УДК 519.85

ПОСЛЕДОВАТЕЛЬНАЯ МЕТАЭВРИСТИКА ДЛЯ ЗАДАЧИ ПРИОРИТЕТНОЙ УПАКОВКИ С НЕСОВМЕСТИМЫМИ КАТЕГОРИЯМИ

Воронов В.С., Пересунько П.В., Виденин С.А., Матюхин Н.Е.

ФГАОУ ВО «Сибирский федеральный университет», Красноярск, e-mail: voronov.volodya2013@yandex.ru

В статье рассматривается задача упаковки прямоугольников, а именно ее новая постановка. Она заключается во введении приоритетов и несовместимых категорий. Эта задача называется задачей приоритетной упаковки с несовместимыми категориями и динамически изменяющимися размерами контейнера. Данная оптимизационная задача возникает при раскрое металлических слитков, где приоритеты отражают порядок производства заготовок, а отдельные группы деталей, обладающие разной толщиной, несовместимы между собой. Часто при изготовлении деталей используется гильотина. Производство деталей разной толщины происходит путем прокатки слитка, в результате чего его размеры изменяются. Это делает необходимым не только вычисление оптимальной схемы раскроя, но и определение длины субконтейнера для каждой группы. В статье предлагается новый последовательный метаэвристический алгоритм для решения поставленной задачи. В ходе упаковки весь набор прямоугольников разделяется на группы, которые упаковываются отдельно. В результате схемы раскроя одинаковых групп объединяются. Предлагаемая метаэвристика использует однопроходной детерминированный приоритетный эвристический алгоритм. Структура метаэвристика позволяет применять любые алгоритмы упаковки, удовлетворяющие приведенным ограничениям. Это дает возможность сохранять и комбинировать свойства используемых алгоритмов. Качество работы алгоритма оценено на основе тестовых примеров, составленных для приведенной постановки задачи.

Ключевые слова: упаковка прямоугольников в контейнер, упаковка прямоугольников в полосу, метаэвристика, несовместимые категории, приоритет

SEQUENTIAL METAHEURISTICS FOR PRIORITY PACKAGING PROBLEM WITH INCOMPATIBLE CATEGORIES

Voronov V.S., Peresunko P.V., Videnin S.A., Matyukhin N.E.

Siberian Federal University, Krasnoyarsk, e-mail: voronov.volodya2013@yandex.ru

The paper considers the bin packing problem, namely its new formulation. It consists of introducing priorities and incompatible categories. This problem is called a priority packaging problem with incompatible categories and dynamically changing container sizes. This optimization problem arises when cutting metal ingots, where the priorities reflect the order of production of parts, and individual groups of parts with different thicknesses are incompatible with each other. Often, the guillotine is used in the manufacture of parts. The production of parts of different thicknesses occurs by rolling the ingot, as a result of which its dimensions change. This makes it necessary not only to calculate the optimal cutting scheme, but also to determine the length of the sub-container for each group. The paper proposes a new sequential metaheuristic algorithm for solving this problem. During packaging, the entire set of rectangles is divided into groups that are packed separately. As a result, the cutting schemes of the same groups are combined. The proposed metaheuristics uses a single-pass deterministic priority heuristic algorithm. The metaheuristics structure allows you to use any packaging algorithms that meet the above restrictions. This allows you to save and combine the properties of the algorithms used. The quality of the algorithm is evaluated based on test examples compiled for the given problem formulation.

Keywords: bin packing problem, strip packing problem, metaheuristic, incompatible categories, priority

Задачи оптимизации распространены во многих сферах, таких как медицина, металлургия, логистика и др. Получение оптимальных параметров позволяет добиться существенных преимуществ, что эффективно сказывается на предприятии в условиях сильной конкуренции. Не является исключением и задача оптимального использования ресурсов, которая позволяет не только сократить количество отходов, но и повысить количество выпускаемой продукции. Одной из таких сфер является металлургия, особый интерес представляет обработка цветных и драгоценных металлов. Неэффективное использование таких материалов влечет безвозвратные потери материала и увеличение затрат на переработку отходов. Часто подобные производства могут быть реализованы следующим образом. Детали разной формы изготавливаются из одного слитка путем его деформации. Исходный слиток прямоугольной формы прокатывается до определенной толщины, затем из него вырезаются заготовки. Такая процедура повторяется до тех пор, пока детали могут быть размещены. Стоит отметить, что для экономии материала резка осуществляется на гильотине. Одна из проблем, связанных с таким способом производства, заключается в планировании. Детали большей толщины располагаются на листе в первую очередь. Таким образом,

вероятность размещения более тонких деталей уменьшается. Однако их приоритет может быть выше прочих. В большинстве случаев человеку не под силу учесть это при изготовлении деталей.

