

11. Mylov G.V., Medvedev A.M., Semenov P.V., Konstantinov P.N. Nauchnye osnovy proektirovaniya mezhsosedineniy na pechatnykh platakh [Scientific bases of design of interconnections on printed circuit boards]. Moscow: Goryachaya liniya – Telekom, 2016, 98 p.
12. Kureychik V.M. Matematicheskoe obespechenie konstruktorskogo i tekhnologicheskogo proektirovaniya s primeneniem SAPR: ucheb. dlya vuzov [Mathematical support of design and technological design using CAD: textbook for universities]. Moscow: Radio i svyaz', 1990, 352 p.
13. Kureychik V.M., Safronenkova I.B. Razrabotka arkhitektury SPPR po vyboru metodov resheniya zadach komponovki [Developing the architecture of a DSS for the selection of methods for solving problems in the layout], *Informatsionnye tekhnologii* [Information technology], 2017, Vol. 23, No. 10, pp. 736-741.
14. Sorokoletov P.V. Postroenie intellektual'nykh sistem podderzhki prinyatiya resheniy [The construction of intellectual systems of support of decision-making], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2009, No. 4 (93), pp. 117-124.
15. Karypis G., Kumar V. A fast and high quality multilevel scheme for partitioning irregular graphs, *SIAM J. Scien. Comput.*, 1999, Vol. 20 (1).
16. Segaran T. Programmiruem kollektivnyy razum [Programmable collective intelligence]: transl. from engl. Saint Petersburg: Simvol-Plyus, 2015, 368 p.
17. Madhulatha S.: An overview on clustering methods, *IOSR Journal of Engineering*, Apr. 2012, Vol. 2 (4), pp: 719-725.
18. Maimon O., Rokach L. Data Mining and Knowledge Discovery Handbook. Springer US, 2010.
19. Tikhonov A.N. *TSvetkov V.Ya. Metody i sistemy podderzhki prinyatiya resheniy* [Methods and systems of decision support]. Moscow: MAKS Press, 2001.
20. Greshilov A.A. Matematicheskie metody prinyatiya resheniy [Mathematical methods of decision-making]. Moscow: Izd-vo MGTU im. N.E. Baumana, 2006, 584 p.

Статью рекомендовал к опубликованию д.т.н. Э.В. Мельник.

Курейчик Виктор Михайлович – Южный федеральный университет; e-mail: vmkureychik@sfedu.ru; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 88634681887; кафедра САПР; г.н.с.; д.т.н.; профессор.

Сафроненкова Ирина Борисовна – e-mail: safronenkova050788@yandex.ru; г. Таганрог, пер. Смирновский, 137, кВ. 64; тел.: 89604678753; кафедра САПР; аспирант

Kureichik Viktor Mikhaylovich – Southern Federal University; e-mail: vmkureychik@sfedu.ru; 44, Nekrasovskiy, Taganrog, 347928, Russia; phone: +78634681887; the department of CAD; chief researcher; dr. of eng. sc.; professor.

Safronenkova Irina Borisovna – e-mail: safronenkova050788@yandex.ru; 137, Smirnovskiy street, 64 apart, Taganrog, Russia; phone: +79604678753; the department of CAD; postgraduate student.

УДК 004.896

DOI 10.23683/2311-3103-2018-4-74-89

Б.К. Лебедев, О.Б. Лебедев, Е.О. Лебедева

КО-ЭВОЛЮЦИОННЫЙ АЛГОРИТМ ТРАССИРОВКИ, ОСНОВАННЫЙ НА МЕТОДЕ МУРАВЬИНОЙ КОЛОНИИ*

Для решения задачи канальной трассировки (ЗКТ) предложен ко-эволюционный алгоритм, основанный на методе муравьиной колонии. Ко-алгоритм предполагает параллельное функционирование заданного числа субалгоритмов муравьиной колонии, которые используют различные, но изоморфные стратегии поиска. Предложенный подход позволяет организовать систему коллективной адаптации с высокой степенью целесообразного поведения и сходимости. Ключевая проблема, которая была решена в данной работе, связана

* Работа выполнена при финансовой поддержке гранта РФФИ № 17-07-00997 А.

с разработкой принципов взаимодействия субпопуляций, отличающиеся стратегиями поиска интерпретаций решений в изоморфных средах функционирования. Периодически агенты мигрируют из одной субпопуляции в другую, передавая свой опыт. В качестве интерпретации решения ЗКТ используется список. Каждый список фактически является косвенной (числовой) схемой кодирования решения ЗКТ. Декодер – оператор, осуществляющий укладку горизонтальных фрагментов по заложенным в нем правилам, позволяющий перейти от косвенной (числовой) схемы кодирования решения задачи к фенотипу. Список декодируется в графическое представление решения (эскиз) только с использованием соответствующей ему стратегии. Основная цель сводится к нахождению такого списка, для которого с помощью последовательной процедуры декодирования фрагменты цепей размещаются в минимальном числе магистралей. Для построения конструктивной (встроенной) процедуры поиска решения муравьем используется модифицированный алгоритм левого конца. Последовательное построение маршрута фактически является процессом последовательного размещения горизонтальных фрагментов в магистралях. Для усиления сходимости алгоритма и способности выхода из локальных оптимумов разработана комбинированная оценка, характеризующая преимущество выбора данной вершины, для включения ее в формируемый маршрут. В основу расчета вероятности включения вершины в формируемый маршрут положены эвристические соображения о предпочтениях того, что вершина входит в состав оптимального маршрута. Ко-эволюционный подход обеспечивает более широкий обзор пространства решений и более высокую вероятность локализации глобального экстремума задачи. По сравнению с существующими алгоритмами достигнуто улучшение качества решений до 5 %.

Биоинспирированная оптимизация; ко-эволюционный алгоритм; муравьиный алгоритм; гибридизация; субпопуляция; канальная трассировка; декодер.

B.K. Lebedev, O.B. Lebedev, E.O. Lebedeva

CO-EVOLUTION ROUTING ALGORITHM BASED ON THE ANT COLONY METHOD FOR SOLVING THE CHANNEL ROUTING PROBLEM

In this paper, a co-evolution algorithm based on the ant colony method has been proposed for solving the channel routing problem (CRP). The co-algorithm involves the parallel operation of a given number of subalgorithms of the ant colony, which use different, but isomorphic search strategies. The proposed approach allows you to organize a system of collective adaptation with a high degree of expedient behavior and convergence. The key problem that was solved in this paper is related to the development of principles for the interaction of subpopulations, which differ in strategies for finding interpretations of solutions in isomorphic functioning environments. Periodically, agents migrate from one subpopulation to another, transferring their experience. A list is used to interpret the CRP solution. In fact, each list is an indirect (numeric) coding scheme for a QCT solution. A decoder is an operator performing the laying of horizontal fragments according to the rules laid down in it, which allows one to go from an indirect (numerical) coding scheme for solving a problem to a phenotype. The list is decoded into a graphical representation of the solution (sketch) only using the appropriate strategy. The main goal is to find a list for which, using a sequential decoding procedure, fragments of chains are placed in the minimum number of trunks. To construct a constructive (built-in) procedure for finding a solution, the ant uses a modified left-end algorithm. Successive construction of a route is actually a process of sequentially placing horizontal fragments in highways. To enhance the convergence of the algorithm and the ability to exit from local optima, a combined estimate has been developed that characterizes the advantage of choosing a given vertex to include it in the generated route. The basis of the formulas for calculating the probability of the inclusion of a vertex in the formed route is based on heuristic considerations about the preferences that the vertex is part of the optimal route. The co-evolutionary approach provides a wider overview of the solution space and a higher probability of localizing the global extremum of the problem. Compared to existing algorithms, an improvement in the quality of solutions has been achieved up to 5 %.

