

11. *Beasley J.* A genetic algorithm for the multidimensional knapsack problem, *Journal of heuristics*, 1998, No. 4, pp. 63-86.
12. *Merz P.* Greedy and local search heuristics for the unconstrained binary quadratic programming problem, *Journal of heuristics*, 2002, No. 2, pp. 197-213.
13. *Krasnogor N., Smith J.* A memetic algorithm with selfadaptive local search: TSP as a case study, *Proc. of the Genetic and Evolutionary Computation Conference (GECCO-2000)*. Las Vegas, USA: Morgan Kaufmann, pp. 987-994.
14. *Kureichik V.V., Kravchenko Y.A.* Bioinspired algorithm applied to solve the travelling salesman problem, *World Applied Sciences Journal*, 2013, Vol. 22, No. 12, pp. 1789-1797.
15. *Topchy A., Lebedko O., Miagkikh V.* Fast learning in multilayered networks by means of hybrid evolutionary and gradient algorithms, *Proc. of int. conf. on evolutionary computation and its applications*, 1996, pp. 390-398.
16. *Areibi S.* An integrated genetic algorithm with dynamic hill climbing for VLSI circuit partitioning, *Proc. of int. conf. on DM&EA*. 2000, pp. 97-102.
17. *Kureichik V.M., Rodzin S.I.* Evolutionary algorithms: genetic programming, *Journal of Computer and Systems Sciences International*, 2002, Vol. 41, No. 1, pp. 123-132.
18. *Bartenev A.S.* Obzor osnovnykh voprosov avtomatizirovannogo sostavleniya raspisaniya zanyatiy v vuze [An overview of the main issues of automated scheduling of classes in high school], *Sovremennye nauchnye issledovaniya i innovatsii* [Modern scientific research and innovation], 2011, № 5. Available at: <http://web.snauka.ru/issues/2011/09/2576> (accessed: 20 February 2014).
19. *Qu R. et al.* A survey of search methodologies and automated system development for examination timetabling, *Journal of scheduling*, 2009, No. 12, pp. 55-89.

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

**Родзин Сергей Иванович** – Южный федеральный университет; e-mail: srodzin@sfedu.ru; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 88634371673; кафедра МОП ЭВМ; профессор.

**Rodzin Sergey Ivanovich** – Southern Federal University; e-mail: srodzin@sfedu.ru; 44, Nekrasovskiy, Taganrog, 347928, Russia; phone: +78634371673; the department of software engineering; professor.

УДК 681.325

**В.А. Ванидовский, О.Б. Лебедев**

### **ДВУМЕРНАЯ УПАКОВКА В ПОЛУОГРАНИЧЕННУЮ ПОЛОСУ НА ОСНОВЕ МОДЕЛИРОВАНИЯ АДАПТИВНОГО ПОВЕДЕНИЯ МУРАВЬИНОЙ КОЛОНИИ\***

*Рассматривается задача двумерной упаковки в полуограниченную полосу (1.5 DBP). В качестве структуры данных, несущих информацию об упаковке, используется последовательность номеров прямоугольников, представляющую порядок их укладки. Существенную роль в получении решения играет декодер, осуществляющий укладку прямоугольников по заданным в нем правилам. Предложены новые способы решения задачи упаковки, использующие математические методы, в которых заложены принципы природных механизмов принятия решений. В качестве базовой структуры декодера выбрана эвристика Floor Ceiling No Rotation (FCNR). Для построения декодера и кодовой последовательности использованы модификации эвристики (FCNR) и метаэвристики, базирующиеся на моделировании адаптивного поведения муравьиной колонии. В отличие от канонической парадигмы муравьиного алгоритма муравьям на графе поиска решений  $G=(X,U)$  строятся маршруты с разбиением на части и формированием на вершинах, входящих в каждую часть, подграфов, на ребрах которых откладывается феромон. Описывается структура графа*

\* Работа выполнена при частичной поддержке РФФИ (проект № 12-01-00100).

поиска решений, процедура поиска решений на графе, способы отложения и испарения феромона. В работе используется циклический (ant-cycle) метод муравьиных систем. Экспериментальные исследования проводились на IBM PC. Временная сложность алгоритма (BCA), полученная экспериментальным путем, практически совпадает с теоретическими исследованиями и для рассмотренных тестовых задач составляет ( $BCA \approx O(n^2)$ ). По сравнению с существующими алгоритмами достигнуто улучшение результатов на 2–3 %.

