

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 004.896

DOI: 10.17587/it.26.667-672

В. В. Курейчик, д-р техн. наук, проф., зав. кафедрой САПР, e-mail: vkur@sfnu.ru,
Южный федеральный университет
Вл. Вл. Курейчик, канд. техн. наук, ведущий инженер, e-mail: kureichik@yandex.ru,
ООО "Газпром подземремонт Уренгой"

Разбиение графов на части на основе комбинированного подхода¹

Рассмотрена одна из важных комбинаторных задач оптимизации — задача разбиения графов на части. Она относится к классу NP-трудных задач оптимизации. Приведена постановка задачи разбиения графов на части. В связи со сложностью данной задачи для ее решения предлагается новая стратегия поиска, основанная на комбинированном подходе, который заключается в разделении процесса поиска решений на два уровня. На первом уровне для быстрого получения подобластей с высоким значением целевой функции применяется метод пчелиной оптимизации, а на втором уровне для улучшения полученных решений используется эволюционный алгоритм. Для реализации данного подхода авторами разработан комбинированный алгоритм, позволяющий получать наборы квазиоптимальных решений за полиномиальное время и в то же время избегать заикливания в локальных областях. Разработан программный модуль и реализованы на ЭВМ алгоритмы разбиения графов на части. Проведен вычислительный эксперимент при разбиении на восемь частей тестовых схем (бенчмарков) фирмы IBM.

Анализ экспериментальных исследований показал, что разработанный комбинированный алгоритм в среднем на 5 % превосходит результаты разбиения, полученные с использованием известных алгоритмов hMetis, PGASComplex при сопоставимом времени решения, что говорит об эффективности предложенного подхода. Временная сложность разработанного комбинированного алгоритма ориентировочно составляет $O(n^2)$.

Ключевые слова: разбиение графов на части, комбинированный подход, комбинированный алгоритм, метод пчелиной оптимизации, эволюционный алгоритм

Введение

Основателем теории графов является математик Леонард Эйлер, решивший в 1736 г. известную в то время задачу о кенигсбергских мостах. В настоящее время теория графов находит широкое применение в компьютерных науках и информационных технологиях, а также является одним из самых востребованных инструментов при создании искусственного интеллекта. Сейчас достижения, полученные при решении многих задач из теории графов, используются в экономике, социологии, биохимии, психологии, инженерных науках и т. д.

Одной из основных комбинаторно-логических оптимизационных задач является разбиение

графа на заданное или произвольное число частей [1, 2]. Задача разбиения графа на части имеет много практических применений. Она используется при проектировании устройств автоматики и вычислительной техники, при создании систем управления, компьютерных и инженерных сетей, а также при решении различных задач искусственного интеллекта. Отметим, что разработка различных методов и алгоритмов ее решения осуществляется на протяжении ряда лет, являясь, по-прежнему, актуальной проблемой. Это связано, в первую очередь, с тем, что задача разбиения графа относится к классу NP-трудных проблем оптимизации, и не существуют оптимальные алгоритмы ее решения за полиномиальное время [3]. В связи с этим появляется необходимость в разработке новых подходов, методов и алгоритмов решения данного класса задач. Одним из таких подходов является раз-

¹Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 19-01-00059.

работка новых и модифицированных методов, использующих стратегии эволюционного моделирования и принципы природных механизмов принятия решений, а также их комбинирование и гибридизация [4—11].

1. Постановка задачи разбиения графов

Оптимизационная задача разбиения графа на части состоит в отыскании такого разбиения из множества возможных разбиений некоторого графа, при котором минимизируется число внешних ребер графа между частями разбиения или максимизируется число внутренних ребер в частях разбиения, и при этом учитываются все поставленные в задаче ограничения. Разбиение графа (гиперграфа) на части относится к задачам дискретной условной оптимизации из-за прерывности ее целевой функции и наличия множества ограничений на переменные. Поэтому, ее выделяют в особый класс комбинаторных задач.