Swarm intelligence; optimization; co-evolutionary algorithm; ant algorithm; hybridization; subpopulation; channel routing; decoder.

Введение. Одной из важнейших задач при построении интеллектуальных систем автоматизированного проектирования (САПР) сверхбольших интегральных схем (СБИС), является трассировка соединений, которая во многом определяет качество топологии СБИС. Для трассировки проводников используются различные виды алгоритмов – волновые, лабиринтные, эвристические и каналные [1, 2]. Ввиду грандиозной сложности современных СБИС наиболее привлекательными являются каналные алгоритмы, отличающиеся возможностью одновременной прокладки всех цепей, малым расходом оперативной памяти, меньшим временем решения по сравнению с остальными алгоритмами. Задача каналной трассировки (КТ) является хорошо изученной, и существуют сотни подходов к ее решению: алгоритмы левого конца, “жадные”, графовые алгоритмы, иерархические и т.д. [3], но по-прежнему является актуальной. Это связано, в первую очередь, с тем, что эта задача является NP-полной, и разработать универсальный алгоритм, позволяющий находить точное оптимальное решение за приемлемое время затруднительно. Однако наличие новых тенденций в проектировании СБИС вызывает потребность в разработке новых подходов к ее реализации. В последнее время для решения различных «сложных» задач, к которым относятся и задачи каналной трассировки, всё чаще используются методы, инспирированные природными системами [4]. Такие методы являются итеративными, эвристическими методами случайного поиска. Среди них особенно активно развиваются методы роевого интеллекта [5,6]. Анализ методов решения сложных прикладных задач показывает, что применение любого одного алгоритма оптимизации (как классического, так и популяционного) далеко не всегда приводит к успеху. В связи с этим, в настоящее время одним из основных путей повышения эффективности методов для решения задач глобального поиска является разработка гибридных алгоритмов [7,8]. Одним из эффективных подходов является ко-алгоритмическая гибридизация – реализующая кооперативную стратегию, обеспечивающую более широкий обзор пространства поиска и более высокую вероятность локализации глобального экстремума задачи [9]. В работе для решения задачи каналной трассировки (ЗКТ) разработан ко-эволюционный алгоритм трассировки, основанный на методе муравьиной колонии. Ко-алгоритм [9] предполагает параллельное функционирование заданного числа субалгоритмов муравьиной колонии, которые используют различные, но изоморфные стратегии поиска. Предложенный подход позволяет организовать систему коллективной адаптации с высокой степенью целесообразного поведения и сходимости.

1. Постановка задачи. Канал представляет собой область, ограниченную двумя линейками контактов: верхний ряд контактов $T=\{t_1, t_2, \dots, t_l\}$ и нижний ряд контактов $B=\{b_1, b_2, \dots, b_l\}$ [3]. К верхнему ряду контактов подходит соответственно множество цепей $N_T=\{nt_i | i=1, 2, \dots, l\}$, а к нижнему – соответственно множество цепей $N_B=\{nb_i | i=1, 2, \dots, l\}$, $N_T \cup N_B = N$. На область трассировки наносится опорная ортогональная сеть, по линиям которой проходят трассы. Вертикальные линии проходят через контакты. Горизонтальные линии называют магистралями. Контакты в линейке пронумерованы слева направо и помечены номерами подходящих к ним цепей. Некоторые цепи могут распространяться за левую и правую границы канала. При каналной трассировке каждая цепь, связывающая эквипотенциальные выводы, представляется в виде набора горизонтальных и вертикальных фрагментов (участков). Задача трассировки в канале рассматривается как задача распределения фиксированного множества горизонтальных участков $F=\{f_i | i=1, 2, \dots, n\}$ в множестве магистралей $M=\{m_j | j=1, 2, \dots, l\}$. На взаимное расположение горизонтальных фрагментов накладываются ограничения, задаваемые с помощью графа вертикальных ограничений (ГВО) $G_V=(X, E)$ и графа горизонтальных ограничений

(ГГО) $G_H=(X,U)$ [1], где X – множество вершин, соответствующих горизонтальным фрагментам. Направленное ребро $e_k=(x_i,x_j) \in E$ существует тогда, и только тогда, когда фрагмент f_i должен быть расположен выше фрагмента f_j для предотвращения наложений вертикальных фрагментов цепей. Ребро $u_k=(x_i,x_j) \in U$ существует, тогда и только тогда, когда для исключения наложения друг на друга горизонтальные фрагменты f_i и f_j должны размещаться в разных магистралях. Соблюдение ограничений гарантирует разнесение соединений в два слоя без пересечений и наложений в пределах слоя [1, 3, 11].

Чтобы задача решалась в рамках классической постановки, граф G_V должен быть ациклическим. Наличие цикла в G_V свидетельствует о невозможности трассировки без нарушения вертикальных ограничений. Для ликвидации цикла в G_V необходимо произвести разбиение какой-нибудь вершины $x_i \in E$, входящей в цикл, другими словами осуществить разбиение горизонтального фрагмента путем введения излома. Существует ряд подходов к процессу разбиения горизонтальных фрагментов [1, 3], выполняемых как до основного процесса трассировки, так и в процессе трассировки. Существуют различные подходы к формированию горизонтальных фрагментов.

В работе рассматривается постановка ЗКТ с априори ликвидированными циклами в графе G_V . Основными критериями при канальной трассировке являются [1, 2, 11]: число магистралей; суммарная длина соединений.

2. Канальная трассировка на основе парадигмы муравьиной колонии. Метаэвристика муравьиного алгоритма [5, 6] основывается на комбинации двух методов: общая структура муравьиного алгоритма строится на базовом методе, который включает ту или иную встроенную процедуру. Важным аспектом есть то, что встроенная процедура – это в большинстве случаев самостоятельный алгоритм решения той же задачи, что и метаэвристический метод в целом. Базовый метод заключается в реализации итерационной процедуры поиска лучшего решения, на основе механизмов адаптивного поведения муравьиной колонии. Основу встроенной процедуры составляет конструктивный алгоритм построения муравьем некоторой конкретной интерпретации решения. В оптимизации муравьиными колониями [6], конструктивный блок - деятельность искусственных муравьев играет ключевую роль.

Для представления и поиска интерпретаций решения ЗКТ используется полный бинарный ориентированный граф $R(F,E)$, где множество вершин F соответствует множеству фрагментов, E – множество ребер полного графа [10]. $|F|=n_f$, $|E|=n_e$. Каждая вершина $f_i \in F$ связана исходящими из неё ориентированными ребрами со всеми остальными вершинами множества F . Таким образом, число ребер графа $n_e=n_f(n_f-1)$, причем каждая пара вершин (f_i, f_j) связана двумя ребрами с взаимно противоположной ориентацией.

В качестве интерпретации решения ЗКТ служит некоторый ориентированный маршрут S_k , включающий все вершины на графе решений $R(F,E)$ [10]. Каждая пара (f_i, f_j) соседних вершин в маршруте S_k , связана ориентированным ребром. Последовательность вершин в маршруте S_k задает расположение соответствующих элементов в списке F_k .

Задача каждого муравья $a_k \in A$ заключается в построении на графе R *встроенным* алгоритмом маршрута S_k , являющегося решением ЗКТ, где k – номер агента. Задача колонии муравьев – нахождение *базовым* алгоритмом в графе $R(F,E)$ маршрута, которому соответствует размещение фрагментов цепей в наименьшем числе магистралей. Для создания равных условий, связанных с выбором первой вершины в маршруте, осуществляется диверсификация графа поиска решений $R(F,E)$ путем ввода дополнительной стартовой вершины – O , связанной ориентированно

ными ребрами со всеми вершинами графа $R(F,E)$. Построение муравьем маршрута всегда начинается на диверсифицированном графе поиска решений $R^*(F,E)$ со стартовой вершины – O . Отметим, что вершина O не входит в состав маршрута, являющегося решением ЗКТ.

