УДК 004.896 DOI 10.18522/2311-3103-2023-4-109-125

Б.К. Лебедев, О.Б. Лебедев, М.А. Ганжур

УПАКОВКА В ПОЛУБЕСКОНЕЧНУЮ ПОЛОСУ НА ОСНОВЕ ДЕКОМПОЗИЦИИ И ГИБРИДИЗАЦИИ БИОИСПИРИРОВАННЫХ МЕТОДОВ

Объектом исследования является задача прямоугольной упаковки в полубесконечную полосу. Задан набор прямоугольников. Дан один большой объект (называемый полосой), чья ширина D задана, а HP высота – искомое значение переменной. Цель состоит в том, чтобы минимизировать высоту HP полосы, содержащей прямоугольники, помещенные в полосу без их взаимного перекрытия. Для решения задачи упаковки предложен новый гибридный подход на основе декомпозиции общей задачи упаковки и гибридизации биоиспирированных методов, а также новый гибридный подход к декомпозиции общей задачи упаковки. Разработаны новые архитектура и методы решения задачи упаковки, построенные на основе декомпозиции и гибридизации разработанных авторами роевых методов, использующие различные стратегии поиска, функционирующие параллельно-последовательно и реализующие более широкий обзор пространства решений, что позволяет обеспечить более высокую вероятность локализации глобального экстремума за приемлемое время. Разработана методология нового направления поиска решений задач ортогональной упаковки на основе моделей адаптивного поведения биологических систем. Разработан высокоэффективный гибридный биоинспирированный метод решения задач одномерной и прямоугольной упаковки, основанный на декомпозиции задачи на множество подзадач и интеграции методов поисковой оптимизации. Предложены новые механизмы решения задачи упаковки, использующие математические методы, в которых заложены принципы природных механизмов принятия решений. В отличие от канонической парадигмы муравьиного алгоритма агентом на графе поиска решений в качестве решения формируется разбиение множества прямоугольных элементов A на подмножества A^k_i, где A^k_j – подмножество элементов, назначенных агентом в блок. Разработаны поисковые методы для решения задач гильотинного и не гильотинного прямоугольного раскроя. Для проведения объективных экспериментов были использованы известные тестовые задачи, представленные в литературе и Интернет. Получены лучшие результаты по сравнению с тестируемыми методами. Предложенные в работе теоретические положения для решения задач упаковки и раскроя промышленных объектов в условиях единичного производства реализованы в виде методик, алгоритмов и прикладного программного обеспечения. По сравнению с существующими алгоритмами достигнуто улучшение результатов на 3–5%. Временная сложность алгоритма, полученная экспериментальным путем, практически совпадает с теоретическими исследованиями и для рассмотренных тестовых задач составляет $O(n^2)$.

Упаковка в полубесконечную полосу; декомпозиция; одномерная упаковка; роевой интеллект; муравьиная колония; адаптивное поведение; гибридизация; поисковая оптимизация.

B.K. Lebedev, O.B. Lebedev, M.A. Ganzhur

SEMI-INFINITE STRIP PACKAGING BASED ON DECOMPOSITION AND HYBRIDIZATION OF BIOINSPIRED METHODS

In this work, the object of study is the problem of rectangular packing in a semi-infinite strip. Given a set of rectangles. Given one large object (called a strip), whose width D is given, and whose height HP is the desired value of the variable. The goal is to minimize the HP height of a strip containing rectangles placed in the strip without overlapping each other. To solve the packaging problem, a new hybrid approach is proposed based on the decomposition of the general packaging problem and hybridization of bioinspired methods, as well as a new hybrid approach to the decomposition of the general packaging problem. New architecture and methods for solving the packing problem have been developed, built on the basis of decomposition and hybridization of swarm methods developed by the authors, using various search strategies, operating in parallel *and sequentially and implementing a wider overview of the solution space, which allows for a higher probability of localizing a global extremum in an acceptable time. A methodology has been developed for a new direction in searching for solutions to orthogonal packing problems based on models of adaptive behavior of biological systems. A highly effective hybrid bioinspired method for solving one-dimensional and rectangular packaging problems has been developed, based on the decomposition of the problem into many subtasks and the integration of search optimization methods. New mechanisms for solving the packaging problem are proposed, using mathematical methods that incorporate the principles of natural decision-making mechanisms. In contrast to the canonical paradigm of the ant algorithm, the agent forms a partition of the set of rectangular elements A into subsets A_{ki} on the solution search graph as a solution, where A_{kj} is a subset of elements assigned by the agent to the block. Search methods have been developed for solving problems of guillotine and non-guillotine rectangular cutting. To conduct objective experiments, well-known test tasks presented in the literature and the Internet were used. Better results were obtained compared to the tested methods. The theoretical principles proposed in the work for solving problems of packaging and cutting industrial objects in single production conditions are implemented in the form of methods, algorithms and application software. Compared to existing algorithms, a 3-5% improvement in results was achieved. The time complexity of the algorithm, obtained experimentally, practically coincides with theoretical studies and for the considered test problems is $O(n^2)$.*

Semi-infinite strip packing; decomposition; one-dimensional packing; swarm intelligence; ant colony; adaptive behavior; hybridization; search engine optimization.

Введение. Объектом исследования является задача прямоугольной упаковки в полубесконечную полосу (1.5 Dimensional Bin Packing, 1.5 DBP). Задан набор прямоугольников. Дан один большой объект (называемый полосой), чья ширина D задана, а H^P высота – искомое значение переменной. Цель состоит в том, чтобы минимизировать высоту H^P полосы, содержащей прямоугольники, помещенные в полосу без их взаимного перекрытия [1-3].

У 1.5 DBP есть два варианта входных данных: когда набор упаковываемых объектов известен заранее (*offline*-проблема) и когда данные поступают порциями (*online*-проблема) [4, 5].

В данной статье предложен метод для решения задачи *offline*-варианта 1.5 DBP. большой размерности. Этот метод превосходит многие существующие алгоритмы по эффективности и временной производительности и почти всегда позволяет получать оптимальное или близкое к нему решение.

Задачи раскроя и упаковки, ориентированные на единичное производство, относятся к классу *NP*–трудных задач комбинаторной оптимизации, т.е. для их решения нет методов и алгоритмов, находящих точное решение за полиномиальное время [6–8].

Существующие точные методы решения задач упаковки и раскроя основаны на схеме полного перебора, поэтому они оказываются малопригодными для решения задач, встречающихся на практике [9–10]. В связи с *NP*–трудностью задач раскроя-упаковки одним из актуальных направлений исследований в настоящее время является разработка эффективных приближенных и эвристических методов. Приближенные алгоритмы находят оптимальное решение с определенной точностью, но не гарантируют оптимальной упаковки для любого набора данных [11, 12].

Результатом непрекращающегося поиска наиболее эффективных методов упаковки стало использование бионических методов и алгоритмов [13, 14]. Особенно интенсивно разрабатывается научное направление, объединяющее математические методы, в которых заложены принципы роевого интеллекта (Swarm Intelligence) [15], в которых совокупность простых агентов конструирует стратегию своего поведения без наличия глобального управления [16]. Среди них особенно активно развиваются методы роя частиц, муравьиной и пчелиной колонии [17].

