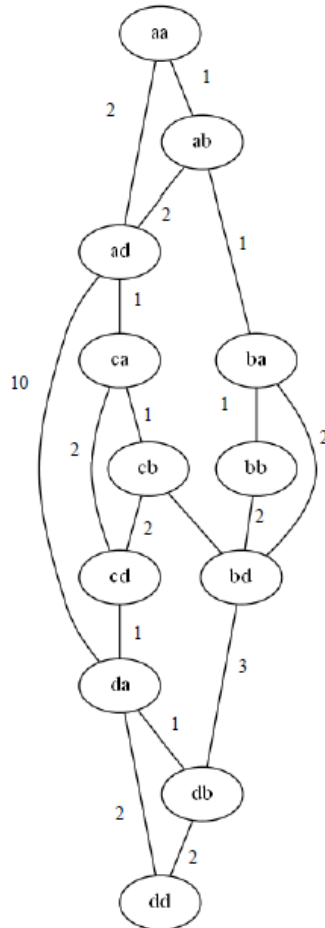


Рис. 6. Преобразованный граф G_2 после сжатия иерархии

Предфрактальные графы являются конечным аналогом фрактальных графов, что позволяет использовать их в ряде практически важных приложений, в том числе при проектировании распределенных МЭМС, стойких к спецвоздействиям.

В статье предложен алгоритм поиска кратчайшего пути в распределенных микросистемах с предфрактальной топологией, являющийся комбинацией алгоритма Дейкстры и алгоритма сжатия иерархий. Приведен пример применения этого алгоритма при поиске кратчайшего пути в предфрактальном графе. Показано, что использование знаний [17] позволяет снизить вычислительные затраты при поиске кратчайшего пути. За счет использования комбинации алгоритмов и асимптотическая временная сложность предложенного алгоритма будет меньше как временной сложности алгоритма Дейкстры, так и временной сложности алгоритма сжатия иерархий.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Bourgeois J., Goldstein S.* Distributed intelligent MEMS: progresses and perspectives. ICT Innovations 2011. Vol. 150. Advances in Intelligent and Soft Computing. – Springer Berlin, Heidelberg, 2012. – P. 15-25.
2. *Shakhnov V.A., Zinchenko L.A., Kosolapov I.A.* Simulation of distributed MOEMS for smart environments // Proc. 10th International Conference on Advanced Semiconductor Devices and Microsystems, ASDAM 2014. – 2014.
3. *Shakhnov V., Zinchenko L., Kosolapov I., Filippov I.* Modeling and Optimization of Radiation Tolerant Microsystems // EMS '14 Proceedings of the 2014 European Modelling Symposium. – 2014. – P. 484-489.
4. *Шахнов В.А., Зинченко Л.А., Терехов В.В., Михайличенко С.С.* Радиационная стойкость микроэлектромеханических систем // Наноинженерия. – 2015. – № 9. – С. 13-17.
5. *Мандельброт Б.* Фрактальная геометрия природы. – М.: Институт компьютерных исследований, 2002. – 656 с.
6. *Krön B.* Growth of self-similar graphs // Journal of Graph Theory. – 2004. – Vol. 45, No. 3. – P. 224-239.
7. *Кочкаров А.А., Кочкаров Р.А.* Параллельный алгоритм поиска кратчайшего пути на предфрактальном графе // Журнал вычислительной математики и математической физики. – 2004. – № 6. – С. 1147-1152.
8. *Пьетронеро Л., Тозатти Э. (ред.).* Фракталы в физике. – М.: Мир, 1988. – 672 с.
9. *Сорокин Б.С.* Использование леммы Лоренца для расчёта многолучевого распространения радиоволн в лабиринтных системах // Труды XV Всероссийской школы-семинара «Физика и применение микроволн». – 2015. – С. 32-34.
10. *Кормен Т., Лейзерсон Ч., Ривест Р., Штейн К.* Алгоритмы: построение и анализ. – Вильямс, 2006.
11. *Hart P. E., Nilsson N. J., Raphael B.* A Formal Basis for the Heuristic Determination of Minimum Cost Paths // IEEE Transactions on Systems Science and Cybernetics. – 1968. – № 2. – P. 100-107.
12. *Dechter R, Pearl J.* Generalized best-first search strategies and the optimality of A* // Journal of the ACM. – 1985. – Т. 32, № 3.
13. *Geisberger R., Sanders P., Schultes D., Delling D.* Contraction Hierarchies: Faster and Simpler Hierarchical Routing in Road Networks // LNCS. – 2008. – Vol. 5038.
14. *Funke S., Storz S.* Provable Efficiency of Contraction Hierarchies with Randomized Preprocessing // LNCS. – 2015. – Vol. 9472. – P. 479-490.
15. *Зинченко Л.А., Косолапов И.А., Шахнов В.А.* Особенности многомасштабного моделирования микрооптоэлектромеханических систем с учетом технологических погрешностей // Датчики и системы. – 2013. – № 9 (172). – С. 29-37.
16. *Зинченко Л.А., Сорокин Б.С.* Бионические информационные системы в проектировании микро- и наносистем // В сб.: КИИ-2008 Одиннадцатая национальная конференция по искусственному интеллекту с международным участием: труды конференции. Российская ассоциация искусственного интеллекта. – 2008. – С. 17-25.
17. *Bova V.V., Kureichik V.V., Lezhebokov A.A.* The integrated model of representation of problem-oriented knowledge in information systems // В сб.: 8th IEEE International Conference on Application of Information and Communication Technologies, AICT 2014 - Conference Proceedings 8. – 2014. – P. 7035923.
18. *Kureichik V.V., Kureichik V.M.* A fractal algorithm for graph partitioning // Journal of Computer and Systems Sciences International. – 2002. – Vol. 41, No. 4. – P. 568-578.
19. *Luchinin V., Afanasjev A., Ilyin V., Korlyakov A., Petrov A.* Family of micro switches based on silicon carbide for extreme conditions and duty // В сб.: Proceedings of the 2017 11th International Workshop on the Electromagnetic Compatibility of Integrated Circuits, EMCCompo 2017 11. – 2017. – P. 97-99.
20. *Глушко А.А.* Приборно-технологическое моделирование в системе TCAD Sentaurus. Методические указания к выполнению лабораторных работ по дисциплине "Автоматизация проектирования электронных средств". – М., 2015.