Для построения встроенной процедуры поиска и метода представления решения ЗКТ воспользуемся модификацией алгоритма левого конца (АЛК), суть которого заключается в следующем [1]. Пусть имеется некоторый список горизонтальных фрагментов $F = \{f_i | i = 1, 2, \dots, n\}$. Магистраль для заполнения фрагментами цепей выбираются последовательно, начиная с первой. Каждая выбранная магистраль m_j последовательно заполняется горизонтальными фрагментами цепей f_i . На каждом шаге t заполнения магистрали среди еще не размещенных фрагментов $F^*(t) \subset F$ выбирается с соблюдением ограничений G_H и G_V фрагмент $f_i \in F^*(t)$, который помещается в магистраль m_j и исключается из $F^*(t)$. Основным достоинством данного алгоритма является возможность быстрого получения результата, но при этом получаемый результат в большинстве случаев далек от глобального оптимума. Если после заполнения текущей магистрали множество еще не размещенных фрагментов не пусто, то для размещения выбирается следующая по порядку магистраль. В результате работы алгоритма строится последовательность S_k .

Очевидно, что существует последовательность S_{opt} , такая, что если осуществлять последовательное заполнение магистралей фрагментами в соответствии с последовательностью S_{opt} , получится оптимальное решение.

Назовем любую последовательность S_k решением ЗКТ. Сопоставим решению S_k оценку ζ_k – число занимаемых магистралей. Очевидно, что существует такой упорядоченный список S_{opt} , с таким взаимным расположением элементов, что соответствующее ему решение ЗКТ оптимально (по числу занимаемых магистралей). Тогда ЗКТ сводится к нахождению решения S_{opt} , для которого

$$\zeta_{opt} \rightarrow \min \{\zeta_k\}. \tag{1}$$

Основная цель сводится к нахождению такого списка S_{opt} , для которого с помощью последовательной процедуры декодирования фрагменты цепей размещаются в минимальном числе магистралей ζ_{opt} , т.е. $\forall k (\zeta_k \geq \zeta_{opt})$.

На рис. 1 представлено оптимальное (по числу занимаемых магистралей) решение, соответствующее последовательности $S_{opt} = \langle f_1, f_5, f_6, f_8, f_2, f_4, f_9, f_{11}, f_3, f_7, f_{10}, f_{12} \rangle$.

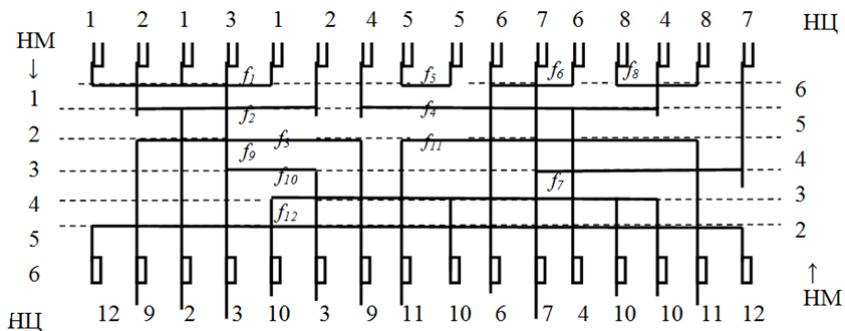


Рис. 1. Пример Г1 с оптимальным решением задачи трассировки в канале

Процесс поиска решений муравьиным алгоритмом итерационный. Каждая итерация l включает три этапа. На первом этапе каждой итерации каждый муравей a_k строит *встроенным* конструктивным алгоритмом на ГПП маршрут S_k . На втором этапе *базового* алгоритма (итерации) производится отложение феромона.

В работе используется циклический (ant-cycle) метод муравьиных систем [6]. В этом случае феромон откладывается на ребрах маршрутов только после того, как маршруты будут построены всеми агентами. Количество феромона, откладываемого на ребрах маршрута, пропорционально оценке маршрута. На третьем этапе итерации осуществляется испарение феромона на ребрах ГПП. Завершается итерация определением лучшего решения, найденного муравьями на этой итерации и лучшего решения среди всех итераций.

3. Ко-эволюционный алгоритм трассировки на основе муравьиной колонии. Основная идея ко-эволюционных алгоритмов (ко-алгоритмов) оптимизации состоит в следующем [4, 9]. Одновременно в пространстве поиска задачи оптимизации эволюционируют несколько субпопуляций, каждая из которых решает исходную оптимизационную задачу и имеет свою стратегию оптимизации. Целью ко-гибридизации алгоритмов поисковой оптимизации является повышение их эффективности, для усиления сходимости алгоритма и способности выхода из локальных оптимумов.

Поскольку канал симметричен (см. рис. 1), то существуют четыре стратегии формирования муравьем решения ЗКТ, в зависимости от порядка нумерации магистралей (НМ) и порядка последовательного заполнения магистралей (ПЗМ) фрагментами:

- Стратегия S_1 – (НМ - сверху вниз, ПЗМ - слева направо);
- Стратегия S_2 – (НМ - сверху вниз, ПЗМ - справа налево);
- Стратегия S_3 – (НМ - снизу вверх, ПЗМ - слева направо);
- Стратегия S_4 – (НМ - снизу вверх, ПЗМ - справа налево).

На рис. 2 для решения примера Т1 (см. рис. 1) показаны маршруты S_{1k} и S_{2k} в графе R , сформированные в соответствии с первой и второй стратегиями.

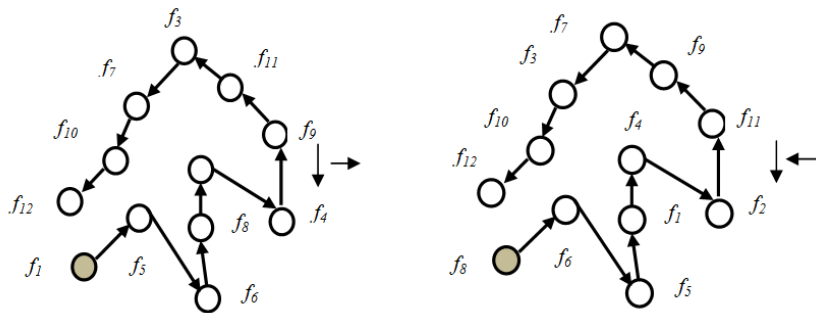


Рис. 2. Маршруты S_{1k} и S_{2k} в графе R

На рис. 3 для примера Т1 (см. рис. 1) показаны маршруты S_{3k} и S_{4k} в графе R , сформированные в соответствии с третьей и четвертой стратегиями.

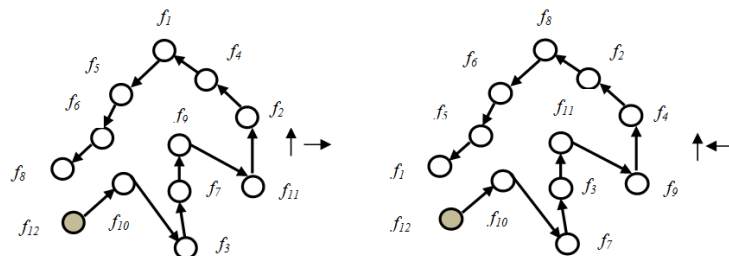


Рис. 3. Маршруты S_{3k} и S_{4k} в графе R

Ниже приведены списки, сформированные с использованием разных стратегий, но соответствующие одному решению на рис. 1.