Эффективность биоинспирированного поиска во многом определяется как учетом специфики решаемой задачи, так и использованием новых и модифицированных процедур поиска.

В настоящее время одним из путей, повышения эффективности методов решения задач глобального поиска, является разработка гибридных алгоритмов [6,18]. В гибридных алгоритмах преимущества одного алгоритма могут компенсировать недостатки другого. В работе излагается метод решения задачи одномерной упаковки на основе интеграции модифицированной модели адаптивного поведения муравьиной колонии с конструктивными эвристиками [19]. Предложена композитная архитектура многоагентной системы бионического поиска.

Разработка методов упаковки осуществлена на основе общего концептуального подхода, базирующегося на применении методов искусственного интеллекта, инспирированных природными системами.

Для решения задачи упаковки предложен новый гибридный подход на основе декомпозиции общей задачи и интеграции биоиспирированных методов.

1. Постановка задачи упаковки. Найти: упаковку прямоугольников в полубесконечную полосу с минимальной высотй занимаемой полосы. При этом заданы следующие параметры:

полубесконечная полоса P;

D – ширина полосы;

H^P – высота занятой части полосы;

$A=\{a_i|i=1,2,...,n\}$ – множество элементов (прямоугольников);

$W=\{w_i|i=1,2,...,n\}$ – вектор ширин прямоугольников;

$H=\{h_i|i=1,2,...,n\}$ – вектор высот 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 координат элементов $<X, Y>$, где $X=(x_1, x_2,\ldots, x_i,\ldots, x_m)$, $Y=(y_1, y_2,...,y_i,...,y_m)$ – векторы координат прямоугольников, (x_i, y_i) – координаты нижнего левого угла прямоугольника соответственно по оси X и Y.

Набор координат элементов $<X,Y>$ называется допустимой прямоугольной упаковкой, если выполнены следующие условия:

1. Стороны прямоугольников параллельны граням полосы (условие ортогональности).

2. Прямоугольники не перекрывают друг друга.

3. Прямоугольники не выходят за границы полосы.

При выполнении условий допустимости требуется найти такую упаковку полосы, для которой высота H^P ее занятой части достигает минимума.

Большинство, разработанных к настоящему времени алгоритмов для задачи упаковки в полосу, основанных на эвристических правилах, используют 2 основных подхода: плоские (*plane*); многоуровневые (*level*) [3, 6, 7].

Плоские алгоритмы размещают прямоугольники в пределах полосы строго вплотную друг к другу.

Декодер плоской упаковки «Нижний левый» (*BL*) представлен на рис. 1.

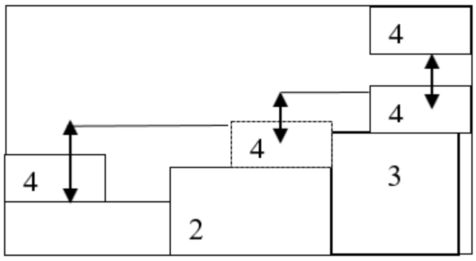

Рис. 1. Декодер плоской упаковки «Нижний левый»

На каждом шаге i: последовательно передвинуть прямоугольник a_i, начиная от верхнего правого угла упакованной площади полосы вниз настолько, насколько это возможно, а затем влево, снова вниз и т.д., до тех пор, пока его можно сдвинуть влево или вниз. Преимущество здесь имеет движение влево.

Основное отличие многоуровневыхуровневых алгоритмов упаковки заключается в том, что полоса представляется в виде многоуровневой иерархической структуры.

Для решения задачи прямоугольной упаковки в полубесконечную полосу предложен новый гибридный подход на основе декомпозиции общей задачи упаковки и построения поисковой процедуры на основе модифицированного метода муравьиной колонии.

Постулат **декомпозиции** применяется для снижения уровня сложности при решении задачи упаковки путем расчленения системы на подсистемы.

В работе используется следующий подход к упаковке полуограниченной полосы, базирующийся на многоуровневой парадигме. Упаковка полосы осуществляется контейнерами.

На первой стадии формируется множество прямоугольных контейнеров $Q=\{q_k|k=1,2,\ldots,n_j\}$ с упакованными в них элементами. На второй стадии осуществляется упаковка снизу вверх полуограниченной полосы контейнерами (рис. 2).

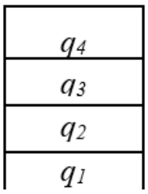

Рис. 2. Упаковка снизу вверх полуограниченной полосы контейнерами

Контейнер q_k представляет собой прямоугольник с шириной, равной ширине полосы – D. Высота контейнера q_k равна H^q_k.

Критерий оптимизации: $F =\sum H^q_{k}$., для $\{k|q_k \in Q\}$. Цель оптимизации – минимизация F.

Верхняя и нижняя стороны контейнера обозначаются как «потолок» и «пол» соответственно. Пол и потолок служат основаниями, на которых размещаются элементы. Элементы, расположенные на одной стороне контейнера между краями полосы, образуют блок. На полу контейнера расположен блок элементов «пол», на потолке контейнера – блок «потолок» (рис. 3).

Высота самого высокого прямоугольника в блоке является высотой блока.

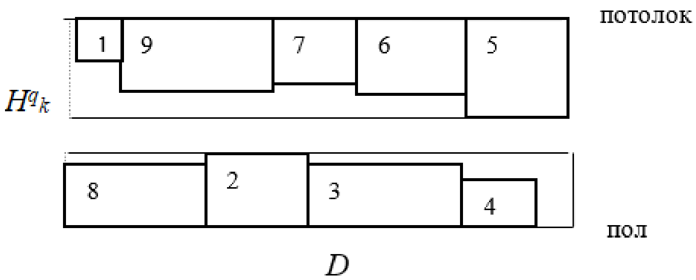

Рис. 3. Пример оснований («пол» и «потолок»), на которых размещаются элементы

Обычно упаковка контейнера элементами выполняется путем последовательного заполнения блоков, начиная с первого. В каждом из которых своя стратегия. Пока есть возможность, элементы пакуются на полу контейнера в блоке «пол» слева направо (рис. 2). Когда вместимость блока «пол» заканчивается, предпринимается попытка упаковки на потолке контейнера в блоке «потолок» справа налево. Если на потолке нет места, то только тогда начинается формирование блоков в новом контейнере.

В работе задача упаковки прямоугольников в полосу решается в три этапа (рис. 4).