REFERENCES

1. *Bourgeois J., Goldstein S.* Distributed intelligent MEMS: progresses and perspectives. ICT Innovations 2011. Vol. 150. Advances in Intelligent and Soft Computing. Springer Berlin, Heidelberg, 2012, pp. 15-25.
2. *Shakhnov V.A., Zinchenko L.A., Kosolapov I.A.* Simulation of distributed MOEMS for smart environments, *Proc. 10th International Conference on Advanced Semiconductor Devices and Microsystems, ASDAM 2014*, 2014.
3. *Shakhnov V., Zinchenko L., Kosolapov I., Filippov I.* Modeling and Optimization of Radiation Tolerant Microsystems, *EMS '14 Proceedings of the 2014 European Modelling Symposium*, 2014, pp. 484-489.
4. *Shakhnov V.A., Zinchenko L.A., Terekhov V.V., Mikhaylichenko S.S.* Radiatsionnaya stoykost' mikroelektromekhanicheskikh sistem [Radiation Tolerant Microsystems], *Nanoingzheneriya [Nanoengineering]*, 2015, No. 9, pp. 13-17.
5. *Mandel'brot B.* Fraktal'naya geometriya prirody [Fractal Geometry of nature]. Moscow: Institut komp'yuternykh issledovaniy, 2002, 656 p.
6. *Krön B.* Growth of self-similar graphs, *Journal of Graph Theory*, 2004, Vol. 45, No. 3, pp. 224-239.
7. *Kochkarov A.A., Kochkarov R.A.* Parallel'nyy algoritm poiska kratchayshego puti na predfraktal'nom grafe [Parallel shortest path search algorithm for predfractal graph], *Zhurnal vychislitel'noy matematiki i matematicheskoy fiziki [Computation mathematics and mathematical physics]*, 2004, No. 6, pp. 1147-1152.
8. *Petronero L., Tozatti E. (red.)* Fraktaly v fizike [Fractals in Physics]. Moscow: Mir, 1988, 672 p.
9. *Sorokin B.S.* Ispol'zovanie lemmy Lorentsa dlya rascheta mnogoluchevogo rasprostraneniya radiovoln v labirintnykh sistemakh [Application of Lorenz's lemma for multipaths radiowave propagation in urban areas], *Trudy XV Vserossiyskoy shkoly-seminara «Fizika i primenenie mikrovoln» [Proc. XV Workshop Waves]*, 2015, pp. 32-34.
10. *Kormen T., Leyzerson Ch., Rivest R., Shteyn K.* Algoritmy: postroyeniye i analiz [Introduction to Algorithms]. Vil'yams, 2006.
11. *Hart P. E., Nilsson N. J., Raphael B.* A Formal Basis for the Heuristic Determination of Minimum Cost Paths, *IEEE Transactions on Systems Science and Cybernetics*, 1968, No. 2, pp. 100-107.
12. *Dechter R, Pearl J.* Generalized best-first search strategies and the optimality of A*, *Journal of the ACM*, 1985, Vol. 32, No. 3.
13. *Geisberger R., Sanders P., Schultes D., Delling D.* Contraction Hierarchies: Faster and Simpler Hierarchical Routing in Road Networks, *LNCS*, 2008, Vol. 5038.
14. *Funke S., Storandt S.* Provable Efficiency of Contraction Hierarchies with Randomized Pre-processing, *LNCS*, 2015, Vol. 9472, pp. 479-490.
15. *Zinchenko L.A., Kosolapov I.A., SHakhnov V.A.* Osobennosti mnogomasshtabnogo modelirovaniya mikrooptoelektromekhanicheskikh sistem s uchetom tekhnologicheskikh pogreshnostey [Features of multiscale modeling of micro-opto-electromechanical systems for DFM], *Datchiki i sistemy [Sensors and systems]*, 2013, No. 9 (172), pp. 29-37.
16. *Zinchenko L.A., Sorokin B.S.* Bionicheskie informatsionnye sistemy v proektirovaniy mikro- i nanosistem [Bionic information systems in the design of micro- and nanosystems], *V sb.: KII-2008 Odinnadtsataya natsional'naya konferentsiya po iskusstvennomu intellektu s mezhdunarodnym uchastiem: trudy konferentsii. Rossiyskaya assotsiatsiya iskusstvennogo intellekta [Proc. KII-2008 Eleventh National Conference on Artificial Intelligence with International Participation. Russian Association of Artificial Intelligence]*, 2008, pp. 17-25.
17. *Bova V.V., Kureichik V.V., Lezhebokov A.A.* The integrated model of representation of problem-oriented knowledge in information systems, *8th IEEE International Conference on Application of Information and Communication Technologies, AICT 2014 - Conference Proceedings 8*, 2014, pp. 7035923.
18. *Kureichik V.V., Kureichik V.M.* A fractal algorithm for graph partitioning, *Journal of Computer and Systems Sciences International*, 2002, Vol. 41, No. 4, pp. 568-578.
19. *Luchinin V., Afanasjev A., Ilyin V., Korlyakov A., Petrov A.* Family of micro switches based on silicon carbide for extreme conditions and duty, *Proceedings of the 2017 11th International Workshop on the Electromagnetic Compatibility of Integrated Circuits, EMCCompo 2017 11*, 2017, pp. 97-99.