$S_{1k} = \langle f_1, f_5, f_6, f_8, f_2, f_4, f_9, f_{11}, f_3, f_7, f_{10}, f_{12} \rangle$ - в соответствии со стратегией C_1 ;

$S_{2k} = \langle f_8, f_6, f_5, f_1, f_4, f_2, f_{11}, f_9, f_7, f_3, f_{10}, f_{12} \rangle$ - в соответствии со стратегией C_2 ;

$S_{3k} = \langle f_{12}, f_{10}, f_3, f_7, f_9, f_{11}, f_2, f_4, f_1, f_5, f_6, f_8 \rangle$ - в соответствии со стратегией C_3 ;

$S_{4k} = \langle f_{12}, f_{10}, f_7, f_3, f_{11}, f_9, f_4, f_2, f_8, f_6, f_5, f_1 \rangle$ - в соответствии со стратегией C_4 .

Будем набор таких списков называть набором изоморфных списков, а набор маршрутов, соответствующих этим спискам - набором изоморфных маршрутов. Для одного и того же решения список, сформированный по какой-либо стратегии путем трансформации, может быть преобразован в остальные три изоморфных ему списка. Список рассматривается, как совокупность групп фрагментов. Группа – совокупность фрагментов, размещенных на одной магистрали. Трансформация реализуется с помощью двух операторов: OT1 – инверсия порядка расположения групп фрагментов в списке; OT2 – инверсия порядка расположения фрагментов в группе. Представим список $S_{1k} = \langle f_1, f_5, f_6, f_8, f_2, f_4, f_9, f_{11}, f_3, f_7, f_{10}, f_{12} \rangle$ в виде совокупности групп фрагментов – $\langle f_1, f_5, f_6, f_8 \rangle$, $\langle f_2, f_4 \rangle$, $\langle f_9, f_{11} \rangle$, $\langle f_3, f_7 \rangle$, $\langle f_{10} \rangle$, $\langle f_{12} \rangle$. С помощью оператора OT2 список S_{1k} трансформируется в изоморфный список S_{2k} . С помощью оператора OT1 список S_{2k} трансформируется в изоморфный список S_{4k} . С помощью оператора OT2 список S_{4k} трансформируется в изоморфный список S_{3k} .

Отметим, что каждый список фактически является косвенной (числовой) схемой кодирования решения ЗКТ. Графическое представление решения ЗКТ можно получить, с помощью декодера. Декодер – оператор, осуществляющий укладку горизонтальных фрагментов по заложенным в нем правилам, позволяющий перейти от косвенной (числовой) схемы кодирования решения задачи к фенотипу. Каждый список декодируется в графическое представление решения (эскиз) только с использованием соответствующей ему стратегии.

С учетом выше перечисленных свойств, для решения ЗКТ разработан ко-эволюционный алгоритм на основе муравьиной колонии. Суть ко-эволюционного подхода состоит в том, что эволюционирующая популяция делится на субпопуляции, которые эволюционируют параллельно. Периодически агенты мигрируют из одной субпопуляции в другую, передавая свой опыт. Ко-эволюционный подход обеспечивает более широкий обзор пространства решений и более высокую вероятность локализации глобального экстремума задачи. Работа ко-эволюционного алгоритма трассировки осуществляется по следующей схеме.

Решение ЗКТ осуществляется четырьмя субпопуляциями (группами) $A_1 - A_4$ муравьев на четырех идентичных друг другу ГПП $R_1^*(F, E) - R_4^*(F, E)$. Муравьи субпопуляции A_1 строят маршруты в соответствии со стратегией C_1 на ГПП $R_1^*(F, E)$. Соответственно: муравьи субпопуляции A_2 – со стратегией C_2 на ГПП $R_2^*(F, E)$; муравьи субпопуляции A_3 – со стратегией C_3 на ГПП $R_3^*(F, E)$; муравьи субпопуляции A_4 – со стратегией C_4 на ГПП $R_4^*(F, E)$.

На начальном этапе на всех ребрах ГПП $R_1^*(F, E) - R_4^*(F, E)$ откладывается одинаковое (небольшое) количество феромона, равное Ω .

Процесс поиска решений итерационный. Каждая итерация l включает три этапа. На первом этапе каждой итерации каждый муравей $a_{tk} \in A_{\tau}$, в соответствии со стратегией C_{τ} , строит на ГПП $R_{\tau}^*(F, E)$ маршрут S_{tk} . Определяется оценка ξ_{tk} маршрута S_{tk} . Муравьи, входящие в состав разных групп, строят решения параллельно.

Множество S_{τ} маршрутов, построенное муравьями каждой субпопуляции A_{τ} , заносится в базу построенных на итерации маршрутов – BS_{τ} .

В каждом множестве S_{τ} выбирается группа лучших маршрутов A_{τ} . Посредством трансформации для каждой группы маршрутов A_{τ} формируются группы изоморфных маршрутов A_{τ}^* . Для A_1 формируются A_{2}^* , A_{3}^* , A_{4}^* . Для A_2 формируются A_{1}^* , A_{3}^* , A_{4}^* . Для A_3 формируются A_{1}^* , A_{2}^* , A_{4}^* . Для A_4 формируются A_{1}^* , A_{2}^* , A_{3}^* .

Далее для каждой субпопуляции A_τ база построенных маршрутов BS_τ дополняется, путем включения в нее групп изоморфных маршрутов A_τ^* маршрутов: для A_1 включение в BS_1 групп A_2^* , A_3^* , A_4^* ; для A_2 включение в BS_2 групп A_1^* , A_3^* , A_4^* ; для A_3 включение в BS_3 групп A_1^* , A_2^* , A_4^* ; для A_4 включение в BS_4 групп A_1^* , A_2^* , A_3^* .

На втором этапе производится отложение феромона. В работе используется циклический (ant-cycle) метод муравьиных систем [6]. В этом случае феромон откладывается на ребрах маршрутов, построенных в графах $R_1^*(F,E)$ - $R_4^*(F,E)$, только после того, как маршруты будут построены всеми агентами. Количество феромона, откладываемого на ребрах каждого маршрута, пропорционально оценке маршрута.

В ГПП $R_1^*(F,E)$ феромон откладывается на ребрах маршрутов, входящих в состав базы BS_1 ; в ГПП $R_2^*(F,E)$ – в состав базы BS_2 ; в ГПП $R_3^*(F,E)$ – в состав базы BS_3 ; в ГПП $R_4^*(F,E)$ – в состав базы BS_4 .

На третьем этапе итерации осуществляется испарение феромона на ребрах графов $R_1^*(F,E)$ - $R_4^*(F,E)$. Завершается итерация определением лучшего решения, найденного на этой итерации и лучшего решения среди всех итераций.

Формирование муравьем $a_{tk} \in A_\tau$ на ГПП $R_\tau^*(F,E)$ маршрута S_{tk} осуществляется последовательно (пошагово), в соответствии с рассмотренным выше модифицированным алгоритмом левого конца. Последовательное построение маршрута фактически является процессом последовательного размещения горизонтальных фрагментов в магистралях. В соответствии с выбранной стратегией S_τ формирования маршрута фиксируется порядок нумерации магистралей и порядок последовательного заполнения магистралей фрагментами. Построение маршрута в ГПП $R_\tau^*(F,E)$ начинается со стартовой вершины O_τ . В процессе заполнения магистрали m на каждом шаге t агент a_{tk} применяет вероятностное правило выбора следующей вершины для включения ее в формируемый маршрут $S_{tk}(t)$. На конечном шаге $t=n$ агентом a_{tk} будет сформирован маршрут $S_{tk} = S_{tk}(n)$.

На шаге t в памяти агента a_{tk} хранится номер формируемой магистрали $m(t)$, список $F1_{tk}(t)$ вершин графа $R_\tau^*(F, E)$, уже включенных в формируемый маршрут и список $F2_{tk}(t)$ оставшихся вершин (не вошедших маршрут), $F1_{tk}(t) \cup F2_{tk}(t) = F$.