Постановка задачи разбиения графа $G = (X, U)$ на части $G_i = (X_i, U_i)$, $X_i \subseteq X$, $U_i \subseteq U$, $i \in I = \{1, 2, \dots, l\}$, где l — число частей, на которое разбивается граф, широко представлена в литературе [12, 13]. Другими словами, совокупность частей $S(G) = \{G_1, G_2, G_i, \dots, G_l\}$ является разбиением графа G , если любая часть из этой совокупности не пустая, если для любых двух частей из $S(G)$ пересечение множества вершин является пустым, пересечение множества ребер при этом может быть не пустым (подразумеваются пересечения по тем ребрам, которые инцидентны вершинам из разных частей разбиения), а также если объединение всех частей l равно графу G . В работе [12] введено обозначение $|U_{i,j}| = K_{i,j}$ — число соединительных ребер частей G_i и G_j графа G . Число соединительных ребер всех частей графа записывается в виде

$$K = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n K_{i,j}, \quad i \neq j. \quad (1)$$

Задача разбиения графа $G = (X, U)$ заключается в нахождении такой совокупности частей, чтобы число соединительных ребер графа G удовлетворяло заданному критерию оптимальности. Обычно за критерий разбиения принимают величину K (1). Тогда целью оптимизации является условие $K \rightarrow \min$.

2. Комбинированный подход для решения задачи разбиения графов

Задача разбиения графа на части является NP-трудной задачей комбинаторной оптимизации, т.е. не существуют методы ее оптимального решения за полиномиальное время (сложность и размерность данной задачи очень велика, что не позволяет ее решать за приемлемое время на современных ЭВМ). В связи с этим возникает проблема практической разрешимости этой задачи, т.е. нахождения практически реализуемого и эффективного метода ее решения. В настоящее время выделяют два подхода к решению данных задач. Первый подход — это упрощение алгоритмов, т.е. снижение их вычислительной сложности. Второй подход — это упрощение решаемых задач за счет уменьшения их размерности или их декомпозиции [14].

На основе этих двух подходов в работе предлагается стратегия поиска эффективных решений в задаче разбиения графа на части. Это последовательная стратегия "биоинспирированный поиск—эволюция" (рис. 1).

Данная стратегия позволяет реализовывать различные методы на разных уровнях поиска: на первом уровне сокращать размерность поиска решаемой задачи, а на втором уровне выполнять эффективную оптимизацию на основе эволюционных процедур поиска.

Для реализации предложенной последовательной стратегии авторы предлагают комбинированную архитектуру поиска, основанную на методах эволюционного моделирования и роевого интеллекта, приведенную на рис. 2 [10, 13, 15, 16].

Опишем работу каждого блока архитектуры более подробно. На вход блока биоинспирированного поиска (БП) поступают исходные данные задачи разбиения, такие как: число вершин, число ребер, число блоков разбиения. В каче-

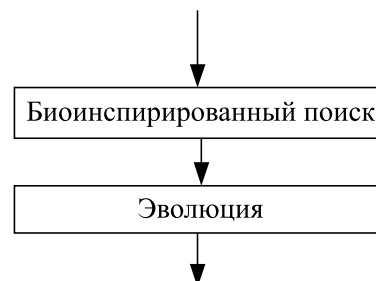


Рис. 1. Последовательная стратегия "биоинспирированный поиск-эволюция"

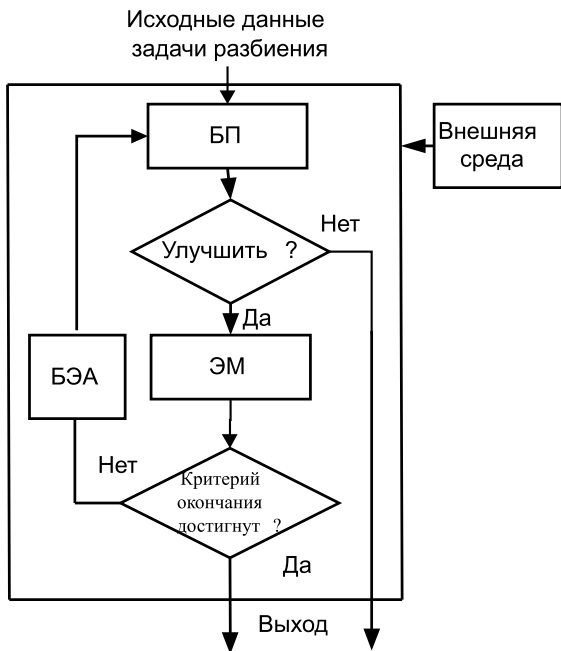


Рис. 2. Комбинированная архитектура поиска

стве БП применяется модифицированный роевой метод пчелиной оптимизации (МПО). Данный метод позволяет достаточно быстро разбивать поисковое пространство на подобласти с высоким значением целевой функции (ЦФ), что существенно сокращает время его работы. Далее проверяется условие, необходимо ли улучшение ЦФ. Если улучшение не требуется, то следует выход, если требуется, то результаты после выполнения МПО передаются в блок эволюционного моделирования (ЭМ). Здесь реализуется быстрый модифицированный эволюционный алгоритм. Данный алгоритм основан на выполнении только одного оператора — мутации и его различных модификаций.