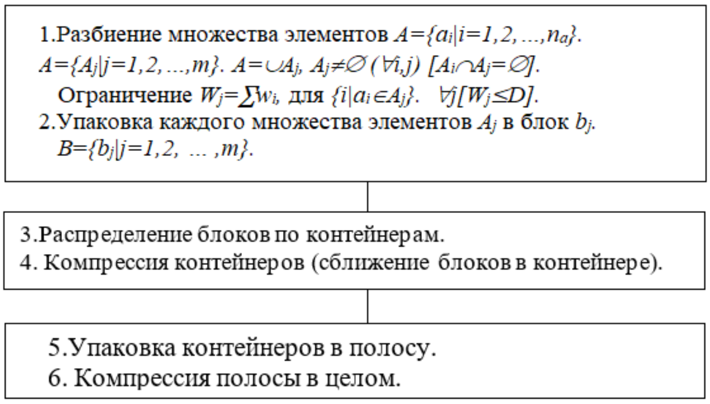

Рис. 4. Процесс решения задачи упаковки прямоугольников в полосу

На первом этапе агентом решается задача упаковки набора элементов прямоугольной формы $A=\{a_i|i=1,2,\dots,n_a\}$ в множестве блоков $B=\{b_j|j=1,2,\dots,m\}$.

Блоки пронумерованы. Каждый блок b_j включает подмножество A_j, элементов, вплотную расположенных на одном уровне (см. рис. 2). $A=\cup A_j$, $A_j\neq\varnothing$ $(\forall i,j)[A_i\cap A_j=\varnothing]$. В качестве веса элемента прямоугольной формы a_i $a_i\in A$ служит его ширина w_i.

Определяется суммарный вес W_j подмножества элементов A_j, назначенных в блок b_j.

$$W_j=\sum w_i,\ \text{для}\ \{i|a_i\in A_j\}.$$

Задается максимально допустимый суммарный вес D элементов, назначенных в каждый блок b_j. Вводится ограничение $W_j\leq D$.

Высота h^b_j каждого блока b_j определяется по самому высокому элементу в нем.

На втором этапе блоки помещаются в контейнеры. Блоки с нечетными номерами размещаются на «полу», а блоки с четными номерами размещаются на «потолке» контейнера.

На третьем этапе в каждом контейнере компрессионными алгоритмами блоки и элементы сближаются на минимально допустимое расстояние, исключающее наложение элементов. Возможны три варианта сближения в зависимости от типа разрезов полосы.

Состояние, при котором два прямоугольника соприкасаются друг с другом по вертикали, будем называть конфликтом.

Первый вариант сближения блоков реализуется при использовании гильотинного разреза ленты (рис. 5). В этом случае верхний и нижний блоки контейнера сближаются до их соприкосновения. Пусть h^b_1 и h^b_2 высоты верхнего и нижнего блоков, входящих в состав контейнера, соответственно. Тогда высота контейнера

q_k определится как $H^q_k=(h^b_1+h^b_2)$. Площадь контейнера q_k определится как $S_k=H^q_k \cdot D$. Пусть $S_э=\sum s_i$, для $\{i|a_i \in A_j\}$ – суммарная площадь элементов, входящих в состав контейнера q_k, где $s_i=w_i \cdot h_i$ – площадь элемента a_i. A_j – набор элементов, входящих в составе блока b_j. $B=\{b_j|j=1,2, \ldots ,m\}$.

Свободная площадь контейнера q_k, определится как $\delta_k=S_k - S_э$.

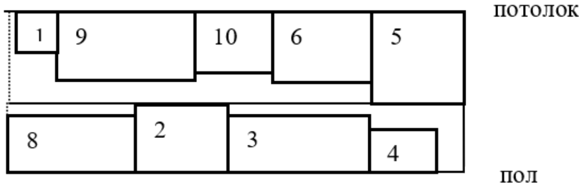

Рис. 5. Сближения блоков при использовании гильотинного разреза ленты

Второй вариант реализуется при использовании гильотинного разреза ленты на контейнеры и не гильотинного разреза контейнеров (методом штамповки). Сближение блоков контейнера выполняется до первого касания элементов по вертикали (рис. 6). Отметим, что положение элементов друг относительно друга по вертикали в обоих блоках зафиксировано. В результате сближения блоков отдельные прямоугольники верхнего уровня будут соприкасаться с прямоугольниками нижнего уровня.

На рис. 6 девятый элемент соприкасается с вторым, а пятый элемент с четвертым.

Высота контейнера $H^q_k = (h_9 + h_2) = (h_5 + h_4)$.

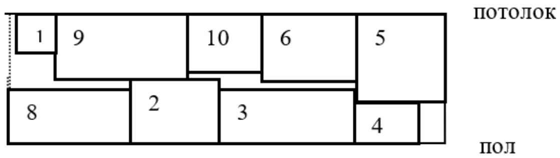

Рис. 6. Использование гильотинного разреза ленты на контейнеры и методом штамповки

При реализации второго варианта сближения блоков в контейнере предварительно прямоугольники, размещенные в блоках, упорядочиваются: в верхнем блоке по возрастанию высоты прямоугольников, а в нижнем блоке по убыванию высоты прямоугольников. Анализ результатов сближения блоков в контейнерах показал, что максимальная плотность упаковки полосы, исключающей наложение элементов в процессе сближения блоков, зависит от распределения блоков по контейнерам.

Третий вариант реализуется при использовании не гильотинного разреза всей полосы (методом штамповки). Блоки упорядочены. Рассматривается пара соседних блоков в полосе начиная с самой нижней. Положение элементов по вертикали в нижнем блоке пары зафиксировано. Для каждого прямоугольника, упакованного в верхнем блоке рассматриваемой пары, проверяется, может ли он сместиться вниз, проникая в нижний блок. Если можно сдвинуть его на нижний уровень так, что часть его останется на верхнем, он сдвигается и частично может быть «вжат» в предыдущий уровень

В результате сближения все прямоугольники верхнего уровня будут соприкасаться с прямоугольниками нижнего уровня (рис. 7).

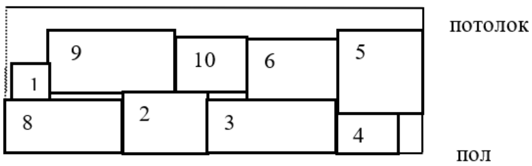

Рис. 7. Результат сближения всех прямоугольников верхнего уровня с прямоугольниками нижнего уровня

Освободившееся таким образом место на потолке верхнего блока может быть использовано элементами, расположенными в последующем блоке полосы (рис. 8). Таким образом, реализуется стратегия плоской упаковки, что позволяет снизить общую высоту полосы и площадь, занимаемую элементами.

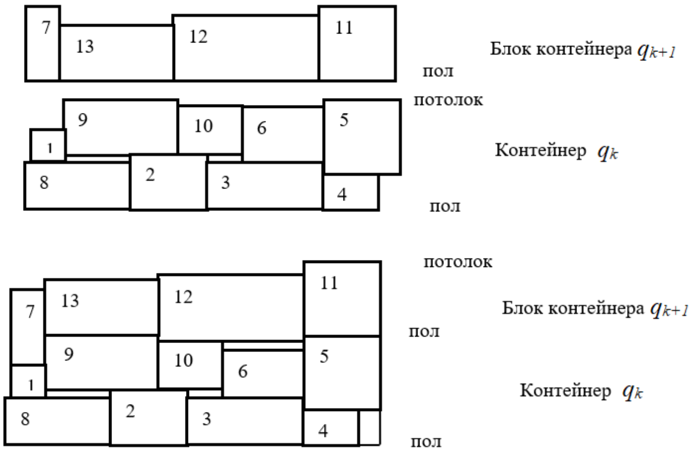

Рис. 8. Освободившееся место на потолке верхнего блока использованное элементами, расположенными в последующем блоке полосы

Анализ результатов третьего варианта сближения блоков в полосе показал, что максимальная плотность упаковки полосы, исключающей наложение элементов в процессе сближения двух блоков к зависит от взаимного расположения по горизонтали прямоугольников в блоках и порядка расположения блоков в полосе снизу вверх.