Выбор следующей вершины для включения ее в формируемый маршрут осуществляется следующим образом. Пусть выполняется l -я итерация и к данному моменту агентом a_{tk} сформирован маршрут $S_{tk}(t)$, а f_z – последняя вершина (фрагмент) маршрута $S_{tk}(t)$. Агент просматривает множество всех свободных на данном шаге вершин $F2_{tk}(t)$ и выделяет из него подмножество вершин $Z2_{tk}(t)$, удовлетворяющих условиям размещения фрагментов в текущей магистрали $m(t)$.

Условия размещения:

а) каждый фрагмент $f_i \in Z2_{tk}(t)$ может быть размещен в текущей заполняемой магистрали $m(t)$ без нарушения ограничений, задаваемых G_H и G_V , с уже размещенными фрагментами;

б) если $f_i \in Z2_{tk}(t)$ поместить в магистраль $m(t)$, то не существует фрагмента $f_j \in Z2_{tk}(t)$, который может быть размещен в оставшейся свободной зоне заполняемой магистрали между фрагментами f_z и f_i без нарушения ограничений и конфликтов с f_i .

Соблюдение рассмотренных условий объясняется следующим образом. Поскольку заполнение магистралей фрагментами производится последовательно с использованием “безвозвратной” стратегии нет смысла оставлять на магистрали не заполненными области, в которые могут быть помещены некоторые фрагменты.

На рис. 4 представлена ситуация, сложившаяся на шаге 5 ($t=5$) построения маршрута для примера на рис. 1.

Номер заполняемой магистрали – $m(t)=2$; сформирован маршрут $S_{lk}(5)=\{f_1, f_5, f_6, f_8, f_2\}$, а $f_z=f_2$ – последняя вершина (фрагмент) маршрута $S_{lk}(5)$. Агент просматривает множество всех свободных на данном шаге вершин $F_{2lk}(5)$ и выделяет из него подмножество вершин $Z_{2tk}(5)=\{f_4, f_{11}, f_7\}$, удовлетворяющих условиям размещения фрагментов в текущей магистрали. Для каждой вершины $f_i \in Z_{2tk}(t)$, рассчитываются три параметра $h_i(l-1)$, d_i , w_i :

$h_i(l-1)$ – суммарное количество феромона, скопившегося на ориентированном ребре, связывающим последнюю вершину f_z маршрута $S_{lk}^c(t)$ и вершину $f_i \in Z_{2tk}(t)$, после выполнения $(l-1)$ итераций;

d_i – расстояние между окончанием последнего в формируемом маршруте фрагмента $f_z \in S_{lk}(t)$ и началом фрагмента f_i , размещенных в одной магистрали. Если f_i располагается в начале магистрали, то d_i – расстояние между началом магистрали и началом f_i ;

w_i – соотношение числа связей f_i с линейками контактов. При НМ сверху вниз $w_i=(tk_i - bk_i)$, при НМ снизу вверх $w_i=(bk_i - tk_i)$; где tk_i - число связей f_i с верхней линейкой, bk_i - число связей f_i с нижней линейкой контактов.

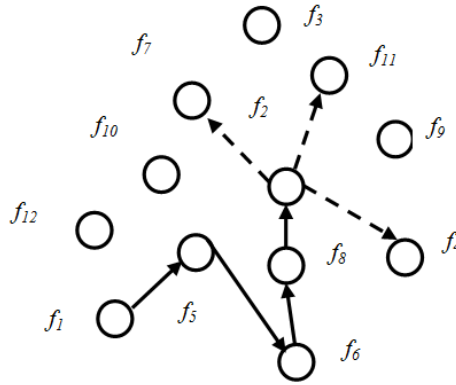


Рис. 4. Выбор муравьем вершины для включения в маршрут

Пусть для нашего примера (см. рис.1): $h_4(l-1)=4$; $h_{11}(l-1)=3$; $h_7(l-1)=3$; $d_4=1$; $d_{11}=2$; $d_7=5$ $w_4=2-1=1$; $w_{11}=0-2= -2$; $w_7=2-1=1$.

Интегральная стоимость $Q_{iz}(t)$ связи вершины $f_i \in Z_{2tk}(t)$ с $f_z \in S_{lk}(t)$ определяется при мультипликативной свертке по формуле (2), а при аддитивной свертке по формуле (3).

$$Q_{iz}(t)=(h_i^\alpha(l-1) \cdot w_i^\gamma)/(d_i^\beta + 1), \tag{2}$$

$$Q_{iz}(t)=\alpha h_i(l-1) + (\beta/(d_i + 1) + \gamma w_{ik}), \tag{3}$$

где α , β , γ – управляющие параметры, которые подбираются экспериментально.

Вероятность $P_i(t)$ включения вершины $f_i \in Z_{2tk}(t)$ в формируемый маршрут $S_{lk}(t)$ определяется следующим соотношением:

$$P_i(t)= Q_{iz}(t)/\theta, \text{ где } \theta=\sum_i(Q_{iz}(t)), i/(f_i \in Z_{2tk}(t)). \tag{4}$$

Для нашего примера: примем значения управляющих параметров α , β , γ равными 1. Тогда, в соответствии с формулой (3) $Q_{4,2}(5)=1 \cdot 4+1/2+1 \cdot 1=5,5$. $Q_{11,2}(5)=1 \cdot 3+1/3-1 \cdot 2=1,333$. $Q_{7,2}(5)=1 \cdot 3+1/6+1 \cdot 1=5,167$. $\theta=5,5+1,333+5,167=12$. Отсюда $P_4(5)=5,5/12=0,458$. $P_{11}(5)=1,333/12=0,111$. $P_7(5)=5,167/12=0,431$. $\sum_i(P_i(t))=1$.

Рассмотрим эвристические соображения, положенные в основу формул (2) и (3). Чем больше $h_i(l-1)$, тем больше вероятность того, что вершина f_i входит в состав оптимального маршрута. Чем меньше d_i , тем выше плотность заполнения магистралей, что способствует уменьшению числа магистралей, необходимого для размещения фрагментов.

Учет в выражениях (2) и (3) параметра w_i способствует сокращению числа вертикальных ограничений в множестве $F2_k^r(t)$ еще не вошедших в маршрут вершин, что косвенным образом способствует повышению эффективности поиска и минимизации целевой оценки – числа магистралей. Кроме того, чем больше w_i , тем меньше суммарная длина вертикальных участков подходящих к рассматриваемому фрагменту f_i .

Агент с вероятностью $P_i(t)$ выбирает одну из вершин $f_i \in Z2_{tk}(t)$, которая включается в маршрут $S_k^r(t)$. После этого маршрут уже обозначается как $S_{tk}(t+1)$, а f_i исключается из множества $F2_k^r(t)$. При $\alpha=0$ наиболее вероятен выбор ближайшего к f_i фрагмента f_i , то есть алгоритм становится подобным алгоритму “левого конца”. При $\beta=0$ вероятность выбора практически зависит только от количества феромона на ребре, что приводит к субоптимальным решениям. Соотношение между этими величинами находится экспериментально. Для каждого построенного маршрута S_{tk} рассчитывается его оценка ζ_{tk} .

После формирования всеми муравьями маршрутов на ребрах каждого ГПР $R_r^*(F, E)$, откладывается феромон. В пределах базы построенных маршрутов $BS_r(l)$ последовательно выбираются маршруты S_{tk} и на ребрах $R_r^*(F, E)$, соответствующих ребрам маршрута $S_{tk} \in BS_r(l)$, откладывается феромон, в количестве δ_{tk} пропорциональном оценке ζ_{tk} маршрута S_{tk} .

$$\delta_{tk} = \Delta / \zeta_{tk}, \quad (5)$$

где Δ – базовое количество феромона, используемое всеми муравьями для расчета количества феромона, откладываемого на ребрах построенных ими маршрутов. ζ_{tk} – целевая функция для данного решения, т.е. *число магистралей*, использованных муравьем a_{tk} на l -ой итерации при прокладке маршрута S_{tk} .