Данный метод позволяет достаточно быстро получать наборы квазиоптимальных решений за полиномиальное время. Еще одной модификацией является введение в комбинированную архитектуру блока эволюционной адаптации (БЭА). Данный блок работает как фильтр. Он позволяет избавляться от решений с низким значением ЦФ. Заметим, что значение данной границы целевой функции будет динамически изменяться на каждой последующей итерации поиска, причем значение этой величины зависит от качества предыдущих решений. Это позволяет ускорить процесс получения результата за счет обработки только областей с высоким значением ЦФ.

3. Комбинированный алгоритм

На основе приведенной и описанной стратегии и архитектуры поиска авторы предлагают укрупненную схему комбинированного алгоритма разбиения графа на части (рис. 3).

Рассмотрим приведенную структурную схему и опишем назначение каждого ее блока более подробно.

В данной схеме процесс поиска реализуется на двух уровнях. Сначала на предварительном этапе осуществляется ввод исходных данных решаемой задачи, таких как графовая модель, число вершин, число связей и число частей, на которое необходимо разбить граф. Далее для решения задачи разбиения графа на части на первом уровне применяется метод пчелиной

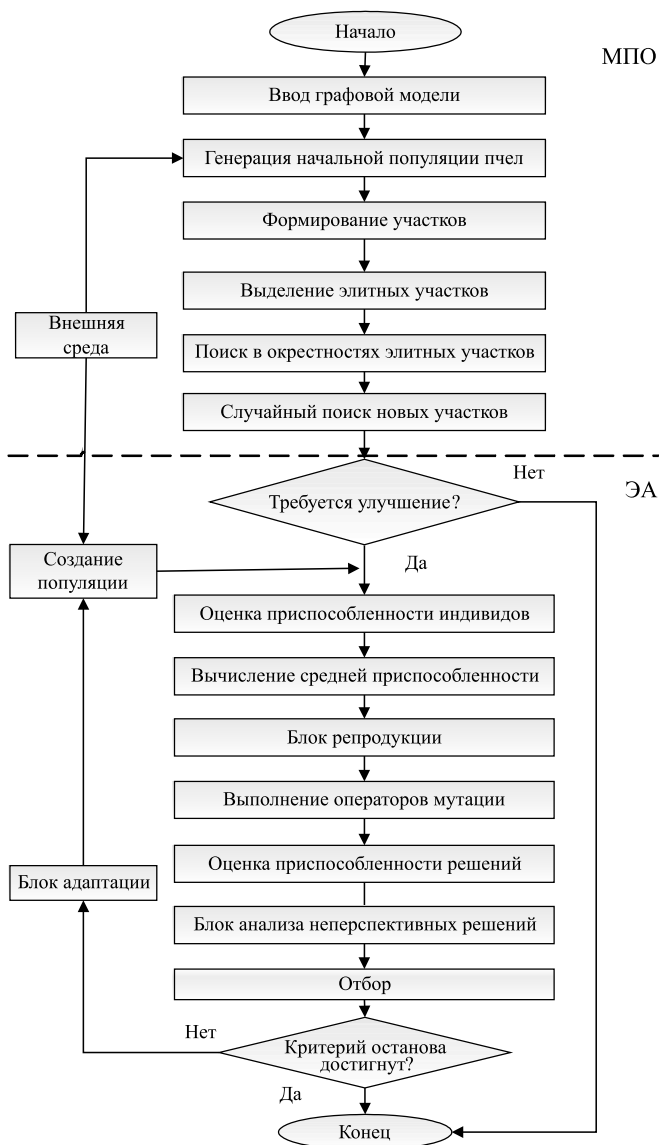


Рис. 3. Укрупненная схема комбинированного алгоритма

оптимизации (МПО) [17, 18]. Здесь сначала на основе известных принципов [4, 5] генерируется начальная популяция пчел. Далее формируются подобласти поиска (элитные участки) с высоким значением ЦФ. После этого проводится анализ элитных участков и их окрестностей, что позволяет увеличить эффективность поиска. Затем реализуется случайный поиск новых областей, который дает возможность избежать попадания алгоритма в локальные оптимумы. И наконец, выполняется оценка полученных решений, если требуется улучшение, то следует переход на второй уровень поиска и выполнение быстрого эволюционного алгоритма, в противном случае завершение поиска.