В связи с этим актуальной является разработка компрессионного алгоритма упорядочивания элементов на верхнем и нижнем блоках контейнера, для максимального сближения элементов двух блоков.

Компрессионные алгоритмы представляют собой процедуры улучшения упаковки. Каждый прямоугольник может располагаться в одной из двух ориентациях. Путем парных перестановок прямоугольников осуществляются дополнительное сближение верхнего и нижнего уровней блока контейнеров. На заключительном этапе производится сжатие по оси *OX*, и определяется размер заполненной части полосы – H^P.

2. Бионические методы формирования множества прямоугольных контейнеров с упакованными в них элементами. Ядром композитной архитектуры системы прямоугольной упаковки в полубесконечную полосу является подсистема формирования множества прямоугольных контейнеров $Q=\{q_k|k=1,2,\ldots,n_k\}$, включающих блоки с упакованными в них элементами.

Задача формирования множества блоков $B=\{b_j|j=1,2,...,m\}$, включающих наборы элементов прямоугольной формы $A=\{a_i|i=1,2,..., n_a\}$. решается алгоритмом одномерной упаковки элементов в одинаковые блоки.

Основу любого алгоритма одномерной упаковки составляют две процедуры: процедура выбора очередного элемента (ПВЭ) и процедура выбора очередного блока (ПВБ) для назначения в него выбранного элемента. В зависимости от способов, лежащих в основе этих процедур, алгоритмы упаковки можно разделить на 4 класса.

1. Порядок выбора элементов и блоков в ПВЭ и ПВБ задается априори.

2. Порядок выбора элементов в ПВЭ задается априори, блоков – в ПВБ на основе эвристик.

3. Порядок выбора элементов в ПВЭ на основе эвристик, блоков в ПВБ – априори.

4. Порядок выбора элементов и блоков в ПВЭ и ПВБ на основе эвристик.

Наименьшей трудоемкостью обладают алгоритмы 1 класса, наибольшей – 4 класса.

Анализ существующих методов и алгоритмов упаковки в блоки [1–11] показал, что в их основе лежит процедура упорядочивания исходного списка элементов, последовательно распределяемых по блокам. Другими словами, в качестве структуры данных, несущих информацию об упаковке, чаще всего используется последовательность номеров прямоугольников, представляющая порядок их укладки, которая называется приоритетным списком. Приоритетный список – это кодированное решение, в терминах генетического алгоритма – «хромосома».

Недостаток этого подхода заключается в том, что для группы элементов, входящих в один блок, их перестановка в списке не изменяет решения, что приводит к ненужным затратам при поиске решения. Пусть имеется некоторое решение задачи упаковки n элементов в m блоков. Пусть n_i – число элементов, уложенных в i-ый блок. Поскольку порядок, в котором элементы множества A_i вошли в i-ый блок не имеет значения, то решению будет соответствовать число списков равное $n_1! \cdot n_2! \cdot ... \cdot n_i! \cdot ... \cdot n_m!$ В связи с этим при поиске решения актуальна проблема синтеза и просмотра только одного списка, соответствующего конкретному решению.

Разработаны новые архитектура и методы решения задачи упаковки, построенные на основе декомпозиции и гибридизации разработанных автором роевых методов, использующие различные стратегии поиска, функционирующие параллельно-последовательно и реализующие более широкий обзор пространства решений, что позволяет обеспечить более высокую вероятность локализации глобального экстремума задачи.

Задача одномерной упаковки элементов в одинаковые блоки может быть сформулирована следующим образом. Дано множество прямоугольных элементов $A=\{a_i|i=1,2,...,n\}$. Вес элементов задается множеством $W=\{w_i|i=1,2,...,n\}$. Обозначим через W_j суммарный вес элементов множества A_j., $W_j=\sum w_i$, для $\{i|a_i \in A_j\}$. Задается максимально допустимый суммарный вес D элементов, множества A_j. Вводится ограничение $\forall j[W_j \le D]$. Необходимо множество A разбить на минимальное число m непустых и непересекающихся подмножеств A_j. $R=\{A_j|j=1,2,...,m\}$. $|R|=m$. $A=\cup A_j$, $A_j \ne \varnothing$, $(\forall i,j)$ $[A_i \cap A_j=\varnothing]$.

В стандартной постановке критерий оптимизации F – минимальное число m подмножеств A_j, таких, что $\cup A_j=A$. $F=m$. Цель оптимизации – минимизация m.

В работе алгоритм одномерной упаковки входит в состав алгоритма прямоугольной упаковки в полубесконечную полосу (1.5 Dimensional Bin Packing,1.5DBP) с конкретизацией некоторых определений. Множество элементов A_j, соответствует множеству элементов a_i, назначенных в блок b_j контейнера при упаковке полубесконечной полосы. Вес прямоугольного элемента a_i равен его ширине w_i.

Все блоки имеют одинаковую ширину D. Высота h_j каждого блока b_j определяется по самому высокому элементу, упакованному в нем. Необходимо множество A разбить на m непустых и непересекающихся подмножеств A_j, таких, чтобы суммарная высота блоков была минимальна. Каждое A_j упаковывается в блок b_j.

Критерий оптимизации $F=\sum h_j$ для $\{j=1, .., m\}$. Цель оптимизации – минимизация F.

Для решения задачи одномерной упаковки элементов в блоки применительно к задаче упаковки полубесконечной полосы разработан биоинспирированный алгоритм на основе модифицированной модели адаптивного поведения муравьиной колонии.

Работа муравьиного алгоритмов базируется на использовании коллективной эволюционной памяти (КЭП). КЭП муравьиной колонии представляет собой набор статистических показателей, отражающих для каждого фрагмента решения число ψ показывающее полезность фрагмента при построении решений на предыдущих итерациях алгоритма.

Структура алгоритма одномерной упаковки включает начальную и основную (итерационную) части (рис. 9). Основная часть включает три этапа, выполняемые на одной итерации.