Обозначим как $h_{tk}^j(l)$ общее количество феромона, скопившегося на ребре $e_j \in E$ в ГПР $R_r^*(F, E)$ после выполнения муравьями второго этапа l -й итерации.

На третьем этапе итерации на ребрах каждого графа $R_r^*(F, E)$ в соответствии с нижеприведенной формулой происходит испарение феромона.

$$h_{tk}^j(l) = h_{tk}^j(l) \cdot (1 - \rho), \quad (6)$$

где ρ – коэффициент обновления.

Последние действия на итерации связаны с нахождением лучшего решения, которое запоминается. После этого выполняется следующая итерация. Временная сложность этого алгоритма зависит от количества вершин графа n , числа муравьев u , числа итераций l , и определяется как $O(n^2ly)$.

Представим общую процедуру поиска решения в виде двух алгоритмов: *Базовый алгоритм поведения муравьиной колонии*; *Встроенный алгоритм поведения муравья*.

Базовый алгоритм поведения муравьиной колонии.

0. Дано: Нижний ряд контактов $B = \{b_i / i = \overline{1, n_b}\}$, верхний ряд контактов $T = \{t_i / i = \overline{1, n_t}\}$. К контактам подходят цепи: к нижнему ряду – множество цепей $C_b = \{c_{bi} / i = \overline{1, n_b}\}$, а к верхнему – множество цепей $C_t = \{c_i / i = \overline{1, n_t}\}$, $C_b \cup C_t = C$.
 Задаются: значения управляющих параметров α, β, γ ;

начальное количество феромона – Ω ;

базовое количество феромона – Δ ;

тип свертки – аддитивная или мультипликативная.

1. Формируется множество горизонтальных участков (фрагментов) $F = \{f_i / i = 1, n_f\}$.
2. Строятся графы ограничений $G_V = (V, E)$, $G_H = (V, U)$.
3. Строится полный граф поиска решений $R(F, E)$.
4. На основе $R(F, E)$ формируются четыре идентичных друг другу диверсифицированных ГПР $R_1^*(F, E)$ - $R_4^*(F, E)$.
5. Для четырех стратегий заполнения магистралей фрагментами задается число n_a муравьев в каждой субпопуляции $A_1 - A_4$.
6. На всех ребрах в ГПР $R_1^*(F, E)$ - $R_4^*(F, E)$ откладывается одинаковое количество феромона, равное Ω . Задается число итераций n_l .
7. $l = 1$. (l – номер итерации).

1-й этап итерации l .

8. Обнуляются базы хранения построенных маршрутов BS_1 - BS_4 .
9. $\tau = 1$. (τ – номер субпопуляции. Выбирается стратегия C_τ).
10. $k = 1$. (k - номер муравья в субпопуляции A_τ).
11. Выполняется *Встроенный алгоритм поведения муравья*.
Агент $a_{tk} \in A_\tau$, в соответствии со стратегией C_τ , строит на графе $R_\tau^*(F, E)$ маршрут S_{tk} из стартовой вершины O_τ . Рассчитывается оценка ζ_{tk} маршрута S_{tk} . Маршрут S_{tk} заносится в базу BS_τ .
12. Если $k < n_a$, то $k = k + 1$ и переход к пункту 11, иначе – к пункту 13.
13. Если $\tau < 4$, то $\tau = \tau + 1$ и переход к пункту 10, иначе – к пункту 14.
14. В каждой базе BS_τ выбирается группа лучших маршрутов A_τ .
15. Посредством трансформации для каждой группы маршрутов A_τ формируются группы изоморфных маршрутов A_τ^* . Для A_1 формируются A_1^* , A_1^* , A_1^* . Для A_2 формируются A_2^* , A_2^* , A_2^* . Для A_3 формируются A_3^* , A_3^* , A_3^* . Для A_4 формируются A_4^* , A_4^* , A_4^* .
16. Каждая база построенных маршрутов BS_τ дополняется, путем включения в нее групп изоморфных маршрутов A_τ^* :
в BS_1 включаются A_2^* , A_3^* , A_4^* ;
в BS_2 включаются A_1^* , A_3^* , A_4^* ;
в BS_3 включаются A_1^* , A_2^* , A_4^* ;
в BS_4 включаются A_1^* , A_2^* , A_3^* .

2-й этап итерации

17. В каждом ГПР $R_\tau^*(F, E)$ на ребрах, соответствующих ребрам маршрутов S_{tk} , хранящихся в базе BS_τ , откладывается феромон в количестве $\delta_{tk} = \Delta / \zeta_{tk}$.

3-й этап итерации

18. На ребрах всех графов $R_\tau^*(F, E)$ в соответствии с формулой (6.5) производится испарение феромона.
19. Выбор лучшего решения, полученного на протяжении всех выполненных итераций.
20. Если все итерации выполнены, то конец работы алгоритма, в противном случае $l = l + 1$ и переход к пункту 8.

Встроенный алгоритм построения маршрута муравьем при трассировке

1. Выбранный агент a_{tk} для построения маршрута в графе $R_\tau^*(F, E)$ помещается в заданную стартовую вершину O_τ . $S_{tk}(0) = \emptyset$.
2. $t = 1$. (t -номер шага). Множество свободных вершин $F_{2tk}(t) = F$, Текущая магистраль $m(t) = 1$.
3. Определяется последний размещенный фрагмент f_z маршрута $S_{tk}(t)$.

4. Агент просматривает множество всех свободных на данном шаге вершин $F2_{tk}(t)$ и выделяет из него подмножество вершин $Z2_{tk}(t)$, удовлетворяющих *условиям размещения* фрагментов в текущей магистрали $m(t)$.

5. Если $Z2_{tk}(t) = \emptyset$, т.е. ни один из еще не размещенных фрагментов не может быть размещен в текущей магистрали, то $m(t) = m(t) + 1$ и переход к пункту 4, иначе переход к пункту 6.

6. Для каждой вершины $f_i \in Z2_{tk}(t)$ рассчитываются три параметра: $h_i(l-1)$, d_i , w_i .

7. При мультипликативной свертке по формуле (2), либо при аддитивной свертке по формуле (3) определяется интегральная стоимость $Q_{ic}(t)$ связи каждой вершины $f_i \in Z2_{tk}(t)$ с $S_{tk}(t)$.

8. Для каждой вершины $f_i \in Z2_{tk}(t)$ по формуле (4) рассчитывается вероятность $P_i(t)$ её включения в прокладываемый маршрут $S_{tk}(t)$.

9. Агент с вероятностью $P_i(t)$ выбирает одну из вершин $f_i \in P_i(t)$, которая включается в маршрут $S_{tk}(t)$, маршрут уже обозначается как $S_{tk}(t+1)$, а f_i исключается из $F2_{tk}(t)$.

10. Если $F2_{tk}(t) = \emptyset$, т.е. все фрагменты уложены по магистралям, то переход к п. 11, иначе $t = t + 1$ и переход к п. 3.

11. Рассчитывается оценка ζ_{tk} маршрута S_{tk} , полученного агентом a_{tk} на l -ой итерации.

12. Конец работы *встроенного алгоритма*.

Временная сложность этого алгоритма зависит от времени жизни колонии l (число итераций), количества вершин графа n и числа муравьев m .

Экспериментальные исследования. Для проведения экспериментов была использована процедура синтеза контрольных примеров с известным оптимумом по аналогии с известным методом PEKU (Placement Examples with Known Upper bounds on wirelength) [12].

Для задачи трассировки на канале созданы четыре набора примеров с известным оптимумом по ширине канала – F_{opt} . Для генерации примеров использовался следующий метод: на вход подается число контактов и число цепей. Затем генерируется пример трассировки в канале, который имеет заранее известное оптимальное значение F_{opt} ширины канала (числа магистралей). Набор PEKU имеется в обоих форматах GSRC BookShelf и LEF/DEF, и они доступны в сети [13]. Наборы G-PEKU и PEKU также доступны в сети [13]. Оптимизация проводилась по критерию минимальной ширины канала и опиралась на методику, изложенную в работе [1].