На втором уровне поиска осуществляется ввод параметров эволюционного алгоритма, а именно: применяемая модель эволюции и селекции; виды и вероятности операторов мутации; выбор критерия останова поиска. Затем на основе известных принципов [4, 5] создается начальная популяция альтернативных решений решаемой задачи разбиения. Данная популяция эволюционирует с учетом модели Ж. Б. Ламарка [19]. Далее выбирается ЦФ, и на ее основе проводится оценка начальной популяции. На следующем этапе реализуется оператор репродукции (селекции). После этого формируются новые решения за счет выполнения различных операторов мутации, и проводится их оценка [4, 5]. Данные, полученные после реализации операторов и их оценок, передаются в блок анализа неперспективных решений. Здесь каждому решению присваивается определенная метка (перспективное, неперспективное, тривиальное и др.). Заметим, что такое ранжирование решений за счет структуризации и проведения операции отбора позволяет учитывать все имеющиеся решения при проведении эволюционного поиска и повышает эффективность работы комбинированного алгоритма в целом. Затем выполняется оценка достижения критерия останова, и если он не достигнут, то далее результаты передаются в БЭА. Данный блок оказывает непосредственное влияние на процесс перестройки текущей популяции и создания на ее основе новой популяции альтернативных решений. Блок внешней среды позволяет проводить выбор используемой на данном этапе модели эволюции и управлять всеми изменяемыми параметрами поиска. Отметим, что поиск продолжается итерационно до получения набора квазиоптимальных решений решаемой задачи.

Процесс работы заканчивается при достижении определенного критерия окончания работы алгоритма. Таким критерием может быть время, число заданных генераций или получение набора квазиоптимальных решений.

4. Вычислительный эксперимент

Чтобы продемонстрировать эффективность, а также вычислительные характеристики предложенного алгоритма, создана программная среда с использованием языка программирования C#. Был проведен вычислительный эксперимент при разбиении тестовых схем IBM [20] на восемь частей известными алгоритмами hMetis [21], PGACComplex и разработанным авторами комбинированным алгоритмом (КА). Приведем зависимости качества и времени работы данных алгоритмов от разбиваемых на восемь частей тестовых схем. Представим полученные результаты вычислительного эксперимента в виде таблицы и рис. 4, 5 (см. вторую сторону обложки).

В результате анализа представленной таблицы и гистограмм можно сделать вывод, что разработанный комбинированный алгоритм

Сравнение результатов разбиения тестовых схем IBM алгоритмами hMetis, PGACComplex и разработанным авторами комбинированным алгоритмом (КА)

Схема	hMetis		PGACComplex		КА	
	<i>K</i> , ед.	<i>t</i> , с	<i>K</i> , ед.	<i>t</i> , с	<i>K</i> , ед.	<i>t</i> , с
ibm01	598	124	621	127	608	125
ibm02	823	249	820	266	815	258
ibm03	1947	472	1853	487	1820	481
ibm04	2446	872	2409	915	2390	894
ibm05	4405	981	4386	1023	4327	993
ibm06	1859	1096	1817	1133	1797	1105
ibm07	2930	1198	2893	1214	2842	1201
ibm08	3306	1219	3272	1237	3207	1225
ibm09	2277	1411	2205	1434	2185	1427
ibm10	3108	1810	3064	1898	3004	1865
ibm11	3001	2315	2935	2380	2895	2365
ibm12	4966	2287	4901	2312	4876	2301

Здесь *K* — целевая функция (число ребер графа, попавших в разрез), *t* — время

в среднем на 5 % превосходит результаты разбиения, полученные с использованием известных алгоритмов hMetis, PGASComplex при сопоставимом времени решения, что говорит об эффективности предложенного подхода.