Двумерная упаковка в полуограниченную полосу; роевой интеллект; муравьиная колония; адаптивное поведение; оптимизация.

V.A. Vanidovsky, O.B. Lebedev

## TWO-DIMENSIONAL STRIP PACKING BASED ON SIMULATION OF ADAPTIVE BEHAVIOR OF AN ANT COLONY

*The problem of two-dimensional Strip Packing (1.5 DBP). As a data structure, carrying information about the package, the sequence of numbers used rectangles representing the order of their placement. An important role in obtaining solutions plays a decoder performing stacking rectangles rules laid down in it. New mechanisms for solving the problem of packaging, using mathematical methods, which lays down the principles of the natural mechanisms of decision-making. In this paper, as the basic structure of the decoder selected heuristics Floor Sealing No Rotation (FCNR). To build the decoder and the code sequence used modification heuristics (FCNR) and metaheuristics based on simulation of adaptive behavior of an ant colony. Unlike the canonical paradigm ant algorithm to find solutions to the graph  $G = (X, U)$  is constructed with a partition on the route of the formation and on the tops within each part, subgraphs whose edges are delayed pheromone. The structure of the graph search for solutions, the procedure of finding solutions on the graph ways pheromone deposition and evaporation. We use a cyclic (ant-cycle) method of ant systems. Experimental studies were carried out on IBM PC. The time complexity of the algorithm (ICA), experimentally obtained, practically coincides with the theoretical studies and for the considered test problems is ( $ICA \approx O(n^2)$ ). Compared with existing algorithms to improve the results achieved by 2–3 %.*

*Two-dimensional strip packing; swarm intelligence; ant colony; adaptive behavior; optimization.*

**Введение.** Задачи раскроя-упаковки (Bin Packing) занимают важное место в современной комбинаторной оптимизации и привлекают внимание многих ученых, как в России, так и за рубежом [1, 2]. Интерес к задачам раскроя-упаковки объясняется, в частности, их большой практической значимостью. Как правило приложения задач раскроя-упаковки относятся к материалоемким производствам, где одним из основных факторов снижения себестоимости выпускаемой продукции является рациональное использование ресурсов.

Существует множество разновидностей этой задачи: двумерная упаковка (2-Dimensional Bin Packing), двумерная упаковка в полуограниченную полосу (2-Dimensional Strip Packing, 2DSP или 1.5 DBP), линейная упаковка, упаковка по весу, упаковка по стоимости и другие. Задачи упаковки могут применяться в области оптимального заполнения контейнеров, загрузки грузовиков с ограничением по весу, созданием резервных копий на съёмных носителях и т.д. На сегодняшний день для решения задач раскроя-упаковки используются различные оптимизационные алгоритмы [3–8].

В работе рассматривается задача упаковки полуограниченной полосы. Задан набор прямоугольников. Дан один большой объект (называемый полосой), чья ширина  $W$  дана, а высота  $H$  – искомое значение переменной. Цель состоит в том, чтобы минимизировать высоту  $H$  полосы, содержащей прямоугольники, помещенные в полосу без их взаимного перекрытия

Задачи раскроя-упаковки относятся к классу NP–трудных задач комбинаторной оптимизации. В связи с этим большое значение приобретает разработка и исследование итерационных методов решения задач раскроя-упаковки, в том числе, и методов локального поиска, хорошо зарекомендовавших себя.

В качестве структуры данных, несущих информацию об упаковке, чаще всего используется последовательность номеров прямоугольников, представляющая порядок их укладки. Существенную роль в общем процессе нахождения решения играет декодер, осуществляющий укладку прямоугольников по заложенным в нем правилам. Важной характеристикой декодера является его способность получить оптимальную упаковку по заранее известному оптимальному коду (приоритетному списку), т.е. способность правильно декодировать.

Как процесс работы декодера, так и процесс формирования последовательности используют алгоритмы с эвристическим и метаэвристическим методом решения для получения квазиоптимальных результатов. Данное представление удобно для его использования в различных метаэвристиках (генетические алгоритмы, муравьиные алгоритмы), поскольку они как раз и работают с последовательностями, кодирующими упаковку. Однако качество решения во многом зависит от способов формирования компонент метаэвристики.