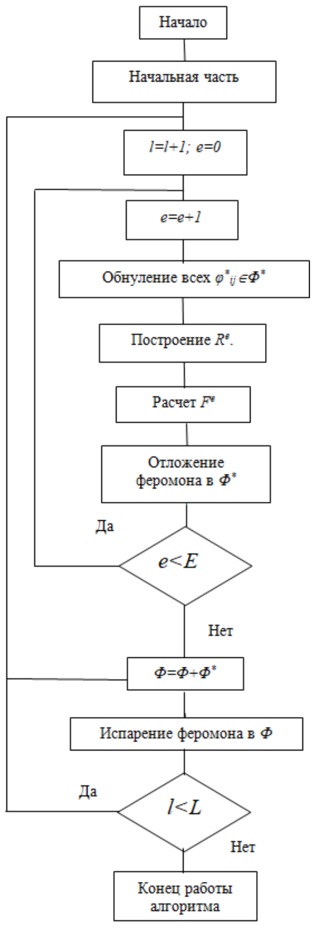

Рис. 9. Структура алгоритма одномерной упаковки включающая начальную и основную части

На начальном этапе алгоритма упаковки:

- фиксируются входные данные;
- задаются начальные значения управляющих параметров L – число итераций и K – агентов;
- формируется полный граф поиска решений $G=(X,U)$. Вершины множества $X=\{x_i|i=1,2,..., n\}$ соответствуют элементам списка $A=\{a_i|i=1,2,...,n\}$. Между вершинами множества X и элементами списка $A=\{a_i|i=1,2,...,n\}$ установлено соответствие $S_1:X\rightarrow A;\ x_i=S_1(a_i)$.

Для отражения коллективной эволюционной памяти в течение жизни популяции агентов и для формирования решения задачи используется полный граф $G=(X,U)$.

В начальной части алгоритма для откладывания феромона формируется множество:

$\Phi=\{\varphi_{ij}|i=1,2,...,n;\ j=1,2,...,n\}$ феромонных точек, соответствующих ребрам полного графа G. Между Φ и U установлено соответствие S_2: $\Phi\rightarrow U;\ \ \varphi_{ij}=S_2(u_{ij})$. u_{ij} – ребро, связывающее вершины x_i и x_j.

На всех феромонных точках откладывается одинаковое (начальное) количество феромона Θ/ν, где $\nu=|U|$. Параметр Θ задается априори.

В общем случае поиск решения задачи упаковки осуществляется коллективом муравьев $Z=\{z_e|e=1,2,...,E\}$.

Процесс поиска решений, реализуемый в основной части итерационный. Каждая итерация l включает три этапа. На первом основном этапе каждой итерации каждый агент z_k формирует свое собственное решение R^e. Ядром муравьиного алгоритма является конструктивный алгоритм, с помощью которого агенты на каждой итерации находят решение задачи. В отличие от канонической парадигмы муравьиного алгоритма агентом z_e на графе поиска решений $G=(X,U)$ в качестве решения R^e формируется разбиение множества прямоугольных элементов A на подмножества A^e_i, где A^e_j – подмножество элементов, назначенных агентом z_e в блок b^e_j. $R^e=\{A^e_j|j=1,2,...,n_j\}$.

Для построенного решения R^e рассчитывается целевая функция F^e и откладывается феромон в промежуточное множество феромоновых точек $\Phi^*=\{\varphi^*_{ij}|i=1,2,...,n;\ j=1,2,...,n\}$.

В работе используется циклический (ant-cycle) метод муравьиных систем. В этом случае феромон, накопленный на всеми агентами популяции, откладывается в феромонных точках Φ после полного формирования решений Для этих целей используется промежуточное множество $\Phi^*=\{\varphi^*_{ij}|i=1,2,...,n;\ j=1,2,...,n\}$ феромонных точек, идентичное Φ. $\varphi^*_{ij}=\delta^*_{io}$, где δ^*_{ij} суммарное количество феромона, отложенного агентами за одну итерацию на φ^*_{ij}.

После построенных агентами роя $Z=\{z_e|k=1,2,...,n_k\}$ множества $R=\{R^e|e=1,2,...,n_e\}$ решений на втором этапе феромон, накопленный в Φ^* откладывается в $\Phi=\{\varphi_{ij}\ |i=1,2,...,n;\ j=1,2,...,n\}$, т.е. $\Phi=\Phi+\Phi^*$.

На третьем этапе осуществляется испарение феромона.

На первом этапе итерации каждый агент z_e находит решение в виде совокупности подмножеств $R^e=\{A^e_{1,}\ A^e_2,\ A^t_3,\ ...,\ A^e_m\}$, полученных путем разбиения множества элементов $A=\{a_i|i=1,2,...,n\}$ на m непустых и непересекающихся подмножеств. На формируемые подмножеств A^k_j накладываются ограничения:

$\cup A^e_j=A,\ (\forall i,j)[A^e_i\cap A^e_j=\varnothing],\ A^e_j\neq\varnothing.$

Каждое подмножество A^e_j формируется агентом z_k последовательно. Подмножество A^e_j включает элементы, помещаемые в один блок b^e_j. Процесс формирования каждого подмножества элементов A^e_j пошаговый. t – номер шага. Пусть $A^e_j(t)$ – подмножество элементов для размещения в блоке e^k_j, сформированное после выполнения t шагов. На шаге $t=0$ $A^e_j(t)=\varnothing$.

Агенты обладают памятью. На каждом шаге t в памяти агента z_e имеется:

- номер формируемого блока – j;
- количество феромона $\delta_{ij}(t)$ отложенного в каждой феромоновой точке множества $\Phi=\{\varphi_{ij}|i=1,2,\ldots,n;\ j=1,2,\ldots,\ n\}$;
- множество элементов $A^e_j(t))$, уже включенных в формируемое подмножество на шаге t.

В начале каждого шага формируется множество еще не упакованных элементов O_j^e, таких, что если $a_i \in O_j^e$, то элемент a_i, может быть помещен в блок $A^e_j(t)$ без переполнения, т.е. без превышения веса D.

Первый, включаемый на шаге $t=1$ в $A^e_j(t)$ элемент $a_i \in O_j^e$, выбирается случайно.

На каждом последующем шаге t агент z_e применяет вероятностное правило выбора следующего элемента $a_i \in O_j^e$ для включения в $A^e_j(t)$.

Если $O_j^e \neq \varnothing$, то для каждого элемента $a_i \in O_j^e$ определяется соответствующая ему вершина $x_i=S_1(a_i)$ в графе G. Определяется множество вершин $X^e_i=S_1(A^e_j(t))$. Определяется множество ребер $U^e_i \subset U$ полного графа G, связывающих вершину x_i с множеством вершин X^e_i. Определяется множество феромонных точек $\Phi^e_i \subset \Phi$, соответствующих множеству U^e_i. Рассчитывается параметр $f_i^e(t)$ – суммарный уровень феромона на множестве феромонных точек $\Phi^e_i \subset \Phi$.

Вероятность P^e_i включения элемента $a_i \in O_j^e$ в множество $A^e_j(t)$ определяется следующим соотношением:

$$P_{ik}=f_{ik}(t) / \textstyle\sum_i (f_{ik}(t)),\ (i|a_i \in O_j^e). \qquad (1)$$

Агент z_e с вероятностью P_{ik} выбирает один из элементов $a_i \in O_j^e$, который включается в $A^e_j(t)$. После этого фиксируется размещение элемента a_i в $A^e_j(t)$.

Далее, если O_j^e оказывается пустым, то среди не упакованных элементов выбирается элемент, который помещается в следующий блок A^e_{j+1}.

