

13. *Kureychik V.M.* Gibridnye geneticheskie algoritmy [Hybrid genetic algorithms], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2007, No. 2 (77), pp. 5-12.
14. *Zhabin D.A., Garays D.V., Kalentyev A.A., Dobush I.M. and Babak L.I.* Automated Synthesis of Low Noise Amplifiers Using S-parameter Sets of Passive Elements, Asia-Pacific Microwave Conference (APMC 2017), Kuala Lumpur, Malaysia, 2017 (accepted for publication).
15. *Kalentyev A.A., Garays D.V. and Babak L.I.* Genetic-Algorithm-Based Synthesis of Low-Noise Amplifiers with Automatic Selection of Active Elements and DC Biases, European Microwave Week 2014, Rome, Italy. October 2014, pp. 520-523,
16. *Babak L.I., Kokolov A.A. and Kalentyev A.A.* A New Genetic-Algorithm-Based Technique for Low Noise Amplifier Synthesis, European Microwave Week 2012, Amsterdam, The Netherlands, November 2012, pp. 520-523.
17. *Mann G.K.I., Gosine R.G.* Three-dimensional min-max-gravity based fuzzy PID inference analysis and tuning, *Fuzzy Sets and Systems*, 2005, Vol. 156, pp. 300-323.
18. *Golitsyn G.A., Petrov V.M.* Informatsiya i biologicheskie printsipy optimal'nosti: Garmoniya i algebra zhivogo [Information and biological principles of optimality: Harmony and algebra of the living]. Moscow: KomKniga, 2005.
19. *Kolesov Yu.B., Senichenkov Yu.B.* Modelirovanie sistem. Dinamicheskie i gibridnye sistemy: ucheb.-metod, posobie [System modeling. Dynamic and hybrid systems: a training manual]. Saint Petersburg: BKhV-Peterburg, 2006.
20. *Abraham A., Grosan G., Ramos V.* Swarm Intelligence in Data Mining. Berlin. Heidelberg: SpringerVerlag, 2007, 267 p.

Статью рекомендовал к опубликованию к.т.н., доцент Ю.П. Волощенко.

Данильченко Евгения Владимировна – Южный Федеральный университет; e-mail: lipkina@sfedu.ru; г. Таганрог, Россия; тел.: +79525691761; кафедра САПР.

Данильченко Владислав Иванович – e-mail: vdanielchenko@sfedu.ru; тел.: +79526088561; кафедра САПР.

Курейчик Виктор Михайлович – e-mail: vmkureychik@sfedu.ru; тел.: +79282132730; кафедра САПР; д.т.н.; профессор.

Danilchenko Yevgenia Vladimirovna – Southern Federal University; e-mail: lipkina@sfedu.ru; Taganrog, Russia; phone: +79525691761; the department of computer-aided design.

Danilchenko Vladislav Ivanovich – e-mail: vdanielchenko@sfedu.ru; phone: +79526088561; the department of computer-aided design.

Kureichik Viktor Mikhailovich – e-mail: vmkureychik@sfedu.ru; phone: +78634311487; the department of computer-aided design; dr. of eng. sc.; professor.

УДК 004.896

DOI 10.18522/2311-3103-2021-2-39-50

В.В. Курейчик, Вл.Вл. Курейчик

ПРОГРАММНАЯ ПОДСИСТЕМА ДЛЯ РЕШЕНИЯ NP-СЛОЖНЫХ КОМБИНАТОРНО-ЛОГИЧЕСКИХ ЗАДАЧ НА ГРАФАХ*

Работа посвящена созданию программной подсистемы для решения NP-трудных и NP-сложных комбинаторно-логических задач на графах. В статье приведено описание комбинаторно-логических задач на графах. Для эффективного их решения предлагаются новые многоуровневые архитектуры поиска, такие как простая комбинированная, параллельная комбинированная, двухуровневая, интегрированная и гибридная. Данные архитектуры основаны на методах, инспирированных природными системами. Ключевым отличии

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

ем данных архитектур является разделение поиска на два или три уровня и применение на них различных алгоритмов эволюционного моделирования и биоинспирированного поиска. Это позволяет получать наборы квазиоптимальных решений выполнять параллельную обработку и частично устранять проблему преждевременной сходимости. В статье приведено подробное описание разработанной программной подсистемы и ее модулей. В качестве модулей в подсистеме имеется пять разработанных архитектур и набор разработанных алгоритмов эволюционного моделирования и биоинспирированного поиска, таких как эволюционный, генетический, пчелиный, муравьиный, светлячковый и обезьяний. Благодаря модульной структуре в подсистеме имеется возможность конструировать более 50 различных вариантов комбинаций поиска. Это позволяет использовать все достоинства методов биоинспирированной оптимизации для эффективного решения NP-сложных комбинаторно-логических задач на графах. Для подтверждения эффективности разработанной программной подсистемы был проведен вычислительный эксперимент на тестовых примерах. Проведенные серии тестов и экспериментов показали преимущество использования программного продукта для решения комбинаторно-логических задач на графах большой размерности, по сравнению с известными алгоритмами, что говорит о перспективности применения такого подхода. Временная сложность разработанных алгоритмов в лучшем случае $\approx O(n \log n)$, в худшем случае – $O(n^3)$.