Исходя из вышеизложенного, представляет интерес разработка и применение новых поисковых алгоритмов для решения задачи раскроя-упаковки прямоугольных предметов на базе эффективных метаэвристик – принципов кодирования упаковок. Результатом непрекращающегося поиска наиболее эффективных методов упаковки стало использование бионических методов и алгоритмов. Одним из новых направлений таких методов являются мультиагентные методы интеллектуальной оптимизации, базирующиеся на моделировании коллективного интеллекта [9–11]. К таким методам можно отнести, и муравьиные алгоритмы (Ant Colony Optimization – ACO) [12, 13]. Основу поведения муравьиной колонии составляет самоорганизация, обеспечивающая достижения общих целей колонии на основе низкоуровневого взаимодействия.

В работе излагается метод решения задачи упаковки, основанный на моделировании адаптивного поведения муравьиной колонии.

**Постановка задачи.** Задача прямоугольной упаковки в полосу (1.5 Dimensional Bin Packing, 1.5DBP) состоит в следующем. Исходная информация задается следующим набором данных:  $H$  – ширина полубесконечной полосы;  $R=\{r_i/i=1,2,\dots,m\}$  – набор прямоугольников, где  $m$  – количество прямоугольников;  $W=\{w_i/i=1,2,\dots,m\}$  – вектор ширин прямоугольников;  $L=\{l_i/i=1,2,\dots,m\}$  – вектор длин прямоугольников. Вводится прямоугольная система координат  $XOY$ , у которой оси  $OX$  и  $OY$  совпадают соответственно с нижней неограниченной и боковой сторонами полосы. Решение задачи представляется в виде набора элементов  $\langle X, Y \rangle$ , где  $X=\{x_i/i=1,2,\dots,m\}$ ,  $Y=\{y_i/i=1,2,\dots,m\}$  – векторы координат прямоугольников.  $(x_i, y_i)$  – координаты нижнего левого угла прямоугольника соответственно по оси  $OX$  и  $OY$ . Набор элементов  $\langle X, Y \rangle$  называется допустимой прямоугольной упаковкой, если выполнены следующие условия:

1. Стороны прямоугольников параллельны граням полосы.
2. Прямоугольники не перекрывают друг друга.
3. Прямоугольники не выходят за границы полосы.

Так как проблема упаковки  $NP$ -трудная, исследования, ведущиеся по этой проблеме, сфокусированы главным образом на разработке приближенных алгоритмов решения. Приближенные алгоритмы находят решение с определенной точностью, но не гарантируют оптимальной упаковки для любого набора данных. Большинство, разработанных к настоящему времени декодеров, основанных на эвристических правилах, делятся на три группы: уровневые (level), шельфовые (shelf), плоские (plane).

Плоские алгоритмы размещают прямоугольники строго вплотную друг к другу.

Шельфовые рассматривают сразу несколько шельфов (полок) и распределяют по ним прямоугольники.

Подход всех уровневых алгоритмов в том, что полоса разделяется на уровни, исходя из высоты имеющихся на данном этапе прямоугольников. Прямоугольники размещаются вдоль основания текущего уровня слева направо, пока это возможно; прямоугольник, который не поместился, упаковывается на следующий уровень. Высота каждого уровня определяется по самому высокому прямоугольнику в нем. Прямоугольники, расположенные на одном уровне, образуют ряд. Вот три варианта упаковки:

Next Fit Level – когда открывается следующий уровень, предыдущий «закрывается» и его больше не рассматриваем;

First Fit Level – на каждом шаге алгоритма просматривается *каждый* уровень, начиная с самого нижнего, и прямоугольник упаковывается на первый подходящий, на котором достаточно места;

Best Fit Level – на каждом шаге алгоритма просматриваются *все* уровни, и выбирается наиболее подходящий – тот, на котором после упаковки останется минимум места.

Модификацией алгоритма Best Fit Level является алгоритм Floor Ceiling No Rotation. Упаковка осуществляется блоками. Каждый блок ограничен двумя уровнями. Нижний уровень блока есть «пол» (floor) и верхний уровень блока – «потолок» (ceiling). Пока есть возможность, прямоугольники пакуются на «пол» слева направо (рис. 1). Когда место заканчивается, предпринимается попытка упаковать на «потолок» справа налево; если нет места на потолке, то только тогда начинается формирование нового блока. В соответствии с эвристикой BFDH, на каждом шаге просматриваются все уровни – сначала «пол», затем «потолок» – на наличие наиболее подходящего места. Метод удачно упаковывает по «потолкам» самые мелкие прямоугольники.