20. *Glushko A.A. Priborno-tehnologicheskoe modelirovanie v sisteme TCAD Sentaurus. Metodicheskie ukazaniya k vypolneniyu laboratornykh rabot po distsipline "Avtomatizatsiya proektirovaniya elektronnykh sredstv" [Instrument-technological modeling in TCAD Sentaurus system. Methodical instructions to performance of laboratory works on discipline "automation of design of electronic means"]*. Moscow, 2015.

Статью рекомендовал к опубликованию д.т.н., профессор И.Б. Фоминых.

Зинченко Людмила Анатольевна – МГТУ им. Н.Э. Баумана; e-mail: lzinchenko@bmstu.ru; 105005, г. Москва, ул. 2-я Бауманская, 5; тел. 84992636553; д.т.н.; профессор.

Никитин Илья Владимирович – студент

Верстов Владимир Александрович – e-mail: v.verstov@gmail.com; ассистент.

Гусев Михаил Владимирович – студент.

Бачурин Антон Сергеевич – студент.

Дмитриев Виктор Евгеньевич – студент.

Сорокин Борис Сергеевич – МГУ им. М.В. Ломоносова; e-mail: me@mralin.ru; 119991, г. Москва, Ленинские горы, 1; аспирант.

Zinchenko Lyudmila Anatol'evna – Bauman Moscow State Technical University; e-mail: lzinchenko@bmstu.ru; 105005, Moscow, 2nd Baumanskaya, 5; phone: +74992636553; dr. of eng. sc.; professor.

Nikitin Ilya Vladimirovich – student.

Verstov Vladimir Alexandrovich – e-mail: v.verstov@gmail.com; assistant.

Gusev Michail Vladimirovich – student.

Bachurin Anton Sergeevich – student.

Dmitriev Victor Evgen'evich – student.

Sorokin Boris Sergeevich – Moscow State University, e-mail: me@mralin.ru; 119991, Leninskie Gory, 1, Moscow; postgraduate student.

УДК 004.021

DOI 10.23683/2311-3103-2018-4-58-66

В.М. Глушань, А.В. Зубрицкий

АЛГОРИТМ РАЗБИЕНИЯ МНОЖЕСТВА ПО ЕГО НОМЕРУ НА СОВОКУПНОСТЬ РАВНОМОЩНЫХ ПОДМНОЖЕСТВ*

Представлены результаты разработки алгоритма формирования (r, n) -разбиений по их номерам в лексикографической последовательности всех возможных комбинаций. На сегодняшний день существуют различные задачи комбинаторной оптимизации, для которых комбинаторное соединение типа (r, n) -разбиения может являться оптимальным решением. Теоретически оптимальное решение можно найти полным перебором всех (r, n) -разбиений, но количество возможных вариантов экспоненциально растет с увеличением числа элементов. Следовательно, для задач большой размерности необходимо сократить время вычислений, то есть сократить число возможных решений или организовать параллельный вычислительный процесс. В статье приведено обоснование формулы, позволяющей вычислить точное количество всех возможных неповторяющихся (r, n) -разбиений. Опираясь на данную формулу, предложен алгоритм построения разбиений по заданному номеру. Он позволяет проанализировать область решений и использовать методы случай-

* Работа выполнена при поддержке РФФИ (проект № 18-01-00041).