Программная подсистема; комбинаторно-логические задачи на графах; многоуровневые архитектуры; эволюционное моделирование; биоинспирированный поиск.

V.V. Kureichik, VI.VI. Kureichik

SOFTWARE SUBSYSTEM FOR SOLVING NP-COMPLEX COMBINATORIAL LOGIC PROBLEMS ON GRAPHS

The paper is devoted to the development of the software for solving NP-hard and NP-hard combinatorial-logical problems on graphs. The paper contains a description of graphs combinatorial-logical problems. New multilevel search architectures such as simple combo, parallel combo, two-levels, integrated, and hybrid are proposed to effectively address them. These architectures are based on methods inspired by natural systems. The key difference between these architectures is the division of search into two or three levels and the use of various algorithms for evolutionary modeling and bioinspired search on them. This allows obtaining sets of quasi-optimal solutions to perform parallel processing and partially eliminate the problem of premature convergence. The article provides a detailed description of the developed software subsystem and its modules. As modules in the subsystem, there are five developed architectures and a set of developed algorithms for evolutionary modeling and bioinspired search, such as evolutionary, genetic, bee, ant, firefly and monkey. Thanks to its modular structure, the subsystem has the ability to design more than 50 different search combinations. This makes it possible to use all the advantages of bioinspired optimization methods for efficiently solving NP-complex combinatorial-logical problems on graphs. To confirm the effectiveness of the developed software subsystem, a computational experiment was carried out on test examples. The series of tests and experiments carried out have shown the advantage of using a software product for solving combinatorial-logical problems on graphs of large dimension, in comparison with known algorithms, which indicates the prospects of using this approach. The time complexity of the developed algorithms is $\approx O(n \log n)$ at best, and $O(n^3)$ at worst.

Program system; graph combinatorial and logic problems; multilevel architecture; evolutionary modeling; bioinspiring search.

Введение. В настоящее время теория графов является одним из самых востребованных инструментов при создании искусственного интеллекта. При решении прикладных задач искусственного интеллекта особое значение приобретают эффективные методы решения комбинаторно-логических задач на графах. Их решения с успехом используются в различных областях науки и техники. Разработка эффективных методов и алгоритмов для решения задач на графах является актуальной проблемой в связи с тем, что эти задачи являются NP-полными и NP-трудными [1]. Появление новых, более совершенных средств электронно-

вычислительной техники является причиной для модернизации и разработки новых эффективных подходов и методов при решении комбинаторно –логических задач на графах. Одним из таких подходов является многоуровневая оптимизация, основанная на методах эволюционного моделирования и биоинспирированного поиска [2–9]. Данный подход позволяет получать наборы квазиоптимальных решений за приемлемое время.

1. Описание задачи. Под комбинаторно-логическими задачами на графах понимается задача, в которой необходимо найти наилучшее решение, или, как говорят, оптимальное. Данное решение должно приниматься оптимальным на основе критерия (меры оценки исследуемого явления) или целевой функции (ЦФ) [10]. В настоящее время существует большое количество комбинаторно-логических задач на графах, и они могут иметь различную природу. Однако их постановка имеет много аналогий. Для эффективного их решения предлагаются новые многоуровневые архитектуры поиска, основанные на методах, инспирированных природными системами. Ключевым отличием данного подхода является разделение поиска на два уровня и применение на них различных биоинспирированных алгоритмов [9, 11–13]. Это позволяет выполнять параллельную обработку и частично устранять проблему их преждевременной сходимости. Для реализации предложенного подхода и разработанных алгоритмов создана программная подсистема.

2. Программная подсистема. Главной целью разработки программной подсистемы стала реализация в вычислительном комплексе построенных многоуровневых архитектур и алгоритмов, позволяющих эффективно решать NP-сложные комбинаторно-логические задачи на графах. На рис. 1. представлена укрупненная архитектура разработанной программной подсистемы, предназначенная для эффективного решения NP-сложных комбинаторно-логических задач на графах, а также для проведения вычислительных экспериментов. Здесь сначала выполняется ввод исходных данных о решаемой задаче на графах. Далее вводятся параметры разработанных биоинспирированных алгоритмов. Затем переход на следующий блок, где выбирается тот или иной вычислительный метод. Отметим, что в разработанной программной подсистеме предусмотрены две базы данных, в которых находятся построенные архитектуры и набор разработанных алгоритмов биоинспирированной оптимизации. Данные базы подключаются к блоку «Вычислительный метод». После реализации поиска в блоке статистики можно ознакомиться с результатами процесса оптимизации.