Описанные действия повторяются при формировании A^e_{j+1}. Процесс завершается тогда, когда $O^e=\varnothing$.

Для построенного агентом z_e решения $R^e=\{A^e_j|j=1,2,\ldots,n_j\}$ рассчитывается значение целевой функции F^e.

Далее на множестве вершин $X^e_i=S_1(A^e_j)$ графа G формируется полный подграф $G^e_j=(X^e_j,U^e_j)$, и определяется множество всех ребер всех подграфов $U^e=\cup U^e_j$. Множеству ребер U^e соответствует вспомогательное множество феромонных точек $\Phi^*_e=S_2(U^e)$. $\Phi^*_e \subset \Phi^*$.

Агент z_e откладывает феромон на вспомогательном множестве феромонных точек Φ^*_e.

Количество феромона τ^e_{ij}, откладываемое муравьем z_e в каждой точке $\varphi^*_{ij} \in \Phi^*_e$ определяется следующим образом:

$$\tau^e_{ij}= \Theta / F^e, \qquad (2)$$

где Θ – базовая порция феромона. F^e – целевая функция для решения, полученного агентом z_e. Чем меньше F^e, тем больше феромона откладывается каждой точке $\varphi^*_{ij} \in \Phi^*_e$ и, следовательно, тем больше вероятность выбора этих элементов решения R^e при построении решений на следующей итерации.

На втором этапе итерации после того, как каждый муравей построил решение R^e и отложил феромон на соответствующем множестве Φ^*_e феромонных точек φ^*_{ij}, феромон накопленный в феромонных точках множества Φ^* добавляется к соответствующим феромонным точкам основного множества Φ.

На третьем заключительном этапе выполняется процедура испарения феромона на всех точках φ_{ij} основного множества Φ, в соответствии с формулой:

$$\tau_{ij} = \rho\ \tau_{ij}, \qquad (3)$$

где τ_{ij} – количество феромона в феромонной точке $\varphi_{ij} \in \Phi$, ρ – коэффициент обновления.

После выполнения всех действий на итерации находится агент с лучшим решением, которое запоминается. Далее осуществляется 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 $O(l \cdot n^2 \cdot m)$.

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

1. На начальном этапе МА упаковки фиксируются входные данные:

D – ширина полубесконечной полосы P;

$A=\{a_i|i=1,2,\dots,n\}$ – множество элементов (прямоугольников);

$W=\{w_i|i=1,2,\dots,n\}$ – вектор ширин прямоугольников;

$H=\{h_i|i=1,2,\dots,n\}$ – вектор высот прямоугольников.

Задаются начальные значения управляющих параметров:

E – число агентов; L– число итераций.

2. В соответствии с исходными данными формируется полный граф поиска решений $G=(X,U)$. Установлено соответствие $S_1:X\rightarrow A;\ x_i=S(a_i)$.

3. Формируется Φ множество феромонных точек φ_{ij}, соответствующих ребрам полного графа G. Установлено соответствие S_2: $\Phi\rightarrow U;\ \varphi_{ij}=S_2(u_{ij})$.

4. Задается значение параметра Θ.

5. На всех феромонных точках множества $\varphi_{ij}\in\Phi$ откладывается начальное количество феромона Θ.

6. $l=1$. *(l – номер итерации).*

7. Элементам φ^*_{ij} вспомогательного множества Φ^* феромонных точек присваивается нулевое значение.

8. $e=1$. *(e – номер агента).*

9. На первом этапе l-й итерации агентом z_k находится соответствующее ему решение задачи упаковки $R^e=\{A^e_1, A^e_2, A^e_{3,\ \dots,}\ A^e_m\}$.

10. Для решения R^e задачи одномерной упаковки находится значение целевой функции $F^e(l)$.

11. Для j в пределах от 1 до m формируются подмножества вершин $X^e_i=S_1(A^e_j)$ графа G.

12. Для j в пределах от 1 до m на базе подмножества вершин X^e_i графа G формируется полный подграф $G^e_j=(X^e_j,U^e_j)$, $X^e_j\subset X$, $U^e_j\subset U$.

13. Определяется множество всех ребер всех подграфов $U^e{=}\cup U^e_j$. Множеству ребер U^e соответствует вспомогательное множество феромонных точек $\Phi^*_e=S_2(U^e)$. $\Phi^*_e\subset\Phi^*$.

14. Агент z_e откладывает феромон на феромонных точках множества $\Phi^*_e\subset\Phi^*$.

15. Если $e<E$, то $e=e+1$ и переход к пункту 9, иначе переход к пункту 16.

16. Феромон, накопленный в феромонных точках множества Φ^*, добавляется к соответствующим феромонным точкам основного множества Φ.

17. Выполняется процедура испарения феромона феромонных точках основного множества Φ.

18. Если $l<L$, то $l=l+1$ и переход к пункту 7, иначе переход к пункту 19.

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

20. Конец работы алгоритма.

3. Экспериментальные исследования. Алгоритм упаковки в полубесконечную полосу на основе декомпозиции и гибридизации биоиспирированных методов реализован в виде программы GBP.

Для проведения экспериментальных исследований программы была использована процедура синтеза контрольных примеров с известным оптимумом по аналогии с известным методом AFEKO – Floorplanning Examples with Known Optimal area (20).

Разработанный алгоритм сравнивался с алгоритмами CAGA, PH и DDA.

Алгоритм GAGA, описанный в исследовании [21], сочетает в себе конструктивный метод и генетический метод, поэтому он имеет высокую вычислительную эффективность. Алгоритм исследования PH [22] использует механизмы, инспирированные природой, с высокой вычислительной эффективностью. Алгоритм DDA основан на эвристике динамической декомпозиции.

Для тестирования четырех вышеуказанных алгоритмов используются тестовые задачи SCPL1 ~ SCPL9 из OR-Benchmark. Результаты расчетов представлены в табл. 1, где *N* – число прямоугольников, H^* оптимальное значение.

Таблица 1

Результаты расчетов

Inst	N	Opt	GAHA	PH	DDA	GBA
		H^*	H	H	H	H
SCPL1	425	110	126.08	183.5	123	121
SCPL2	127	120	140.16	163.5	128	125
SCPL3	225	84	98.84	102	98	90
SCPL4	365	102	108.263	109.5	106	102
SCPL5	165	102	118.89	118	110	102
SCPL6	657	126	142.4	134	132	122
SCPL7	357	198	229.93	242	216	204
SCPL8	475	156	168.419	167	165	164
SCPL9	175	117	129.96	127	125	121

Как видно из табл. 1, результаты алгоритма динамической декомпозиции значительно лучше, чем у трех предыдущих алгоритмов, и некоторые результаты достигли предельной высоты.

Набор данных случайным образом выбран из OR-Benchmark. Элементы имеют ширину 135, длина не ограничена (http://www.laria.u-picardie.fr/hifi/OR-Benchmark).

Используя тестовые данные OR-Library, сравниваем разработанный алгоритм с алгоритмами EA и DDA. Результаты испытаний приведены в табл. 2.