В результате эта задача является одной из разновидностей известной задачи упаковки прямоугольников (ВРР). ВРР - это задача оптимизации, которая минимизирует количество отходов или используемых ресурсов. ВРР имеет большое количество разновидностей. Так, наиболее известными являются варианты с наличием гильотинных ограничений, фиксированной или неограниченной длиной листа, возможностью поворота деталей [1]. Использование дополнительных ограничений приводит к появлению новых вариантов задачи. В этой работе рассматривается задача приоритетной упаковки (Р) в несколько контейнеров неизвестной ограниченной длины (М) с несовместимыми категориями (IC) и гильотинными ограничениями (G). Мы закодируем эту задачу как BPP|P|M|IC|G.

Учет приоритетов в ВРР|Р|М|ІС|С позволяет размещать приоритетные элементы в первую очередь, что дает возможность соблюдать порядок производства. Несовместимость групп деталей вытекает из их геометрических размеров, а именно из толщины. Детали разной толщины не могут быть размещены в одном контейнере. Стоит отметить, что оптимальное количество контейнеров, на которые необходимо разбить исходный контейнер, заранее не известно, как не известна и их длина. В результате в качестве решения задачи необходимо не только предоставлять схему упаковки, но и вычислить оптимальное разбиение исходного контейнера на несколько частей.

Для решения BPP|P|M|IC|G предлагается новая метаэвристика. Она основывается на применении приоритетной эвристики, предложенной в [2], с несколькими модификациями. Предлагаемый алгоритм тестируется на наборах прямоугольников, специально составленных для этой задачи.

Целями данного исследования являются получение математической формулировки задачи приоритетной упаковки с несовместимыми категориями и динамически изменяемыми размерами контейнеров, а также разработка алгоритма ее решения.

Стоит отметить, что задача ВРР является NP-полной. Это означает, что не существует алгоритмов, решающих эту задачу за полиномиальное время. Таким образом, необходимо разрабатывать эвристические алгоритмы, которые позволяют получить

приближенное решение за приемлемое время работы.

Задача упаковки прямоугольников (ВРР) хорошо изучена и описана в различных источниках (например, [3]). Существует также множество вариаций этой задачи. В контексте нашей постановки рассмотрим несколько вариаций наиболее близких задач. Одним из вариантов задачи упаковки является задача с учетом приоритета (2DBPP|D), которая называется задачей дуальной, или приоритетной, упаковки. В литературе уделяют мало внимания приоритетности изготовления деталей в промышленности и в большей степени склоняются к физическим и ценовым характеристикам элементов. Поэтому неудивительно, что эта вариация получила свое развитие в сфере медицины для оптимальной загрузки медицинского оборудования. Проблема приоритетной упаковки рассматривается в работах [4, 5], где ставится задача распределения ресурсов с учетом ограниченного времени проведения операции. Дальнейшее развитие она получила в работе [6], где была описана математическая модель задачи планирования и приводилось ее решение с помощью эвристики на основе алгоритма First Fit Decreasing.

Другим вариантом задачи 2DBPP является упаковка в несколько полос (2DMBPP). Эта задача имеет два варианта постановки: с одинаковыми и с переменными размерами контейнеров (2DVSBPP). Эта задача встречается при оптимизации нагрузки на вычислительные ресурсы, в логистике и т.д. Так, в работе [7] предлагается ряд эвристических методов и рассматривается их приложение для решения задачи логистики как обобщенной задачи упаковки контейнеров с прибылями, зависящими от контейнера. Применение 2DMBPP для пакетного планирования с распределением ресурсов было рассмотрено в [8], где эвристика использовалась для минимизации потребления ресурсов. Исследование алгоритма First Fit для задачи упаковки с ограничением по количеству элементов проводилось в работе [9]. В основном авторы рассматривают упаковку в контейнеры фиксированного или разного размера. Никаких ограничений по динамически изменяющимся размерам или по неопределенному количеству контейнеров не ставится.