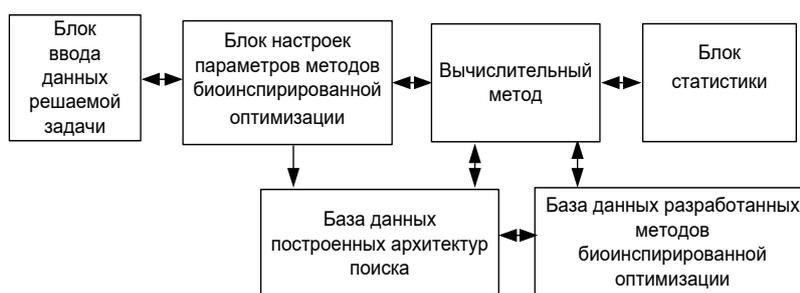


Рис. 1. Укрупненная архитектура программной подсистемы

Интерфейс главного окна программной подсистемы приведен на рис. 2.

Разработка выполнялась в среде программирования Borland C++ Builder, Visual C++. Отладка и тестирование проводилось на ЭВМ типа IBM PC с процессором core i7 с ОЗУ-8Гб. Входными параметрами для работы подсистемы являются

ся случайные графы (количество вершин и ребер), матрица смежности графа выбор схемы поиска; выбор реализуемого алгоритма, размер популяции, количество итераций, порядок и вероятности применения операторов. При этом выходными параметрами являются: время работы алгоритмов и лучшее решение, достигнутое в процессе работы.

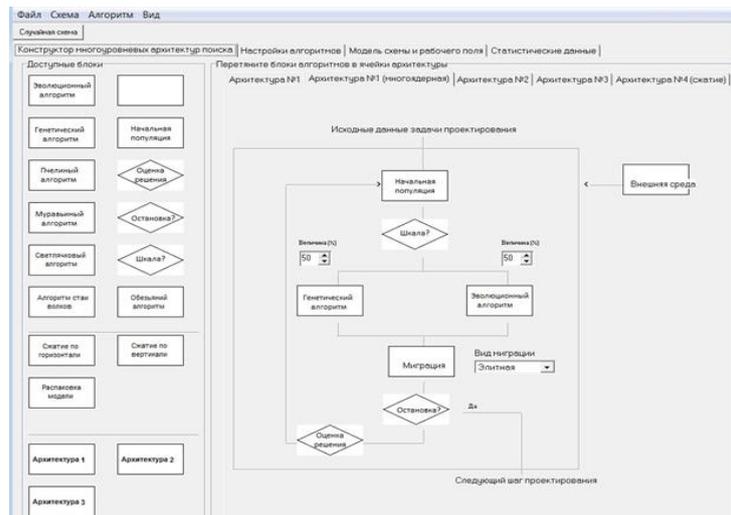


Рис. 2. Интерфейс главного окна программной подсистемы

Главной особенностью программной подсистемы является наличие конструктора архитектур, реализующие многоуровневые подходы. Многоуровневые подходы легко адаптируются к внешней среде, так как позволяют достаточно легко встраивать различные эвристики, направленные на улучшение получаемого решения, в свою структуру. Конструктор архитектур состоит из области доступных блоков и пяти разработанных типовых шаблонов. В качестве архитектуры №1 в программной подсистеме используется модифицированная комбинированная архитектура поиска, состоящая из двух вычислительных алгоритмов эволюционного и генетического [14] рис. 3., хотя можно использовать и другие методы биоинспирированного поиска.

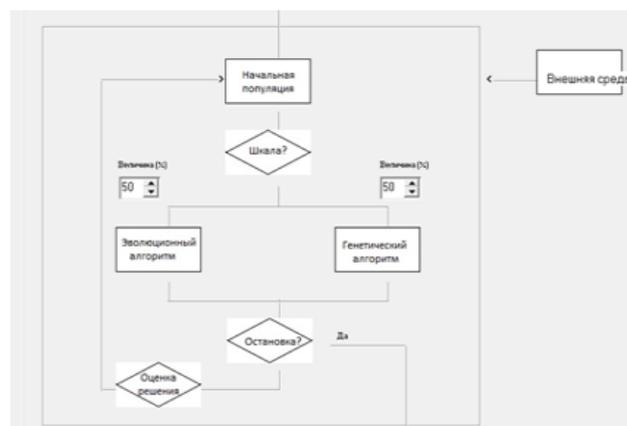


Рис. 3. Простая комбинированная архитектура

На основе данной архитектуры можно реализовать два вида поиска: последовательный и параллельный [15]. При реализации последовательного поиска реализуется только 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 приемлемое время. При получении неудовлетворительных результатов подключается блок выполнения генетического алгоритма. Генетический алгоритм реализует весь набор операторов и их модификаций. Этот процесс реализуется через блок эволюционной адаптации.

Параллельный поиск, используя многопроцессорную или многоядерную систему, позволяет одновременно и независимо друг от друга реализовать генетический и эволюционный алгоритмы. Процесс поиска ведется в течение нескольких заданных генераций. В архитектуру данного поиска вводится блок миграции рис. 4. Блок миграции позволяет производить обмен решениями, что ускоряет процесс получения эффективных решений и помогает избежать попадания в локальные оптимумы.