Заключение

В работе описана одна из важных задач комбинаторной оптимизации — задача разбиения графов на части. Для ее эффективного решения предложена новая стратегия, основанная на комбинированном подходе. Отличительной особенностью данного подхода является разделение поиска решений на два уровня. На первом уровне на основе метода пчелиной оптимизации выделяются подобласти с высоким значением целевой функции, а на втором проводится улучшение полученных решений на основе эволюционного алгоритма. Для реализации этого подхода разработан комбинированный алгоритм, позволяющий получать наборы квазиоптимальных решений за полиномиальное время и избегать заикливания в локальных областях. Разработан программный модуль на языке C#. Проведен вычислительный эксперимент на тестовых примерах (бенчмарках) фирмы IBM. Проведенные экспериментальные исследования показали преимущество использования разработанного комбинированного подхода для решения задач разбиения графов на части большой размерности по сравнению с известными методами.

Качество разбиения, полученное, на основе разработанного комбинированного алгоритма, в среднем на 5 % превосходит результаты разбиения, полученные с использованием известных алгоритмов hMetis, PGASComplex при сопоставимом времени решения, что говорит об эффективности предложенного подхода. Временная сложность разработанного комбинированного алгоритма ориентировочно составляет $O(n^2)$.

Список литературы

1. Мелихов А. Н., Берштейн Л. С., Курейчик В. М. Применение графов для проектирования дискретных устройств: учебник. М.: Наука, 1974. 304 с.
2. Курицкий Б. Я. Оптимизация вокруг нас: учебник. СПб.: Машиностроение, 1989. 145 с.
3. Кормен Т., Лейзерсон И., Ривест Р. Алгоритмы: построения и анализ. М.: МЦМО, 2000. 960 с.

4. Holland John H. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Application to Biology, Control, and Artificial Intelligence. USA: University of Michigan, 1975. 183 p.

5. De Jong K. Evolutionary Computation: Recent Development and Open Issues // Proceedings 1st International conf., Evolutionary Computation and Its Application. Moscow, 1996. P. 7—18.

6. Карпенко А. П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: Учеб. пособ. М.: Издательство МГТУ им. Н. Э. Баумана, 2014. 446 с.

7. Abraham A., Ramos V., Grosan G. Swarm Intelligence in Data Mining. Berlin. Heidelberg: Springer Verlag, 2007. P. 100—113.

8. Hassanien E., Emary E. Swarm Intelligence. Principles Advances and Applications. CRC Press, 2015. 228 p.

9. Родзин С. И., Курейчик В. В. Состояние, проблемы и перспективы развития биоэвристик // Программные системы и вычислительные методы. 2016. № 2. С. 158—172.

10. Курейчик В. В., Курейчик Вл.Вл., Бова В. В. Комбинированный поиск при проектировании // Образовательные ресурсы и технологии. 2014. № 2 (5). С. 90—94.

11. Курейчик В. В., Курейчик Вл. Вл. Биоинспирированный поиск при проектировании и управлении // Известия ЮФУ. Технические науки. 2012. № 11 (136). С. 178—183.

12. Курейчик В. М., Курейчик В. В. Генетический алгоритм разбиения графа // Известия АН. Теория и системы управления 1999. № 5. С. 79—87.

13. Kureichik V., Zaruba D., Kureichik VI. Hybrid approach for graph partitioning // Advances in Intelligent Systems and Computing. 2017. 573. P. 64—73.

14. Базилович Р. П. Декомпозиционные и топологические методы автоматизированного конструирования электронных устройств: монография. Львов: Вишашкола, 1981. 81 с.

15. Hendrickson B., Leland R. A Multilevel Algorithm for Partitioning Graphs // Proceedings of the 1995 ACM/IEEE conference on Super computing. P. 626-657.

16. Kureichik V., Kureichik VI., Bova V. Placement of VLSI fragments based on a multilayered approach // Advances in Intelligent Systems and Computing. 464. 2016. P. 181—190.

17. Karaboga D. An idea based on honey bee swarm for numerical optimization // Erciyes University, Engineering Faculty, Computer Engineering Department. 2005. 110 p.

18. Курейчик В. В., Запорожец Д. Ю. Роевой алгоритм в задачах оптимизации // Известия ЮФУ. Технические науки. 2010. № 7 (108). С. 28—32.

19. Курейчик В. В., Курейчик В. М., Сороколетов П. В. Анализ и обзор моделей эволюции // Известия Российской академии наук. Теория и системы управления. 2007. № 5. С. 114—126.