Таблица 2

Результаты испытаний

TEST	H^*	Type	n	EA	DDA	GBA
				H	H	H
cgcut1	23	7	16	23	23	23
cgcut2	63	10	23	67	63	63
cgcut3	636	20	62	670	642	640
ngcut1	20	5	10	23	20	20
ngcut2	28	7	17	30	28	28
ngcut3	28	10	21	28	28	28
ngcut4	18	5	7	20	18	18
ngcut5	36	7	14	36	36	36
ngcut6	29	10	15	31	29	29

ngcut7	10	5	8	20	10	10
ngcut8	33	7	13	33	33	33
ngcut9	49	10	18	50	49	49
ngcut10	59	5	13	80	59	59
ngcut11	51	7	15	52	51	51
ngcut12	77	10	22	87	77	77
beng01	30	20	20	30	30	30
beng02	57	40	40	58	57	57
beng03	84	60	60	85	84	84
beng04	107	80	80	108	107	107
beng05	134	100	100	134	134	134
beng06	36	40	40	37	36	36
beng07	67	80	80	67	67	67

Заключение. Разработана методология нового направления поиска решений задач ортогональной упаковки на основе моделей адаптивного поведения биологических систем

Разработан высокоэффективный гибридный биоинспирированный метод решения задач одномерной и прямоугольной упаковки, основанный на декомпозиции задачи на множество подзадач и интеграции методов поисковой оптимизации.

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 подграфов на вершинах, входящих в каждую часть, на ребрах которых откладывается феромон.

Разработаны поисковые методы для решения задач гильотинного и не гильотинного прямоугольного раскроя.

Разработанные компрессионные алгоритмы формирования карты раскроя на основе биоинспирированного поиска позволяют сократить отходы материала в 2-3 раза.

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.

Получены лучшие результаты по сравнению с тестируемыми методами.

Предложенные в работе теоретические положения для решения задач упаковки и раскроя промышленных объектов в условиях единичного производства реализованы в виде методик, алгоритмов и прикладного программного обеспечения.

По сравнению с существующими алгоритмами достигнуто улучшение результатов на 3–5 %.

Временная сложность алгоритма, полученная экспериментальным путем, практически совпадает с теоретическими исследованиями и для рассмотренных тестовых задач составляет $O(n^2)$.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Stephen C.H. Leung, Defu Zhang.* A fast layer-based heuristic for non-guillotine strip packing // Expert System with Application. – 2011. – P. 13032-13042.
2. *Тимофеева О.П., Соколова Э.С., Милов К.В.* Генетический алгоритм в оптимизации упаковки контейнеров // Тр. НГТУ. Информатика и системы управления. – 2013. – С. 167-172.
3. *Валеева А.Ф.* Применение конструктивных эвристик в задачах раскроя-упаковки // Приложение к журналу «Информационные технологии». – 2006. – С. 1-24.

4. *Лебедев Б.К., Лебедев О.Б. Лебедева Е.М.* Модернизированный муравьиный алгоритм синтеза идентифицированного дерева гильотинного разреза при планировании СБИС // Известия ЮФУ. Технические науки. – 2017. – № 7. – С. 15-28.
5. *Лебедев О.Б., Зорин В.Ю.* Упаковка на основе метода муравьиной колонии // Известия ЮФУ. Технические науки. – 2010. – С. 25-30.
6. *Defu Zhang, Yuxin Che, Furong Ye, Yain-Whar Si, Stephen C.* A hybrid algorithm based on variable neighbourhood for the strip packing problem // Journal of Combinatorial Optimization. – 2016. – P. 513-530.
7. *Ахтямов А.А., Картак В.М.* Упаковка и оценка плотности упаковки ортогональных многоугольников в полубесконечную полосу // Информационные технологии интеллектуальной поддержки принятия решений. – 2014. – С. 117-121.
8. *Jaya Thomas, Chaudhari Narendra S.* Hybrid approach for 2d strip packing problem using genetic algorithm // Proceedings of the 12th international conference on Artificial Neural Networks: advances in computational intelligence. – 2013. – P. 213-230.
9. *Месягутов М.А., Мухачева Э.А., Белов Г.Н., Шайтхауэр Г.* Упаковка одномерных контейнеров с продолженным выбором идентичных предметов: точный метод поиска оптимального решения // Автоматика и телемеханика. – 2011. – С. 154-173.
10. *Lei Wang, Aihua Yin.* A quasi-human algorithm for the two-dimensional rectangular strip packing problem: in memory of Prof. Wenqi Huang // Journal of Combinatorial Optimization. – 2016. – P. 416-444.
11. *Потарусов Р.В., Курейчик В.М.* Проблема одномерной упаковки элементов // Известия ТРТУ. – 2006. – С. 88-93.
12. *Карпенко А.П.* Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учеб. пособие. – М,: Изд-во МГТУ им. Н.Э. Баумана, 2016. – 448 с.
13. *Лебедев О.Б.* Модели адаптивного поведения муравьиной колонии в задачах проектирования. – Таганрог: Изд-во ЮФУ, 2013. – 199 с.
14. *Лебедев Б.К., Лебедев О.Б.* Муравьиные алгоритмы разбиения, использующие представления задачи, отличные от канонического // Вестник Ростовского государственного университета путей сообщения. – 2016. – С. 42-47.
15. *Blum C., Roli A.* Metaheuristics in combinatorial optimization: overview and conceptual comparison // ACM computing surveys. – 2003. – P. 268-308.
16. *Лебедев Б.К., Лебедев О.Б.* Распределение ресурсов на основе гибридных моделей роевого интеллекта // Научно-технический вестник информационных технологий, механики и оптики. – 2017. – С. 1063-1073.
17. *Житников В.П.* Задача прямоугольной упаковки в полубесконечную полосу: поиск решения в окрестности локальной нижней границы // Информационные технологии. – 2007. – № 6. – С. 55-61.
18. *Raidl G.R. A* Unified View on Hybrid Metaheuristics // In: Lecture Notes In Computer Science. – Springer, Verlag, 2006. – P. 1-12.
19. *Лебедев Б.К., Лебедев О.Б., Лебедева Е.О.* Роевой алгоритм планирования работы многопроцессорных вычислительных систем // Инженерный вестник Дона. – 2017.
20. *Cong J., Romesis M., Xie M.* Optimality, Scalability and Stability Study of Partitioning and Placement Algorithms // Proc. of the International Symposium on Physical Design. – 2003. – P. 88-94.
21. *Hifi M., M'Hallah R.* A hybrid algorithm for the two-dimensional layout problem: The cases of regular and irregular shapes // International Transactions in Operational Research. – 2003. – P. 195-216.
22. *Zhang DeFu, Han ShuiHua, Ye WeiGuo.* A Bricklaying Heuristic Algorithm for the Orthogonal Rectangular Packing Problem // Chinese Journal of Computers. – 2008. – P. 509-515.