В качестве архитектуры №2 в программной подсистеме используется двухуровневый подход, сочетающий в себе методы биоинспирированного поиска и эволюционного моделирования. В данной архитектуре, приведенной на рис. 5. в качестве оптимизационных методов на первом уровне применяются методы биоинспирированного поиска, а на втором уровне применены - эволюционный и генетический алгоритмы, хотя можно использовать и другие методы биоинспирированной оптимизации.

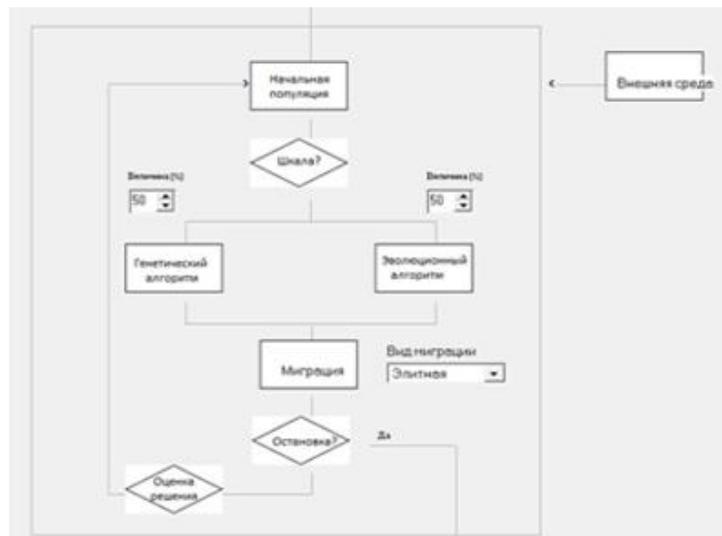


Рис. 4. Комбинированная архитектура для выполнения параллельных процедур

В качестве архитектуры №3 в программной подсистеме используется интегрированная схема поиска на основе многоуровневого подхода [16]. Такая трехуровневая архитектура, в которой реализованы четыре алгоритма биоинспирированного поиска, приведена на рис. 6.

Здесь на первом уровне используется пчелиный алгоритм [17], который быстро разбивает всю область поиска на подобласти с высоким значением ЦФ. На втором уровне реализуется муравьиный алгоритм [18], который позволяет за кратчайшее время получить список критических связей, а также определять и запоминать длину кратчайших маршрутов.

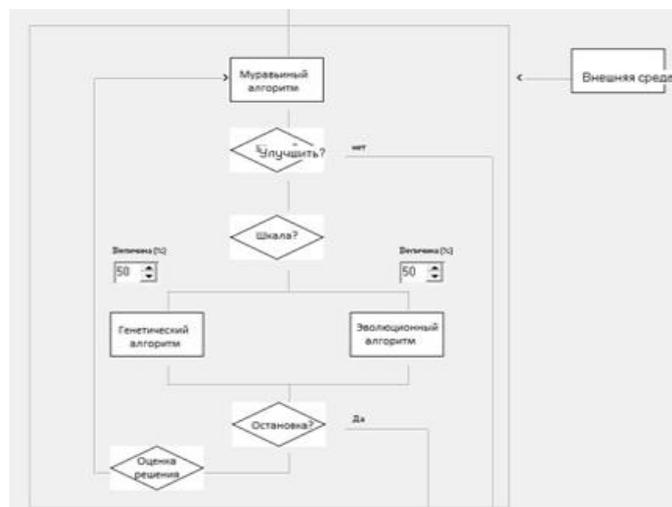


Рис. 5. Двухуровневая архитектура поиска

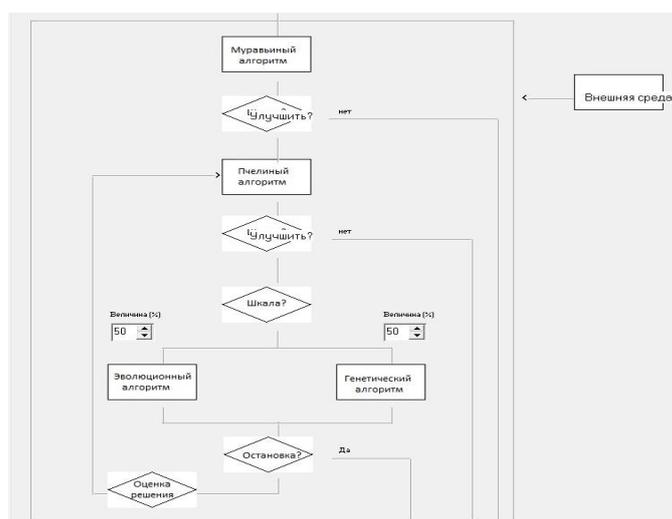


Рис. 6. Интегрированная архитектура поиска

И на третьем уровне если это необходимо на основе шкалы производится выбор того или иного метода эволюционного моделирования. При наличии многопроцессорной или многоядерной системы данные методы могут выполняться параллельно.