20. Alpert C. J. The ISPD-98 Circuit Benchmark Suite. // In Proc. ACM/IEEE International Symposium on Physical Design, April 1998. P. 80—85.

21. Karypis G., Kumar V. METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill—Reducing Orderings of Sparse Matrices Version 5.1.0: Department of Computer Science and Engineering University of Minnesota Minneapolis, MN. 2013.

V. V. Kureichik, Dr. Tech. Sci., Professor, Head of CAD Department, e-mail: vkur@sfedu.ru,
Southern Federal University, Rostov-on-don, 344006, Russian Federation,
VI. VI. Kureichik, Ph.D., Principal Engineer, e-mail: kureichik@yandex.ru,
"Gazprom subterranean repair Urengoy" LLC, Saint-Petersburg, 190000, Russian Federation

Graph Partitioning Based on the Combined Approach

The article considers one of the most important combinatorial optimization problems — the problem of graph partitioning. It belongs to the class of NP-complex optimization problems. The article presents the partitioning problem statement. Due to the complexity of this task, the article proposes a new search strategy based on a combined approach. The combined approach is to divide the decision-making process into two levels. At the first level, the bee optimization method is used to quickly obtain subdomains with a high value of the objective function, and at the second level, an evolutionary algorithm is used to improve obtained solutions. To implement this approach, the authors developed a combined algorithm that can obtain sets of quasi-optimal solutions in polynomial time and avoid looping in local regions at the same time. A software module is developed and algorithms for partitioning graphs into parts are implemented. A computational experiment has been carried out when dividing into 8 parts of test circuits (benchmarks) by IBM. An analysis of experimental studies showed that the developed combined algorithm is on average 5 % higher than the partition results obtained by well-known hMetis, PGACOMPLEX algorithms with comparable solution time, which indicates the effectiveness of the proposed approach. The time complexity of the developed combined algorithm is approximately $O(n^2)$.

Keywords: graph partitioning, combined approach, combined algorithm, bee colony optimization, evolutionary algorithm

DOI: 10.17587/it.26.667-672

References

1. Melikhov A. N., Bershtein L. S., Kureychik V. M. Usus graphs enim consilium discreta cogitationes: textbook, Moscow, Nauka, 1974, 304 p. (in Russian).
2. Kuritzky B. Y. Optimization around us: textbook, SPb., Mechanics engineering, 1989, 145 p. (in Russian).
3. Kormen T., Leiserson I., Rivest R. L. Algorithms: construction and analysis, Moscow, MCMO, 2000, 960 p. (in Russian).
4. Holland John H. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Application to Biology, Control, and Artificial Intelligence, USA, University of Michigan, 1975, 183 p.
5. De Jong K. Evolutionary Computation: Recent Development and Open Issues, *Proceedings 1st International conf., Evolutionary Computation and Its Application*, Moscow, 1996, pp. 7–18.
6. Karpenko F. P. Modern search engine optimization algorithms. Algorithms inspired by nature: textbook, Moscow, Publishing house of MGTU im. N. Je. Bauman, 2014, 446 p. (in Russian).
7. Abraham A., Ramos V., Grosan G. Swarm Intelligence in Data Mining. Berlin, Heidelberg, Springer Verlag, 2007, pp. 100–113.
8. Hassanien E., Emary E. Swarm Intelligence. Principles Advances, and Applications, CRC Press, 2015, 228 p.
9. Rodzin S. I., Kurejchik V. V. State problems and prospects of development of bio-heuristics, *Programmnye Sistemy i Vychislitel'nye Metody*, 2016, no. 2, pp. 158–172 (in Russian).
10. Kurejchik V. V., Kurejchik VI. VI., Bova V. V. Combined search in design, *Educational Resources et Technology*, 2014, no. 2 (5), pp. 90–94 (in Russian).
11. Kurejchik V. V., Kurejchik VI. VI. Bioinspiratum quærer in consilio et administratione, *Izvestija JuFU. Tehnicheskie nauki*, 2012, no. 11 (136), pp. 178–183.
12. Курейчик В. М., Курейчик В. В. Genetic algorithm for graph splitting, *Fama Academiae Scientiarum, Theoria, et imperium ratio*, 1999, no. 5, pp. 79–87.
13. Kureichik V., Zaruba D., Kureichik VI. Hybrid approach for graph partitioning, *Advances in Intelligent Systems and Computing*, 2017, no. 573, pp. 64–73.
14. Bazelevich R. P. Decomposition and topological methods of automated design of electronic devices, Lviv, Widescale, 1981, 81 p. (in Russian).
15. Hendrickson B., Leland R. A Multilevel Algorithm for Partitioning Graphs, *Proceedings of the 1995 ACM/IEEE conference on Super computing*, pp. 626–657.
16. Kureichik V., Kureichik VI., Bova V. Placement of VLSI fragments based on a multilayered approach, *Advances in Intelligent Systems and Computing*, 2016, no. 464, pp. 181–190.
17. Karaboga D. An idea based on honey bee swarm for numerical optimization, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005, 110 p.
18. Kureichik V., Zaporozhets D. Yu. Swarming algorithm in optimization problems, *Izvestija JuFU. Tehnicheskie nauki*, 2010, no. № 7 (108), pp. 28–32.
19. Kureychik V. V., Kureychik V. M., Sorokoletov P. V. Analysis and review of evolution models, *Fama Academiae Scientiarum, Theoria, et imperium ratio*, 2007, no. 5, pp. 114–126 (in Russian).
20. Alpert C. J. The ISPD-98 Circuit Benchmark Suit, *In Proc. ACM/IEEE International Symposium on Physical Design*, April 1998, pp. 80–85.
21. Karypis G., Kumar V. METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill—Reducing Orderings of Sparse Matrices Version 5.1.0: Department of Computer Science and Engineering University of Minnesota Minneapolis, MN. 2013.