В работе в качестве базовой структуры декодера выбрана эвристика Floor Ceiling No Rotation (FCNR). Для построения декодера и кодовой последовательности использованы модификации эвристики (FCNR) и метаэвристики, базирующиеся на моделировании адаптивного поведения муравьиной колонии

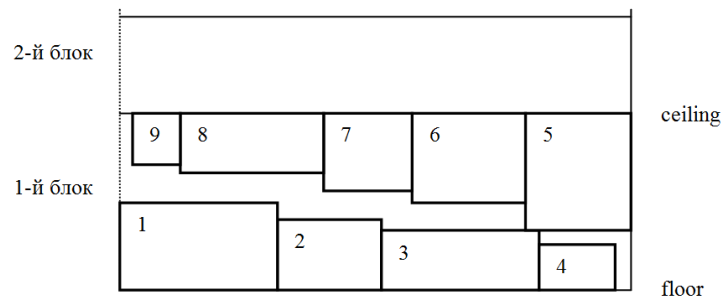


Рис. 1. Укладка в соответствии с эвристикой Floor Ceiling No Rotation

**Механизмы упаковки на основе муравьиной колонии.** Обычно заполнение уровней производится в соответствии с последовательностью, формируемой путем сортировки всего списка по не возрастанию высоты прямоугольников. В работе используется следующий подход к упаковке полубесконечной полосы. На первом этапе множество прямоугольников  $R = \{r_i | i = 1, 2, \dots, m\}$  разбивается на  $c$  подмножеств  $R_j$ , таким образом, чтобы суммарная ширина  $H_j$  прямоугольников, входящих в  $R_j$ , не превышала ширины  $H$  полосы.  $R = \cup R_j, (\forall i, j) [R_i \cap R_j = \emptyset], (\forall j) [R_j \neq \emptyset], (\forall j) [H_j \leq H]$ . Величина  $c$  и длина заполненной части полосы определя-

ется в результате разбиения [14]. На втором этапе прямоугольники каждого подмножества  $R_j$  сортируются по не возрастанию высоты. На третьем этапе производится последовательное заполнение блоков, начиная с самого нижнего. Нижние уровни блоков заполняются подмножествами  $R_j$  с нечетными номерами, верхние – с четными номерами. Заполнение уровней производится в соответствии с последовательностями, полученными на втором этапе. Уровни в каждом блоке сближаются на минимально допустимое расстояние, исключающее наложение прямоугольников. На заключительном этапе производится сжатие по оси  $Ox$ , и определяется размер заполнения заполненной части полосы –  $L$ .

В работе множество прямоугольников  $R = \{r_i | i = 1, 2, \dots, m\}$  разбивается на  $c$  подмножеств  $R_j$  методом муравьиной колонии

Большинство рассмотренных метаэвристических моделей адаптивного поведения муравьиной колонии опираются на парадигму поиска муравьем кратчайшего пути на графе. Однако для ряда задач весьма затруднительно или просто невозможно воспользоваться этой парадигмой.

В работе основное внимание сосредоточено на модификации канонической парадигмы муравьиного алгоритма для расширения сферы его применения. Пусть имеется некоторое решение задачи укладки  $m$  прямоугольников на  $c$  уровнях, задаваемое некоторой последовательностью  $R$ . Пусть  $m_i$  – число прямоугольников, укладываемых на  $i$ -м уровне. Поскольку при разбиении порядок, в котором будут уложены элементы множества  $R_i$ , не учитывается, то решению разбиения будет соответствовать число списков равно  $m_1! \cdot m_2! \cdot \dots \cdot m_i! \cdot \dots \cdot m_c!$ . В связи с этим при поиске решения актуальна проблема синтеза и просмотра только одного списка, соответствующего конкретному решению.

Поиск решений осуществляется на полном графе поиска решений  $G = (E, U)$ . Вершины множества  $E = \{e_i | i = 1, 2, \dots, m\}$  соответствуют элементам  $r_i$ . В общем случае поиск решения задачи упаковки осуществляется коллективом муравьев  $A = \{a_k | k = 1, 2, \dots, m\}$ . На каждой итерации муравьиного алгоритма каждый муравей  $a_k$  строит свое конкретное решение – маршрут в графе  $G$ , разбитый на части. Маршруты строятся муравьями по очереди один за другим. Другими словами муравьями независимо друг от друга строятся  $m$  маршрутов с использованием данных, накопленных на предыдущих итерациях. Вершинам каждой части маршрута соответствуют прямоугольники, укладываемые на одном уровне. Для равномерного распределения муравьев и создания равных стартовых условий в качестве начальных вершин у формируемых муравьями маршрутов с разбиением на подмножества, используются вершины множества  $E$  общим числом  $m$ .