В качестве архитектуры №4 в программной подсистеме используется гибридная схема поиска рис. 7. Его основная идея состоит в том, что на первом уровне уменьшается размерность задачи за счет сжатия графовой модели [19], а на втором уровне выполняется оптимизация на основе методов эволюционного моделирования или биоинспирированного

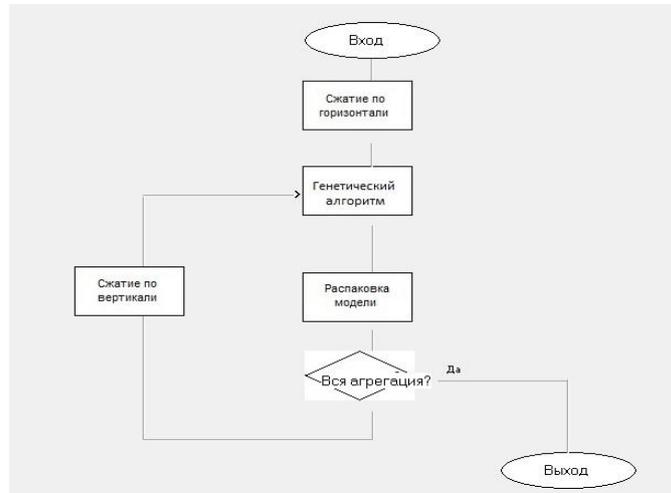


Рис. 7. Гибридная трехуровневая архитектура поиска

Заметим, что данный подход позволяет уменьшать размерность решаемой задачи, получать наборы квазиоптимальных решений за полиномиальное время.

Отметим, что разработанные в программной подсистеме архитектуры могут быть модифицированы на любое число уровней в зависимости от используемых средств вычислительной техники и временных ограничений на получение результата.

Данная программная подсистема, кроме разработанных архитектур, также поддерживает различные разработанные алгоритмы биоинспирированной оптимизации, из которых как из строительных блоков, показанных на рис. 8, собираются архитектуры различной сложности. Это эволюционный, генетический, пчелиный, муравьиный, светлячковый и обезьяний алгоритмы [3].



Рис. 8. Набор алгоритмов и строительных блоков

Далее для работы программной подсистемы проводится настройка параметров разработанных алгоритмов биоинспирированной оптимизации. Данное окно показано на рис. 9.

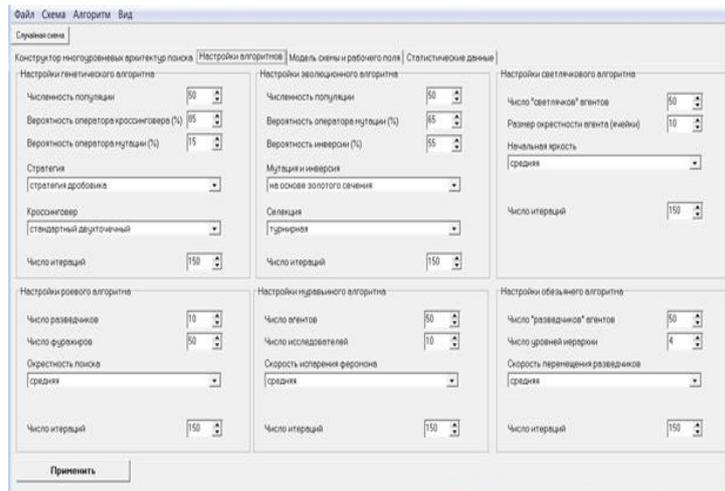


Рис. 9. Окно настройки параметров алгоритмов

Здесь производится ввод параметров алгоритмов биоинспирированной оптимизации и выполняется запуск программной системы на решение поставленной задачи. После окончания работы программной подсистемы выводится окно статистики и график изменения значения целевой функции (ЦФ), что отражает рис. 10. В данном окне показываются входные данные графовой модели и выходные результаты моделирования, такие как: значения начальной и конечной ЦФ; используемая в процессе поиска архитектура; а также время работы подсистемы в целом.

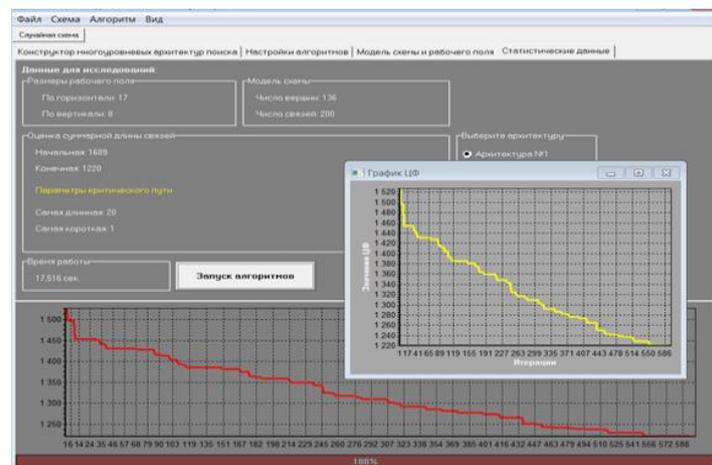


Рис. 10. Окно статистики

Заметим, что в разработанной программной подсистеме имеется возможность в строительных блоках изменять некоторые значения управляющих параметров поиска решений. Это блоки шкала выбора алгоритма, миграции и критерия остановки поиска.