Рисунки к статье В. В. Курейчика, Вл. Вл. Курейчика

«РАЗБИЕНИЕ ГРАФОВ НА ЧАСТИ НА ОСНОВЕ КОМБИНИРОВАННОГО ПОДХОДА»

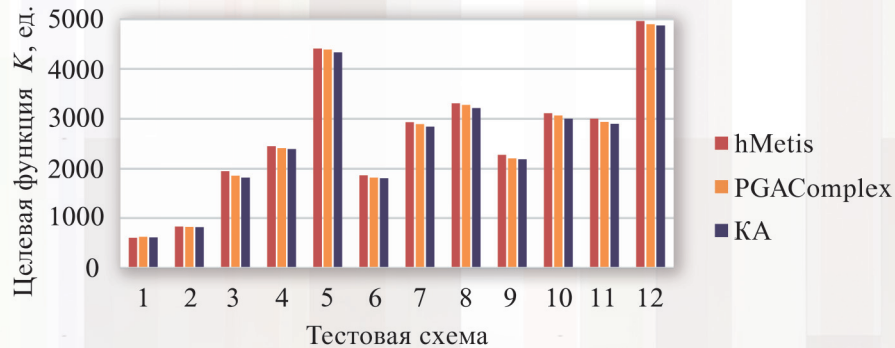


Рис. 4. Гистограмма сравнения значений целевой функции алгоритмов разбиения в зависимости от тестовых схем IBM

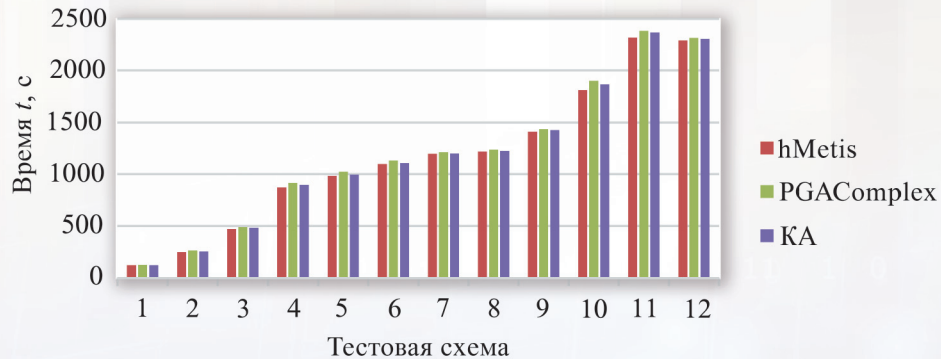


Рис. 5. Гистограмма сравнения времени работы алгоритмов разбиения в зависимости от тестовых схем IBM