Моделирование поведения муравьев в задаче упаковки связано с распределением феромона на ребрах графа  $G$ . На начальном этапе на всех ребрах графа  $G$  откладывается одинаковое (небольшое) количество феромона  $Q$ . Параметр  $Q$  задается априори. Процесс поиска решений итерационный. Каждая итерация  $s$  включает три этапа.

На первом этапе каждый муравей  $a_k$  находит решение (разбиение множества  $R$  на  $c$  подмножеств  $R_j$ ). Затем с помощью декодера производится укладка прямоугольников по уровням и определяется оценка решения  $L_k$ . На втором этапе муравьи откладывают феромон, на третьем этапе осуществляется общее испарение феромона. В работе используется циклический (ant-cycle) метод муравьиных систем. В этом случае феромон откладывается каждым агентом на ребрах графа после полного формирования решений всеми муравьями. На первом этапе каждой итерации каждый  $k$ -й муравей формирует свой собственный маршрут  $M_k$  с разбиением его на части  $M_{kj}$ . Процесс поиска решений итерационный. Каждая часть  $M_{kj} \in M_k$  включает вершины, соответствующие множеству прямоугольников  $R_j$ , размещаемым в ряду на уровне  $d_j$ . Процесс построения маршрута  $M_k$  с разбиением его на

части  $M_{kj}$  пошаговый. На каждом шаге  $t$  агент  $a_k$  применяет вероятностное правило выбора следующей вершины для включения ее в формируемый маршрут  $M_k(t)$ . Пусть на шаге  $t$  формируется часть  $M_{kj}(t)$ , являющаяся хвостом  $M_k(t)$ .  $M_{kj}(t) \in M_k(t)$ . Для этого выделяется множество вершин  $X_k(t) \in X$ , таких, что если  $x_i \in X_k(t)$ , то прямоугольник  $r_i \in R$ , соответствующий вершине  $x_i$ , может быть размещен в ряду на уровне  $d_j$  без переполнения, т.е. без превышения суммарной ширины  $H_j$  прямоугольников, входящих в  $R_j$ .

Пусть  $e_k(t)$  – множество вершин части  $M_{kj}(t)$  маршрута  $M_k(t)$ . Агент просматривает все вершины  $x_i \in X_k(t)$  и для каждой вершины рассчитывается параметр  $f_{ik}$  – суммарный уровень феромона на ребрах графа  $G$ , связывающих  $x_i$  с вершинами множества  $e_k(t)$ . Вероятность  $P_{ik}$  включения вершины  $x_i \in X_k(t)$  в часть  $M_{kj}(t)$  маршрута  $M_k(t)$  определяется следующим соотношением:

$$P_{ik} = f_{ik} / \sum_i f_{ik}. \quad (1)$$

Агент  $a_k$  с вероятностью  $P_{ik}$  выбирает одну из вершин  $x_i \in X_k(t)$ , которая включается в конец маршрута  $M_k(t)$ . После этого фиксируется включение прямоугольника  $r_i$ , соответствующего  $x_i \in X_k(t)$  в подмножество  $R_j$ . Если  $X_k(t)$  оказывается пустым, то выбирается не входящая в маршрут вершина, а соответствующий ей элемент помещается в следующее подмножество  $R_{j+1}$ .

Полученные путем разбиения подмножества  $R_j$  сортируются по не возрастающей высоте прямоугольников. С помощью декодера производится укладка прямоугольников по уровням и определяется оценка решения  $L_k(s)$ , полученного муравьем  $a_k$  на итерации  $s$ .

На втором этапе итерации, каждый муравей  $a_k$  откладывает феромон на ребрах  $s$  полных подграфов, каждый из которых построен на множестве вершин, входящих в состав одной из частей  $M_{kj}(t)$  маршрута  $M_k(t)$ . Количество феромона  $\Delta\tau_k(s)$ , откладываемое муравьем  $a_k$  на каждом ребре подграфов, построенных на  $s$ -й итерации, определяется следующим образом:

$$\Delta\tau_k(s) = \Phi / L_k(s), \quad (2)$$

где  $s$  – номер итерации,  $\Phi$  – общее количество феромона, откладываемое муравьем на ребрах полных подграфов  $G_{kj}$ ,  $L_k(l)$  – целевая функция для решения, полученного муравьем  $a_k$  на  $s$ -й итерации. Чем меньше  $L_k(l)$ , тем больше феромона откладывается на ребрах полных подграфов и, тем больше вероятность выбора этих ребер при построении маршрутов на следующей итерации.