Задача упаковки с конфликтами (ВРС) рассматривает возможность несовместимости отдельных элементов. Часто в таких задачах для определения совместимости между элементами используется граф, где вершины обозначают элементы или группы, которые должны быть упакованы, а ребра — совместимость между ними. Напри-

мер, в работе [10] рассматривается решение задачи с помощью эвристических методов на основе древесной декомпозиции. Алгоритм разбивает задачу на подзадачи, которые решаются независимо. Затем происходит объединение полученных решений. В работе [11] авторы приводят обновленные верхние и нижние границы для ВРС, а также детерминированный алгоритм на основе формулировки покрытия множества, решаемый с помощью алгоритма ветвления и цены.

В последнее время появилась новая задача упаковки с совместимыми категориями (ВРСС) [12]. Она является обобщением задачи упаковки с конфликтами (ВРС), где рассматривается попарная несовместимость. ВРСС предполагает совместимость отдельных групп упаковываемых элементов. В работе [12] впервые ставится задача ВРСС. Авторы приводят модификацию алгоритма поиска переменной окрестности (VNS), а также предлагают набор тестовых примеров, на котором проводят обширное тестирование алгоритма.

Таким образом, приведенные здесь задачи упаковки так или иначе схожи с предлагаемой нами формулировкой задачи. Однако ни одна из предложенных задач не рассматривает случай с не известным заранее количеством контейнеров и их динамически изменяющимися размерами. Кроме этого, насколько нам известно, это первое рассмотрение задачи с совокупностью ограничений, таких как гильотинные разрезы, наличие приоритетов и несовместимые группы прямоугольников.

Постановка задачи

Обычная ВРР заключается в размещении набора прямоугольников $R = \{r_i, i=1, ..., n\}$ с размерами $r_i = (w_i, l_i)$, где w_i — ширина, l_i — длина i-го прямоугольника. Прямоугольники размещаются на листе фиксированного размера с шириной W_0 и длиной L_0 без перекрытия. Целевой функцией в данном случае является площадь отходов:

$$f(R, I') = W_0 L_0 - \sum_{i \in I'} w_i l_i = \min_{I}$$
 (1)

Здесь Γ – множество индексов размещенных прямоугольников, $\Gamma \subseteq I = \{i = 1, ..., n\}$.

Для постановки задачи BPP|P|M|IC|G представим прямоугольник в виде $r_i = (w_i, \ l_i, \ h_i, \ p_i)$. Здесь $h_i \in H = \{h_j, j \in I\}$ — толщина i-го прямоугольника. Толщина прямоугольника задает толщину листа, из которого он может быть изготовлен. Например, на листе толщиной 2 мм могут быть размещены прямоугольники с соот-

ветствующей толщиной. Лист постепенно деформируют (прокатывают) до определенной толщины, изготавливают детали, и оставшийся лист деформируют повторно. Это делает невозможным одновременное размещение прямоугольников разной толщины. Использование толщины позволяет сгруппировать прямоугольники в $m = |supp\ H|$ групп. При этом:

$$n = \sum_{j=1}^{m} |R_{j}| = \sum_{j=1}^{m} n_{j} , R_{j} = \{r_{i} | h_{i} = h_{j} \forall i \in I\} \subseteq R,$$

где h_j — толщина группы прямоугольников, $j \in J = \{1, ..., m\}$ и $H_j = \{n_j \bullet h_i \mid h_i = h_j \, \forall i \in I \}$. Эти группы являются несовместимыми.

Приоритет i-го прямоугольника $p_i \in P = \{p_i, i \in I \mid p_i \in \mathbb{Z}_{\geq 0}\}$ характеризует степень важности для производства. Чем больше p_i , тем предпочтительнее деталь для размещения на текущем листе. В случае одинакового приоритета порядок размещения прямоугольников не важен. Мы примем $p_i = 1$ как максимальный приоритет, т.е., чем выше значение p_i , тем ниже приоритет. Прямоугольники в каждой группе H_i (i = 1, ..., i) можно сгруппировать в i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i = i =

$$\left| R_j \right| = \sum_{k=1}^{q_j} \left| P_{jk} \right| = \sum_{k=1}^{q_j} g_{jk} , P_{jk} \subseteq P_j ,$$

где $P_{jk} = \{g_{jk} \bullet p_i \mid p_i = p_{jk} \forall i \in I_{jk} \subseteq I\}$, а I_{jk} индексы прямоугольников с одинаковым приоритетом в группе R_j . Стоит отметить, что при $p_i = const$, $\forall i \in I$ задача BPP|P|M|IC|G вырождается в BPP|M|IC|G, т.е. без учета приоритета.