Первая часть исследований посвящена исследованию на одной популяции механизмов муравьиной колонии для задачи канальной трассировки. Для этого исследовалось влияние управляющих параметров, таких как: объем популяции, количество итераций, начальное количество феромона, откладываемого на ребрах графов и т.д. В результате определялось такое сочетание значений этих параметров, которое обеспечивает наивысшую эффективность адаптивных процедур для задачи канальной трассировки.

Во второй части исследовался ко-алгоритм на четырех субпопуляциях.

Эксперименты показали, что начальное количество феромона Ω должно быть в 10–12 раз больше среднего количества значения феромона $\tau_k(l)$, откладываемого муравьями на каждой итерации. На основе обработки экспериментальных исследований была построена средняя зависимость качества решений от числа итераций (рис. 5). Оценкой качества служит величина F_{opt}/F , где F – оценка полученного решения. Число итераций, при которых муравьиный алгоритм МА находил лучшее решение лежит в пределах 70–80, а ко-алгоритм – в пределах 60–65 (рис. 5).

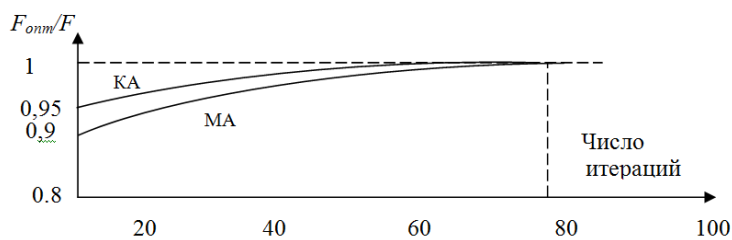


Рис. 5. Зависимость качества решений от числа итераций

Исследования показали, что размер популяции, при котором муравьиный алгоритм МА находил лучшее решение лежит в пределах 100, а размер субпопуляции в ко-алгоритме, лежит в пределах 80 (рис. 6).

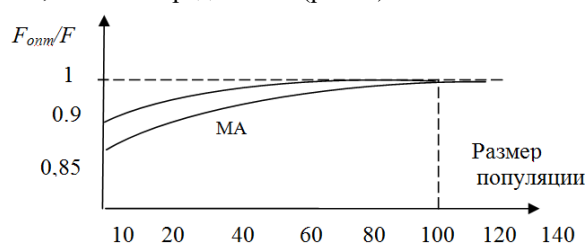


Рис. 6. Зависимость качества решений от размера популяции

Для сравнения качества решений задачи канальной трассировки использовались стандартные тесты Ex1, Ex3b, Ex3c, Ex4b, Ex5. Результаты работы ко-алгоритма КА сравнивались с результатами работы «эффективного» алгоритма «ЕА» [14], эволюционного алгоритма «ЕВ» [15] и генетического алгоритма «GA» [16–20]. Минимальное число магистралей, найденное различными алгоритмами, приведено в табл. 1.

Таблица 1

Результаты сравнения

Тест	ЕА	ЕВ	GA	КА
Ex1	12	13	13	12
Ex3b	18	18	17	17
Ex3c	18	18	18	18
Ex4b	17	17	17	17
Ex5	20	21	21	19

В колонке "КА" показаны результаты работы гибридного роевого алгоритма канальной трассировки (рассматриваемый алгоритм).

Из табл. 1 видно, что результаты разработанного ко-алгоритма (КА) на 3–5 % лучше аналогов.

Заключение. В работе для решения задачи канальной трассировки (ЗКТ) предложен ко-эволюционный алгоритм, основанный на методе муравьиной колонии. Ко-алгоритм предполагает параллельное функционирование заданного числа субалгоритмов муравьиной колонии, которые используют различные, но изоморфные стратегии поиска. Предложенный подход позволяет организовать систему коллективной адаптации с высокой степенью целесообразного поведения и сходимости. Ключевая проблема, которая была решена в данной работе, связана с разработкой принципов взаимодействия субпопуляций, отличающиеся стратегиями поиска интерпретаций

решений в изоморфных средах функционирования. Периодически агенты мигрируют из одной субпопуляции в другую, передавая свой опыт. В качестве интерпретации решения ЗКТ используется список. Список декодируется в геометрическое представление решения (эскиз) только с использованием соответствующей ему стратегии. Основная цель сводится к нахождению такого списка, для которого с помощью последовательной процедуры декодирования фрагменты цепей размещаются в минимальном числе магистралей ξ_{omm} , т.е. $\forall p(\xi_p \geq \xi_{omm})$.

Для построения конструктивной (встроенной) процедуры поиска решения муравьем используется модифицированный алгоритм левого конца (АЛК). Для усиления сходимости алгоритма и способности выхода из локальных оптимумов разработана комбинированная оценка, характеризующая преимущество выбора данной вершины, для включения ее в формируемый маршрут. В основу формул расчета вероятности включения вершины в формируемый маршрут положены эвристические соображения о предпочтениях того, что вершина входит в состав оптимального маршрута.

Ко-эволюционный подход обеспечивает более широкий обзор пространства решений и более высокую вероятность локализации глобального экстремума задачи. По сравнению с существующими алгоритмами достигнуто улучшение качества решений до 5 %.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Лебедев Б.К.* Интеллектуальные процедуры синтеза топологии СБИС. – Таганрог: Изд-во ТРТУ, 2003. – 96 с.
2. *Sarrafzadeh M. and Wong C.K.* An Introduction to VLSI Physical Design. – New York: McGraw Hill, 1996. – 198 p.
3. *Деньдобренко Б.П., Малика А.С.* Автоматизация проектирования радиоэлектронной аппаратуры. – М.: Высш. шк., 2002. – 156 с.
4. *Карпенко А.П.* Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учеб. пособие. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2014. – 448 с.
5. *Dorigo M. and Stützle T.* Ant Colony Optimization. MIT Press, Cambridge, MA, 2004. – 187 p.
6. *Лебедев О.Б.* Модели адаптивного поведения муравьиной колонии в задачах проектирования. – Таганрог: Изд-во ЮФУ, 2013. – 199 с.
7. *Wang X.* Hybrid nature-inspired computation method for optimization: Doctoral Dissertation. Helsinki University of Technology, TKK Dissertations, Espoo, 2009. – 161 p.
8. *Лебедев Б.К., Лебедев О.Б., Лебедев В.Б.* Гибридизация роевого интеллекта и генетической эволюции на примере размещения // Программные продукты, системы и алгоритмы. – 2017. – № 4.
9. *Воробьева Е.Ю., Карпенко А.П., Селиверстов Е.Ю.* Ко-гибридизация алгоритмов роя частиц // Наука и образование. – 2012. – № 4. – Режим доступа: <http://www.technomag.edu.ru/doc/355729.html>.
10. *Лебедев О.Б.* Трассировка в канале методом муравьиной колонии // Известия ЮФУ. Технические науки. – 2009. – № 2 (91). – С. 46-52.
11. *Лебедев В.Б.* Построение кратчайших связывающих сетей на основе роевого интеллекта // Известия ЮФУ. Технические науки. – 2011. – № 7 (120). – С. 37-44.
12. *Cong J., Romesis M., and Xie M.* Optimality, Scalability and Stability Study of Partitioning and Placement Algorithms // Proc. of the International Symposium on Physical Design, Monterey, CA. – 2003. – P. 88-94.
13. MCNC Electronic and Information Technologies (Online). Available: www.mcnc.org.
14. *Yoshimura T. and Kuh E.S.* Efficient algorithms for channel routing // IEEE Trans. Computer Aided Design Integrated Circuits & Syst. – 1982. – Vol. 1, No. 1. – P. 25-35.
15. *Yan T. and Wong M.D.F.* BSG-Route: A Length-Constrained Routing Scheme for General Planar Topology // In Proc. Int. Conf. Comp.-Aided Des. – 2008. – P. 499-505.
16. *Кныш Д.С., Курейчик В.М.* Генетический алгоритм трассировки коммутационных блоков // Известия вузов. Электроника. Схемотехника и проектирование. – 2009. – № 5 (79). – С. 28-34.