После того, как каждый агент сформировал решение и отложил феромон, на третьем этапе происходит общее испарение феромона на ребрах полного графа  $G$  в соответствии с (3):

$$f_{ik} = f_{ik}(1 - \rho), \quad (3)$$

где  $\rho$  – коэффициент обновления.

После выполнения всех действий на итерации находится агент с лучшим решением, которое запоминается. Далее переход на следующую итерацию.

Временная сложность этого алгоритма зависит от времени жизни колонии  $l$  (число итераций), количества вершин графа  $n$  и числа муравьев  $m$ , и определяется как  $O(l * n^2 * m)$ .

Алгоритм одномерной упаковки на основе метода муравьиной колонии формулируется следующим образом:

1. В соответствии с исходными данными формируется полный граф поиска решений  $G=(E,U)$ .

2. Задается число агентов и указываются начальные вершины, в которые они помещаются, а так же задается значение параметра  $\Phi$  и число итераций –  $N_s$ .

3. На всех ребрах графа  $G$  откладывается начальное количество феромона.  $s=1$ .

4. На первом этапе  $s$ -й итерации **каждым агентом**  $a_k$ : на ГПР  $G$  находится маршрут  $M_k(s)$  с разбиением на части и соответствующее ему разбиение множества  $R$  на  $s$  подмножеств  $R_j$ ; производится сортировка всех  $R_j$  по не возрастанию высоты прямоугольников, с помощью декодера для каждого  $R_j$  производится укладка прямоугольников по уровням.

5. Для каждого решения задачи упаковки находится значение критерия  $L_k(s)$ .

6. В графе  $G$  на рёбрах полных подграфов  $G_{k_j}$ , каждый из которых построен на множестве вершин, входящих в состав одной из частей  $M_{k_j}(t)$  маршрута  $M_k(l)$ , откладывается феромон. Количество феромона, откладываемого каждым агентом, пропорционально  $L_k(l)$ .

7. Выполняется процедура испарения феромона на ребрах графа  $G$ .

8. Выбирается лучшее решение, полученное на протяжении всех выполненных итераций.

9. Если все итерации выполнены, то конец работы алгоритма, в противном случае – переход к пункту 4 для выполнения очередной итерации.

**Заключение.** Предложены новые механизмы решения задачи упаковки, использующие математические методы, в которых заложены принципы природных механизмов принятия решений. В отличие от канонической парадигмы муравьиного алгоритма муравьем на графе поиска решений  $G=(X,U)$  строится маршрут с разбиением на части и формированием на вершинах, входящих в каждую часть, подграфов, на ребрах которых откладывается феромон. Агент на графе решений  $G(E,U)$  совершает перемещение не от вершины к вершине, а от группы вершин к вершине. Экспериментальные исследования проводились на IBM PC. Временная сложность алгоритма (BCA), полученная экспериментальным путем, практически совпадает с теоретическими исследованиями и для рассмотренных тестовых задач составляет (BCA  $\approx O(n^2)$ ).

Для проведения объективных экспериментов были использованы известные тестовые задачи, представленные в литературе и Интернет. Задачи, на которых был протестирован разработанный алгоритм, доступны в библиотеке OR-объектов (<http://www.ms.ic.ac.uk/info.html>). Для составления достоверных выводов был проведен не один, а серия опытов-экспериментов. По сравнению с существующими алгоритмами достигнуто улучшение результатов на 2–3 %.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Bischoff E.E. and Wäscher G.* Cutting and packing. European Journal of Operational Research. – 1995. – No 84. – P. 503-505.
2. *Ross P., Marin-Blazquez J.G., Schulenburg, S. and Hart E.* Learning a Procedure That Can Solve Hard Bin-Packing Problems: A New GA-Based Approach to Hyper-heuristics, Proceeding of the Genetic and Evolutionary Computation Conference, GECCO 2003, Chicargo, Illinois, USA, 2003. – P. 1295-1306.
3. *Потарусов Р.В., Курейчик В.М.* Проблема одномерной упаковки элементов // Известия ТРТУ. – 2006. – № 8 (66). – С. 88-93.
4. *Курейчик В.М.* Алгоритмы одномерной упаковки элементов // Известия ЮФУ. Технические науки. – 2013. – № 7 (144). – С. 8-11.
5. *Валеева А.Ф., Аглиуллин М.Н.* Моделирование прямоугольной упаковки на базе метаэвристики муравьиной колонии // Межвуз. науч. сб. «Принятие решений в условиях неопределенности». – Уфа, 2005. – Вып. 2. – С. 55-63.
6. *Levine J. and Ducatelle F.* Ant Colony Optimization and Local Search for Bin Packing and Cutting Stock Problems. Centre for Intelligent Systems and their Applications, School of Informatics, University of Edinburgh, 2003.