REFERENCES

1. *Stephen C.H. Leung, Defu Zhang.* A fast layer-based heuristic for non-guillotine strip packing, *Expert System with Application*, 2011, pp. 13032-13042.
2. *Timofeeva O.P., Sokolova E.S., Milov K.V.* Geneticheskiy algoritm v optimizatsii upakovki konteynerov [Genetic algorithm in optimization of container packaging], *Tr. NGTU. Informatika i sistemy upravleniya* [Proceedings of NSTU. Informatics and control systems], 2013, pp. 167-172.

3. *Valeeva A.F.* Primenenie konstruktivnykh evristik v zadachakh raskroya-upakovki [Application of constructive heuristics in cutting-packing problems], *Prilozhenie k zhurnalu «Informatsionnye tekhnologii»* [Supplement to the journal «Information Technologies»], 2006, pp. 1-24.
4. *Lebedev B.K., Lebedev O.B. Lebedeva E.M.* Modernizirovannyy murav'inyy algoritm sinteza identifitsirovannogo dereva gil'otinnogo razreza pri planirovanii SBIS [Modernized ant algorithm for synthesizing an identified guillotine cut tree when planning VLSI], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2017, No. 7, pp. 15-28.
5. *Lebedev O.B., Zorin V.Yu.* Upakovka na osnove metoda murav'inoy kolonii [Packaging based on the ant colony method], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2010, pp. 25-30.
6. *Defu Zhang, Yuxin Che, Furong Ye, Yain-Whar Si, Stephen C.* A hybrid algorithm based on variable neighbourhood for the strip packing problem, *Journal of Combinatorial Optimization,* 2016, pp. 513-530.
7. *Akhtyamov A.A., Kartak V.M.* Upakovka i otsenka plotnosti upakovki ortogonal'nykh mnogougol'nikov v polubeskonechnuyu polosu [Packing and estimation of the packing density of orthogonal polygons in a semi-infinite strip], *Informatsionnye tekhnologii intellektual'noy podderzhki prinyatiya resheniy* [Information technologies for intelligent decision support], 2014, pp. 117-121.
8. *Jaya Thomas, Chaudhari Narendra S.* Hybrid approach for 2d strip packing problem using genetic algorithm, *Proceedings of the 12th international conference on Artificial Neural Networks: advances in computational intelligence,* 2013, pp. 213-230.
9. *Mesyagutov M.A., Mukhacheva E.A., Belov G.N., Shaytkhauer G.* Upakovka odnomernykh konteynerov s prodolzhennym vyborom identichnykh predmetov: tochnyy metod poiska optimal'nogo resheniya [Packing one-dimensional containers with continued selection of identical items: an exact method for finding the optimal solution], *Avtomatika i telemekhanika* [Automation and Remote Control], 2011, pp. 154-173.
10. *Lei Wang, Aihua Yin.* A quasi-human algorithm for the two-dimensional rectangular strip packing problem: in memory of Prof. Wenqi Huang, *Journal of Combinatorial Optimization,* 2016, pp. 416-444.
11. *Potarusov R.V., Kureychik V.M.* Problema odnomernoy upakovki elementov [The problem of one-dimensional packing of elements], *Izvestiya TRTU* [Izvestiya TSURE], 2006, pp. 88-93.
12. *Karpenko A.P.* Sovremennye algoritmy poiskovoy optimizatsii. Algoritmy, vdokhnovlennye prirodoy: ucheb. posobie [Modern search engine optimization algorithms. Algorithms inspired by nature: a tutorial]. Moscow: Izd-vo MGTU im. N.E. Baumana, 2016, 448 p.
13. *Lebedev O.B.* Modeli adaptivnogo povedeniya murav'inoy kolonii v zadachakh proektirovaniya [Models of adaptive behavior of an ant colony in design problems]. Taganrog: Izd-vo YuFU, 2013, 199 p.
14. *Lebedev B.K., Lebedev O.B.* Murav'inye algoritmy razbieniya, ispol'zuyushchie predstavleniya zadachi, otlichnye ot kanonicheskogo [Ant partitioning algorithms using problem representations other than the canonical one], *Vestnik Rostovskogo gosudarstvennogo universiteta putey soobshcheniya* [Bulletin of the Rostov State Transport University], 2016, pp. 42-47.
15. *Blum C., Roli A.* Metaheuristics in combinatorial optimization: overview and conceptual comparison, *ACM computing surveys*, 2003, pp. 268-308.
16. *Lebedev B.K., Lebedev O.B.* Raspredelenie resursov na osnove gibridnykh modeley roevogo intellekta [Resource distribution based on hybrid models of swarm intelligence], *Nauchno-tekhnicheskiy vestnik informatsionnykh tekhnologiy, mekhaniki i optiki* [Scientific and technical bulletin of information technologies, mechanics and optics], 2017, pp. 1063-1073.
17. *Zhitnikov V.P.* Zadacha pryamougol'noy upakovki v polubeskonechnuyu polosu: poisk resheniya v okrestnosti lokal'noy nizhney granitsy [The problem of rectangular packing in a semi-infinite strip: searching for a solution in the neighborhood of the local lower boundary], *Informatsionnye tekhnologii* [Information Technologies], 2007, No. 6, pp. 55-61.
18. *Raidl G.R. A* Unified View on Hybrid Metaheuristics, *In: Lecture Notes In Computer Science*. Springer, Verlag, 2006, pp. 1-12.
19. *Lebedev B.K., Lebedev O.B., Lebedeva E.O.* Roevoy algoritm planirovaniya raboty mnogoprotsessornykh vychislitel'nykh sistem [Swarm algorithm for scheduling the operation of multiprocessor computing systems], *Inzhenernyy vestnik Dona* [Engineering Bulletin of the Don], 2017.

20. *Cong J., Romesis M., Xie M.* Optimality, Scalability and Stability Study of Partitioning and Placement Algorithms, *Proc. of the International Symposium on Physical Design,* 2003, pp. 88-94.
21. *Hifi M., M'Hallah R.* A hybrid algorithm for the two-dimensional layout problem: The cases of regular and irregular shapes, *International Transactions in Operational Research,* 2003, pp. 195-216.
22. *Zhang DeFu, Han ShuiHua, Ye WeiGuo.* A Bricklaying Heuristic Algorithm for the Orthogonal Rectangular Packing Problem, *Chinese Journal of Computers,* 2008, pp. 509-515.

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

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

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

Ганжур Марина Александровна – Донской государственный технический университет; e-mail: mganzhur@yandex.ru; г. Ростов-на-Дону, Россия; тел.: 89081819111; кафедра вычислительных систем и информационной безопасности; старший преподаватель.

Lebedev Boris Konstantinovich – Southern Federal University; e-mail: lebedev.b.k@gmail.com; Taganrog, Russia; phone: + 79282897933; the department of computer aided design; professor.

Lebedev Oleg Borisovich – e-mail: lebedev.ob@mail.ru; phone: +79085135512; associate professor.

Ganzhur Marina Aleksandrovna – Don State Technical University; e-mail: mganzhur@yandex.ru; Rostov-on-Don, Russia; phone: +79081819111; the department of computing systems and information; senior lecturer.