17. Лебедев Б.К., Лебедев О.Б., Лебедева Е.О. Модифицированный муравьиный алгоритм построения минимального дерева Штейнера // Известия ЮФУ. Технические науки. – 2017. – № 7 (192). – С. 38-53.
18. Лебедев О.Б., Пурчина О.А. Роевой алгоритм трассировки в приканальной надъязыечной области // Известия ЮФУ. Технические науки. – 2015. – № 6 (167). – С. 41-51.
19. Kureichik V.M., Lebedev B.K., Lebedev O.B. Channel Routing Based on Ant Colony Adaptive Behavior Model // Journal of Computer and Systems Sciences International. – 2015. – Vol. 54, No. 2. – P. 278-293. ISSN 10642307.
20. Лебедев О.Б., Лебедева Е.М., Орлов А.Н. Решение задачи трассировки с помощью ортогональных деревьев Штейнера // Информатика, вычислительная техника и инженерное образование. – 2014. – № 2 (17).

REFERENCES

1. Lebedev B.K. *Intellectual'nye protsedury sinteza topologii SBIS [Intelligent synthesis procedures for VLSI topology]*. Taganrog: Izd-vo TRTU, 2003, 96 p.
2. Sarrafzadeh M. and Wong C.K. *An Introduction to VLSI Physical Design*. New York: McGraw Hill, 1996, 198 p.
3. Den'dobrenko B.P., Malika A.S. *Avtomatizatsiya proektirovaniya radioelektronnoy apparatury [Automation design of electronic equipment]*. Moscow: Vyssh. shk., 2002, 156 p.
4. Karpenko A.P. *Sovremennyye algoritmy poiskovoy optimizatsii. Algoritmy, vdokhnovlennyye prirodoy: ucheb. posobie [Modern search engine optimization algorithms. Algorithms inspired by nature: tutorial]*. Modcow: Izd-vo MGTU im. N.E. Baumana, 2014, 448 p.
5. Dorigo M. and Stützle T. *Ant Colony Optimization*. MIT Press, Cambridge, MA, 2004, 187 p.
6. Lebedev O.B. *Modeli adaptivnogo povedeniya murav'inoy kolonii v zadachakh proektirovaniya [Models of adaptive behavior of the ant colony in design problems]*. Taganrog: Izd-vo YuFU, 2013, 199 p.
7. Wang X. *Hybrid nature-inspired computation method for optimization: Doctoral Dissertation*. Helsinki University of Technology, TKK Dissertations, Espoo, 2009, 161 p.
8. Lebedev B.K., Lebedev O.B., Lebedev V.B. *Gibridizatsiya roevogo intellekta i geneticheskoy evolyutsii na primere razmeshcheniya [Hybridization of Swarm Intelligence and Genetic Evolution on the Example of Placement]*, *Programmnye produkty, sistemy i algoritmy [Software, systems and algorithms]*, 2017, No. 4.
9. Vorob'eva E.Yu., Karpenko A.P., Seliverstov E.Yu. *Ko-gibridizatsiya algoritmov roya chastits [Co-hybridization of particle swarm algorithms]*, *Nauka i obrazovanie [Science and Education]*, 2012, No. 4. Available at: <http://www.technomag.edu.ru/doc/355729.html>.
10. Lebedev O.B. *Trassirovka v kanale metodom murav'inoy kolonii [Channel tracing using the ant colony method]*, *Izvestiya YuFU. Tekhnicheskie nauki [Izvestiya SFedU. Engineering Sciences]*, 2009, No. 2 (91), pp. 46-52.
11. Lebedev V.B. *Postroenie krachayshikh svyazyvayushchikh setey na osnove roevogo intellekta [Construction of the shortest connecting networks based on swarm intelligence]*, *Izvestiya YuFU. Tekhnicheskie nauki [Izvestiya SFedU. Engineering Sciences]*, 2011, No. 7 (120), pp. 37-44.
12. Cong J., Romesis M., and Xie M. *Optimality, Scalability and Stability Study of Partitioning and Placement Algorithms*, *Proc. of the International Symposium on Physical Design, Monterey, CA*, 2003, pp. 88-94.
13. MCNC Electronic and Information Technologies (Online). Available: www.mcnc.org.
14. Yoshimura T. and Kuh E.S. *Efficient algorithms for channel routing*, *IEEE Trans. Computer Aided Design Integrated Circuits & Syst.*, 1982, Vol. 1, No. 1, pp. 25-35.
15. Yan T. and Wong M.D.F. *BSG-Route: A Length-Constrained Routing Scheme for General Planar Topology*, *In Proc. Int. Conf. Comp.-Aided Des.*, 2008, pp. 499-505.
16. Knysh D.S., Kureychik V.M. *Geneticheskyy algoritm trassirovki kommutatsionnykh blokov [Genetic switching unit tracing algorithm]*, *Izvestiya vuzov. Elektronika. Skhemitekhnika i proektirovanie [News of universities]*, 2009, No. 5 (79), pp. 28-34.
17. Lebedev B.K., Lebedev O.B., Lebedeva E.O. *Modifitsirovannyy murav'inyy algoritm postroeniya minimal'nogo dereva Shteynera [A modified ant algorithm for constructing a minimal Steiner tree]*, *Izvestiya YuFU. Tekhnicheskie nauki [Izvestiya SFedU. Engineering Sciences]*, 2017, No. 7 (192), pp. 38-53.

18. *Lebedev O.B., Purchina O.A.* Roevoy algoritm trassirovki v prikanal'noy nad'yacheechnoy oblasti [Swarm tracing algorithm in the over-the-cell region], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2015, No. 6 (167), pp. 41-51.
19. *Kureichik V.M., Lebedev B.K., Lebedev O.B.* Channel Routing Based on Ant Colony Adaptive Behavior Model, *Journal of Computer and Systems Sciences International*, 2015, Vol. 54, No. 2, pp. 278-293. ISSN 10642307.
20. *Lebedev O.B., Lebedeva E.M., Orlov A.N.* Reshenie zadachi trassirovki s pomoshch'yu ortogonal'nykh derev'ev Shteynera [Solving the trace problem using Steiner orthogonal trees], *Informatika, vychislitel'naya tekhnika i inzhenernoe obrazovanie* [Computer science, computer engineering and engineering education], 2014, No. 2 (17).

Статью рекомендовал к опубликованию д.т.н., профессор А.Г. Коробейников.

Лебедев Борис Константинович – Южный федеральный университет; e-mail: lebedev.b.k@gmail.com; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 89282897933; кафедра систем автоматизированного проектирования; профессор.

Лебедев Олег Борисович – e-mail: lebedev.ob@mail.ru; тел.: 89085135512; кафедра систем автоматизированного проектирования; доцент.

Лебедева Екатерина Олеговна – e-mail: lbedevakate@mail.ru; тел.: 89289591426; кафедра систем автоматизированного проектирования; магистрант.

Lebedev Boris Konstantinovich – Southern Federal University; e-mail: lebedev.b.k@gmail.com; 44, Nekrasovsky, Taganrog, 347928, Russia; phone: +7928-2897933; the department of computer aided design; professor.

Lebedev Oleg Borisovich – e-mail: lebedev.ob@mail.ru; phone: +79085135512; the department of computer aided design; associate professor.

Lebedeva Ekaterina Olegovna – e-mail: lbedevakate@mail.ru; phone: +79289591426; the department of computer aided design; master student.