7. Тимофеева О.П., Соколова Э.С., Милов К.В. Генетический алгоритм в оптимизации упаковки контейнеров // Труды НГТУ. Информатика и системы управления. – 2013. – № 4 (101). – С. 167-172.
8. Лебедев О.Б., Зорин В.Ю. Упаковка на основе метода муравьиной колонии // Известия ЮФУ. Технические науки. – 2010. – № 12 (113). – С. 25-30.
9. Лебедев В.Б., Лебедев О.Б. Роевой интеллект на основе интеграции моделей адаптивного поведения муравьиной и пчелиной колоний // Известия ЮФУ. Технические науки. – 2013. – № 7 (144). – С. 41-47.
10. Лебедев Б.К., Лебедев В.Б. Оптимизация методом кристаллизации россыпи альтернатив // Известия ЮФУ. Технические науки. – 2013. – № 7 (144). – С. 11-17.
11. Лебедев Б.К., Лебедев О.Б. Моделирование адаптивного поведения муравьиной колонии при поиске решений, интерпретируемых деревьями // Известия ЮФУ. Технические науки. – 2012. – № 7 (132). – С. 27-35.
12. Dorigo M. and Stützle T. Ant Colony Optimization. MIT Press, Cambridge, MA, 2004.
13. Лебедев О.Б. Модели адаптивного поведения муравьиной колонии в задачах проектирования. – Таганрог: Изд-во ЮФУ, 2013.
14. Курейчик В.М., Лебедев Б.К., Лебедев О.Б. Разбиение на основе моделирования адаптивного поведения биологических систем // Нейрокомпьютеры: разработка, применение. – 2010. – № 2. – С. 28-34.

#### REFERENCES

1. Bischoff E.E. and Wäscher G. Cutting and packing, *European Journal of Operational Research*, 1995, No. 84, pp. 503-505.
2. Ross P., Marin-Blazquez J.G., Schulenburg, S. and Hart E. Learning a Procedure That Can Solve Hard Bin-Packing Problems: A New GA-Based Approach to Hyper-heuristics, *Proceeding of the Genetic and Evolutionary Computation Conference, GECCO 2003, Chicargo, Illinois, USA, 2003*, pp. 1295-1306.
3. Potarusov R.V., Kureychik V.M. Problema odnomernoy upakovki elementov [The one-dimensional problem of packing items], *Izvestiya TRTU [Izvestiya TSURE]*, 2006, No. 8 (66), pp. 88-93.
4. Kureychik V.M. Algoritmy odnomernoy upakovki elementov [Algorithms of one-dimensional packing items], *Izvestiya YuFU. Tekhnicheskie nauki [Izvestiya SFedU. Engineering Sciences]*, 2013, No. 7 (144), pp. 8-11.
5. Valeeva A.F., Agliullin M.N. Modelirovanie pryamougolnoy upakovki na baze metaevristiki muravinoy kolonii [Modeling rectangular packing on the basis of metaheuristic ant colony], *Mezhvuzovskiy nauchnyy sbornik «Prinyatie resheniy v usloviyakh neopredelennosti» [Interuniversity collection of scientific works "problems of decision Making under uncertainty"]*. Ufa, 2005, Vol. 2, pp. 55-63.
6. Levine J. and Ducatelle F. Ant Colony Optimization and Local Search for Bin Packing and Cutting Stock Problems. Centre for Intelligent Systems and their Applications, School of Informatics, University of Edinburgh, 2003.
7. Timofeeva O.P., Sokolova Eh.S., Milov K.V. Geneticheskiy algoritm v optimizatsii upakovki konteynerov [Genetic algorithm optimization of packaging containers], *Trudy NGTU. Informatika i sistemy upravleniya [Proceedings of NSTU. Informatics and control systems]*, 2013, No. 4 (101), pp. 167-172.
8. Lebedev O.B., Zorin V.Yu. Upakovka na osnove metoda muravinoy kolonii [Packaging on the basis of ant colony], *Izvestiya YuFU. Tekhnicheskie nauki [Izvestiya SFedU. Engineering Sciences]*, 2010, No. 12 (113), pp. 25-30.
9. Lebedev V.B., Lebedev O.B. Roevoy intellekt na osnove integratsii modeley adaptivnogo povedeniya muravinoy i pchelinoy koloniy [Swarm intelligence-based integration models of adaptive behavior of ant and bee colonies], *Izvestiya YuFU. Tekhnicheskie nauki [Izvestiya SFedU. Engineering Sciences]*, 2013, No. 7 (144), pp. 41-47.
10. Lebedev B.K., Lebedev V.B. Optimizatsiya metodom kristallizatsii rossypi alternativ [Optimization method of crystallization placer alternatives], *Izvestiya YuFU. Tekhnicheskie nauki [Izvestiya SFedU. Engineering Sciences]*, 2013, No. 7 (144), pp. 11-17.