Для некоторых предустановленных блоков архитектуры в типовых шаблонах предусмотрена возможность изменения значений управляющих параметров, например, таких как процент миграции начальной популяции, шкала выбора пути следования (по вероятностному критерию), настройка сходимости для определения критерия остановки или необходимости применения этапа выхода из локального оптимума.

3. Вычислительный эксперимент. Для подтверждения эффективности разработанной программной подсистемы был проведен вычислительный эксперимент на тестовых примерах [20]. Сущность тестирования заключается в запуске предложенных архитектур на одной конкретной тестовой задаче и определении значений ЦФ и времени решения. Здесь в качестве методов, реализующих тестовые примеры были выбраны известные методы QPlace [21], mPG [22] и разработанные и программно реализованные авторами четыре многоуровневые архитектуры A1, A2, A3, A4. Приведем зависимость времени и качества работы данных архитектур от тестовых схем. Полученные результаты вычислительного эксперимента представлены в виде гистограмм зависимостей, приведенных на рис. 11 и 12.

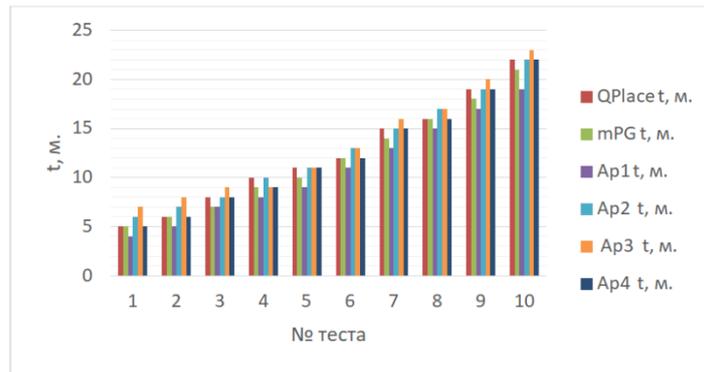


Рис. 11. Гистограмма сравнения времени работы архитектур поиска при реализации тестового примера

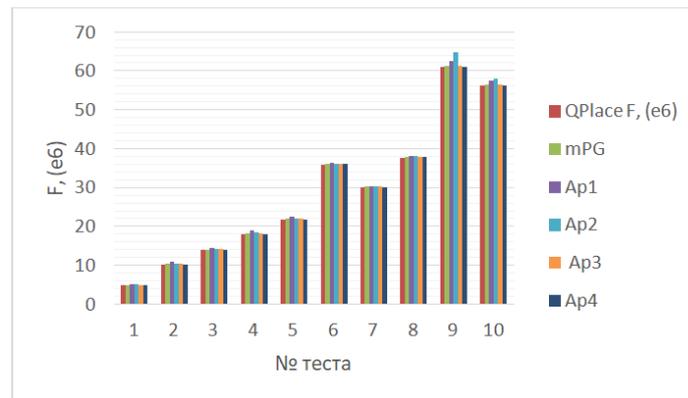


Рис. 12. Гистограмма сравнения значений целевой функции архитектур поиска при реализации тестового примера

В результате анализа представленных графиков зависимостей можно сделать вывод, что разработанные архитектуры биоинспирированного поиска (Ар1, Ар2, Ар3, Ар4) при реализации тестового примера [20] показали сопоставимые результаты по времени решения с известными программными средствами Qplace и mPG. Причем разработанная архитектура Ар4 по качеству решения при реализации тестового примера в среднем на 10 % превосходит другие разработанные архитектуры поиска и на 8 % превосходит известные программные средства Qplace и mPG.

Заключение. Отметим, что предложенный и разработанный функционал конструирования многоуровневых модифицированных архитектур биоинспирированного поиска работает на основе пяти типовых шаблонов и ограниченного числа алгоритмов (блоков), но даже этот ограниченный набор управляющих параметров предоставляет возможность комбинирования более 50 вариантов конкретных архитектур поиска. Это позволяет использовать все достоинства методов биоинспирированной оптимизации для эффективного решения NP-сложных комбинаторно-логических задач на графах. Такой подход дает возможность распараллеливать процесс оптимизации, получать оптимальные и квазиоптимальные решения за полиномиальное время, а также частично решать проблему преждевременной сходимости.

Проведенные экспериментальные исследования, показали преимущество использования программного продукта для решения комбинаторно-логических задач на графах большой размерности, по сравнению с известными методами, так как предложенные архитектуры многоуровневого поиска позволяют получать решения, сопоставимые с известными методами по времени решения, и превосходят последние по качеству получаемых решений в среднем на 8 %.

Временная сложность алгоритмов, реализованных на таких архитектурах, лежит в пределах $O(n \log n) - O(n^3)$, где n – число входов алгоритмов.