Мы полагаем, что при деформации листа (прокатке) его размеры изменяются по правилу сохранения объема, т.е. $V_0 = V_1$, где $V_0 = W_0 L_0 h_0$ — объем до деформации, $V_1 = W_1 L_1 h_1$ — объем после деформации. Пренебрегая изменениями размеров одной из сторон листа, ортогонально которой происходит деформация (например, W_0), получаем:

$$L_1 = \gamma \frac{L_0 h_0}{h_1}, \, \gamma > 0,$$
 (2)

где γ — коэффициент для учета дефектов, возникающих на концах листа при деформации. Далее для простоты принимается $\gamma = 1$.

Таким образом, BPP|P|M|IC|G заключается в размещении прямоугольников с учетом приоритета и разбиении исходного контейнера на субконтейнеры с учетом динамически изменяющейся длины. Как упоминается в работе [13], задачу (1) с учетом

приоритета можно представить в виде задачи многокритериальной оптимизации:

$$f(R, I'_1, ..., I'_m) =$$

$$= (f_1(R, I'_1, ..., I'_m), f_2(R, I'_1, ..., I'_m)) = \min, (3)$$

$$f_1(R, I'_1, ..., I'_m) = \sum_{j=1}^m \left(W_j L_j - \sum_{i \in I'_j} w_i I_i \right),$$

$$f_2(R, I'_1, ..., I'_m) = \sum_{j=1}^m \left(\sum_{i \in I'_j} p_i \right),$$

где Γ_j — множество индексов размещенных прямоугольников группы R_j . Если $\Gamma_j = \emptyset$, то прямоугольники из группы R_j не были размещены. Размеры контейнера (W_j, L_j) , выделенного для упаковки прямоугольников группы R_j , должны определяться исходя из правила деформации (2) и конфигурации соответствующей группы. При этом если $\Gamma_j = \emptyset$, то $W_j L_j = 0$.

Накладывая дополнительные ограничения на задачу (3), можно получать различные вариации задачи упаковки:

- стороны прямоугольников должны быть параллельны сторонам листа;
- прямоугольники разрешено поворачивать на 90 градусов (R);

- все разрезы должны быть только гильотинными (G);
- деформация листа производится в фиксированном направлении (FD) (например, вдоль L_0).

В данной работе мы рассматриваем задачу гильотинной приоритетной упаковки прямоугольников с несовместимыми категориями и фиксированным направлением деформации BPP|P|M|IC|G|FD.

Приоритетная эвристика

Приоритетная эвристика (РН) – это однопроходная детерминированная эвристика, предложенная в [2]. Она представляет собой рекурсивную процедуру упаковки для обеспечения гильотинных ограничений. Выбор прямоугольника из набора осуществляется с помощью «внутренних» приоритетов, характеризующих один из пяти возможных вариантов размещения. Таким образом, прямоугольник с большим приоритетом выбирается первым. Алгоритм имеет один параметр для настройки, а именно вариант сортировки: по длине или по ширине. В своей работе авторы проводят результаты работы алгоритма на разных примерах из литературы, в том числе на основе дата сета ZDF, предложенного в [14, 15]. В результате отмечается, что РН хорошо подходит для крупномасштабных задач.

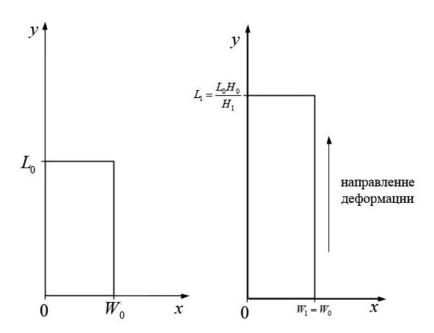


Рис. 1. Положение исходного контейнера и направление его деформации

Для соответствия BPP|P|M|IC|G в PH были внесены некоторые изменения. Так, сортировка прямоугольников выполняется в каждой группе R_{jk} отдельно. В алгоритм упаковки добавлен параметр сдвига по оси Y. Это позволяет последовательно упаковывать группы прямоугольников, сохраняя их положение.

Для учета приоритетности прямоугольников в алгоритме упаковки деталей с RG итерации проводятся только по элементам наивысшего приоритета p. При этом в процедуре рекурсивной упаковки RecursivePacking(x, y, w, h) этап оценки внутреннего приоритета проводится в порядке убывания приоритета (по подгруппам R_{jk}) всей группы элементов R_j с одной толщиной. Это позволяет заполнять пространство, в котором невозможно разместить прямоугольники высокого приоритета, прямоугольниками более низкого приоритета.