11. *Lebedev B.K., Lebedev O.B.* Modelirovanie adaptivnogo povedeniya muravinoy kolonii pri poiske resheniy, interpretiruemykh derevyami [Simulation of adaptive behavior ant colony to find solutions, interpreted trees], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2012, No. 7 (132), pp. 27-35.
12. *Dorigo M. and Stützle T.* Ant Colony Optimization. MIT Press, Cambridge, MA, 2004.
13. *Lebedev O.B.* Modeli adaptivnogo povedeniya muravinoy kolonii v zadachakh proektirovaniya [Models of adaptive behavior of ant colonies in the design tasks]. Taganrog: Izd-vo YuFU, 2013.
14. *Kureychik V.M., Lebedev B.K., Lebedev O.B.* Razbienie na osnove modelirovaniya adaptivnogo povedeniya biologicheskikh sistem [Split based on the simulation of adaptive behavior of biological systems], *Neurokompyutery: razrabotka, primeneniye* [Neurocomputers: design, application], 2010, No. 2, pp. 28-34.

Статью рекомендовал к опубликованию д.т.н., профессор Я.Е. Ромм.

**Ванидовский Владислав Андреевич** – Южный федеральный университет; e-mail: vanidovskiy.v.a@gmail.com; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 89508575638; кафедра систем автоматизированного проектирования; аспирант.

**Лебедев Олег Борисович** – e-mail: lebedev.ob@mail.ru; тел.: 88634371743; кафедра систем автоматизированного проектирования; доцент.

**Vanidovsky Vladislav Andreevich** – Southern Federal University; e-mail: vanidovskiy.v.a@gmail.com; 44, Nekrasovsky, Taganrog, 347928, Russia; phone: +79508575638; the department of computer aided design; postgraduate student.

**Lebedev Oleg Borisovich** – e-mail: lebedev.ob@mail.ru; phone: +78634371743; the department of computer aided design; associate professor.

УДК 681.325

**Б.К. Лебедев, В.Б. Лебедев**

### **ГЛОБАЛЬНАЯ ТРАССИРОВКА МЕТОДОМ КРИСТАЛЛИЗАЦИИ РОССЫПИ АЛЬТЕРНАТИВ\***

*Предлагается метод и алгоритм глобальной трассировки на основе новой, предложенной авторами, парадигмы коллективного интеллекта – кристаллизации россыпи альтернатив (КРА). Решение задачи глобальной трассировки осуществляется на основе комбинаторного подхода. Наряду с метаэвристиками, на которых построены роевые алгоритмы, используется метаэвристика, имеющая тенденцию к использованию альтернатив (вариантов компонентов) из наилучших найденных решений. Сущность комбинаторного алгоритма заключается в том, что для каждой связывающей сети формируется набор вариантов ее построения. Целью задачи является нахождение на заданном наборе таких вариантов, которые обеспечивают наилучшее решение. В процессе эволюционной коллективной адаптации методами дискриминантного анализа формируются оценки приспособленности альтернатив. Приспособленность альтернатив рассматривается как вероятность ее использования в формируемом решении. Совокупность данных об альтернативах и их оценках составляет россыпь альтернатив. Дискриминантный анализ альтернатив в процессе эволюционной коллективной адаптации назван по аналогии с процессами вычленения объектов (формирования кристаллов) кристаллизацией. Другими словами, в процессе эволюционной коллективной адаптации производится вычленение из множества вариантов наиболее приспособленных альтернатив. Отсюда название метода оптимизации – метод кристаллизации россыпи альтернатив (КРА), (Crystallization of alternatives field*

\* Работа выполнена при частичной поддержке РФФИ (проект № 13-01-00596).