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

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Кормен Т., Лейзерсон И., Ривест Р.* Алгоритмы: построения и анализ. – М.: МЦМО, 2000.
2. *De Jong K.* Evolutionary Computation: Recent Development and Open Issues // Proceedings 1st International conf., Evolutionary Computation and Its Application. – М., 1996. – P. 7-18.
3. *Карпенко А.П.* Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учеб. пособие. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2014. – 446 с.
4. *Abraham A., Ramos V., Grosan G.* Swarm Intelligence in Data Mining. – Berlin. Heidelberg: Springer Verlag, 2007.
5. *Hassaniien E. Emary E.* Swarm Intelligence. Principles Advances, and Applications. – CRC Press, 2015.
6. *Mourelle M., Nedjah L. De.* Swarm intelligent systems. – Berlin: Heidelberg: Springer Verlag, 2006.
7. *Родзин С.И., Курейчик В.В.* Состояние, проблемы и перспективы развития биоэвристик // Программные системы и вычислительные методы. – 2016. – № 2. – С. 158-172.
8. *Курейчик В.В., Курейчик Вл.Вл.* Биоинспирированный поиск при проектировании и управлении // Известия ЮФУ. Технические науки. – 2012. – № 11 (136). – С. 178-183.
9. *Kureichik V., Kureichik Vl., Bova V.* Placement of VLSI fragments based on a multilayered approach // Advances in Intelligent Systems and Computing. – 2016. – Vol. 464. – P. 181-190.
10. *Курицкий Б.Я.* Оптимизация вокруг нас: учебник. – СПб.: Машиностроение, 1989.
11. *Hendrickson B., Leland R.* A Multilevel Algorithm for Partitioning Graphs // Proceedings of the 1995 ACM/IEEE conference on Super computing. – P. 626-657.
12. *Schloegel K., Karypis G., Kumar V.* Multilevel diffusion schemes for repartitioning of adaptive meshes. – University of Minnesota, Department of Computer Science, 1997. – P. 109-124.

13. *Barnard S.T., Simon H.D.* A fast multilevel implementation of recursive spectral bisection for partitioning unstructured problems // *Proceedings 6th SIAM Conf. Parallel Processing for Scientific Computing*. – 1993. – P. 711-718.
14. *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.
15. *Курейчик В.М., Курейчик В.В., Родзин С.И.* Модели параллелизма эволюционных вычислений // *Вестник Ростовского государственного университета путей сообщения*. – 2011. – № 3 (43). – С. 93-97.
16. *Бова В.В., Курейчик В.В.* Интегрированная подсистема гибридного и комбинированного поиска в задачах проектирования и управления // *Известия ЮФУ. Технические науки*. – 2010. – № 12 (113). – С. 37-42.
17. *Karaboga D.* An idea based on honey bee swarm for numerical optimization. – Erciyes University, Engineering Faculty, Computer Engineering Department, 2005. – 110 p.
18. *Dorigo M., Maniezzo V., Colomi A.* The Ant System: Optimization by a colony of cooperating objects // *IEEE Trans. on Systems, Man, and Cybernetics*. – 1996. – Part B. – No. 26 (1). – P. 29-41.
19. *Курейчик В.В., Курейчик Вл.Вл.* Размещения фрагментов СБИС на основе механизма агрегации фракталов // *Известия ЮФУ. Технические науки*. – 2015. – № 2 (163). – С. 196-205.
20. *Adya S.N., Markov I.L.* ISPD02 IBM-MS Mixed-size Placement Benchmarks. – <http://vlsicad.eecs.umich.edu/BK/ISPD02bench/>.
21. *Adya S.N., Markov I.L.* Consistent placement of macro-blocks using floor planning and standard-cell placement // *In Proc. Intl. Symp. on Physical Design*. – 2002. – P. 12-17.
22. *Wang M., Yang X., Sarrafzadeh M.* Dragon 2000: Standard-cell Placement Tool for Large Industry Circuits // *ICCAD – 2000*. – P. 260-263.

REFERENCES

1. *Kormen T., Leyzerson I., Rivest R.* *Algorithms: constructions and analysis*. Moscow: MTSMO, 2000.
2. *De Jong K.* Evolutionary Computation: Recent Development and Open Issues, *Proceedings 1st International conf., Evolutionary Computation and Its Application*. Moscow, 1996, pp. 7-18.
3. *Karpenko A.P.* *Sovremennyye algoritmy poiskovoy optimizatsii. Algoritmy, vdokhnovlennyye prirodoy: ucheb. posobie* [Modern search engine optimization algorithms. Algorithms inspired by nature: a textbook]. Moscow: Izd-vo MGTU im. N.E. Bauman, 2014, 446 p.
4. *Abraham A., Ramos V., Grosan G.* *Swarm Intelligence in Data Mining*. Berlin. Heidelberg: Springer Verlag, 2007.
5. *Hassanien E., Emary E.* *Swarm Intelligence. Principles Advances, and Applications*. CRC Press, 2015.
6. *Mourelle M., Nedjah L. De.* *Swarm intelligent systems*. Berlin: Heidelberg: Springer Verlag, 2006.
7. *Rodzin S.I., Kureychik V.V.* Sostoyanie, problemy i perspektivy razvitiya bioevristik [State, problems and prospects of development of bio-heuristics], *Programmnye sistemy i vychislitel'nye metody* [Software systems and computational methods], 2016, No. 2, pp. 158-172.
8. *Kureychik V.V., Kureychik V.I.* Bioinspirirovannyy poisk pri proektirovanii i upravlenii [Bioinspired search in design and management], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2012, No. 11 (136), pp. 178-183.
9. *Kureichik V., Kureichik V.I., Bova V.* Placement of VLSI fragments based on a multilayered approach, *Advances in Intelligent Systems and Computing*, 2016, Vol. 464, pp. 181-190.
10. *Kuritskiy B.Ya.* *Optimizatsiya vokrug nas: uchebnyk* [Optimization around us: a tutorial]. Saint Petersburg: Mashinostroenie, 1989.
11. *Hendrickson B., Leland R.* A Multilevel Algorithm for Partitioning Graphs, *Proceedings of the 1995 ACM/IEEE conference on Super computing*, pp. 626-657.
12. *Schloegel K., Karypis G., Kumar V.* Multilevel diffusion schemes for repartitioning of adaptive meshes. University of Minnesota, Department of Computer Science, 1997, pp. 109-124.
13. *Barnard S.T., Simon H.D.* A fast multilevel implementation of recursive spectral bisection for partitioning unstructured problems, *Proceedings 6th SIAM Conf. Parallel Processing for Scientific Computing*, 1993, pp. 711-718.

14. *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.
15. *Kureychik V.M., Kureychik V.V., Rodzin S.I.* Modeli parallelizma evolyutsionnykh vychisleniy [Models of parallelism of evolutionary calculations], *Vestnik Rostovskogo gosudarstvennogo universiteta putey soobshcheniya* [Bulletin of the Rostov State University of Railway Engineering], 2011, No. 3 (43), pp. 93-97.
16. *Bova V.V., Kureychik V.V.* Integrirovannaya podsistema gibridnogo i kombinirovannogo poiska v zadachakh proektirovaniya i upravleniya [Integrated subsystem of hybrid and combined search in design and management tasks], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2010, No. 12 (113), pp. 37-42.
17. *Karaboga D.* An idea based on honey bee swarm for numerical optimization. Erciyes University, Engineering Faculty, Computer Engineering Department, 2005, 110 p.
18. *Dorigo M., Maniezzo V., Colnani A.* The Ant System: Optimization by a colony of cooperating objects, *IEEE Trans. on Systems, Man, and Cybernetics*, 1996, Part B, No. 26 (1), pp. 29-41.
19. *Kureychik V.V., Kureychik V.V.* Razmeshcheniya fragmentov SBIS na osnove mekhanizma agregatsii fraktalov [Placement of VLSI fragments based on the fractal aggregation mechanism], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2015, No. 2 (163), pp. 196-205.
20. *Adya S.N., Markov I.L.* ISPD02 IBM-MS Mixed-size Placement Benchmarks. Available at: <http://vlsicad.eecs.umich.edu/BK/ISPD02bench/>.
21. *Adya S.N., Markov I.L.* Consistent placement of macro-blocks using floor planning and standard-cell placement, *In Proc. Intl. Symp. on Physical Design*, 2002, pp. 12-17.
22. *Wang M., Yang X., Sarrafzadeh M.* Dragon 2000: Standard-cell Placement Tool for Large Industry Circuits, *ICCAD – 2000*, pp. 260-263.

Статью рекомендовала к опубликованию д.т.н., профессор Л.С. Лисицына.

Курейчик Владимир Викторович – Южный федеральный университет; e-mail: vkur@sfedu.ru; г. Таганрог, Россия; тел.: 88634371651; кафедра систем автоматизированного проектирования; зав. кафедрой САПР; д.т.н.; профессор.

Курейчик Владимир Владимирович – ООО «Газпром подземремонт Уренгой»; e-mail: Kureichik@yandex.ru; г. Санкт-Петербург, Россия; тел.: +78126096700; отдел автоматизации систем управления; к.т.н.; ведущий инженер.

Kureichik Vladimir Victorovich – Southern Federal University; e-mail: vkur@sfedu.ru; Taganrog, Russia; phone: +78634371651; the department of computer aided design; head of CAD department; dr. of eng. sc.; professor.

Kureichik Vladimir Vladimirovich – «Gazprom podzemremont Urengoi» company; e-mail: Kureichik@yandex.ru; St. Peterburg, Russia; phone: +78126096700; Control systems automation department; cand. of eng. sc.; lead engineer.

УДК 004.896

DOI 10.18522/2311-3103-2021-2-50-59

В.В. Курейчик, Вл.Вл. Курейчик, А.Э. Саак

ЭВОЛЮЦИОННЫЙ АЛГОРИТМ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ДИСПЕТЧЕРИЗАЦИИ*

Рассмотрена одна из важных задач оптимизации – задача диспетчеризации. Она относится к классу NP- сложных оптимизационных задач. В работе приведена и описана постановка задачи диспетчеризации. Здесь массив заявок пользователей на компьютерное

* Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-01-00148.