Последовательный метаэвристический алгоритм

Примем такое расположение исходного контейнера, что его горизонтальная сторона соответствует ширине W, а вертикальная соответствует длине L (рис. 1). Пусть деформация (прокатка) происходит вдоль L, а W остается постоянной.

Одна из задач алгоритма — распределить длины субконтейнеров для прямоугольников из групп H_p , $j \in J$. Очевидно, что длина каждого субконтейнера лежит в пределах $L_0 \leq L \leq L_0 h_0 / min\ H$, где $h_0 \geq min\ H$ толщина исходного контейнера. Таким образом, одним из результатов работы алгоритма должно стать отображение L_{bins} : $supp\ H \to \mathbb{R}^+$, которое ставит в соответствие толщине упакованной группы длину субконтейнера, на которые будет разбит исходный контейнер с учетом его деформации. Последовательность всех длин субконтейнеров должна удовлетворять неравенству:

$$L_0 \le \sum_{h \in \text{Supp} H} L_{buns}\left(h\right) \le \frac{L_0 h_0}{\min H} .$$

Стоит отметить, что, чем больше выделен размер контейнера для группы с одной толщиной, тем меньше места остается для упаковки остальных групп. Это необходимо учитывать при распределении длин. Другими словами, упаковка прямоугольников группы R_j оставляет меньше шансов на то, что все прямоугольники первого приоритета группы R_i $(i \neq j)$ будут упакованы.

Предлагаемая последовательная метаэвристика (SM) рассматривает подгруппы прямоугольников R_{jk} в порядке их следования в отсортированной последовательности $G = \{(h_j, p_{jk}), k = 1, ..., q_j, j = 1, ..., m\}$. Сортировка осуществляется лексикографически по неубыванию приоритета и невозрастанию толщины. Это позволяет упаковывать прямоугольники большей толщины в первую очередь. Использование p_{jk} позволяет учесть приоритетность упаковки деталей. Благодаря такому упорядочиванию подгруппы R_{jk} с высоким приоритетом упаковываются в первую очередь, затем оставшееся место распределяется между подгруппами в порядке увеличения p_{jk} .

Итеративная процедура алгоритма (рис. 2) построена следующим образом. Контейнер деформируется до толщины группы деталей R_i по правилу (2), после чего упаковка осуществляется с помощью алгоритма РН. Результатом упаковки выступают необходимая длина контейнера L' и мультимножество прямоугольников R, В завершение итоговое решение обновляется, длина и толщина исходного контейнера корректируются. Стоит отметить, что в отсортированной последовательности С элементы \hat{h}_i не обязательно будут упорядочены в порядке невозрастания. Это происходит потому, что значения p_{jk} имеют приоритет над h_{j} . Поэтому в процессе поиска решения длина контейнера может как увеличиваться $(h_0 > h_1$ в (2)), так и уменьшаться $(h_1 > h_0$ в (2)). Это приводит к неупорядоченной упаковке относительно h_i . Однако в процессе производства лист проќатывается в порядке убывания толщины. Поэтому по завершении процедуры необходимо отсортировать группы размещенных прямоугольников в нужном порядке.

Во время упаковки могут возникнуть два случая. Первый заключается в том, что все прямоугольники подгруппы R_{ik} могут быть размещены в контейнер текущей длины, а оставшееся место будет отведено для упаковки других подгрупп. Тогда необходимо воспользоваться алгоритмом для задачи упаковки в полосу, который, помимо схемы размещения, возвращает необходимую длину. За это отвечает процедура PackStrip(W, R_p , y_c), где y_c отвечает за сдвиг координат схемы размещения по оси У. Второй случай возникает, когда длины контейнера не хватает для размещения всех прямоугольников подгруппы R_{jk} . Тогда необходимо явно задать его длину и воспользоваться вариантом алгоритма для задачи упаковки контейнера, который представлен процедурой PackBin(W, \mathcal{L} , R, y), где \mathcal{L} – оставшаяся длина контейнера. В качестве этих процедур у нас выступает алгоритм РН для SPP и ВРР. Необходимо отметить, что можно использовать и другие эвристики для решения этих задач. Ограничения накладываются

только на вариант решаемой задачи (SPP и BPP). Не ограничивается применение разных эвристик с целью получения оптимального плана раскроя. Кроме того, свойства используемых алгоритмов сохраняются, например, если эвристика предназначена для решения задачи с фиксированной ориентацией прямоугольников (SPP|O, BPP|O) или с возможностью их поворота (SPP|R, BPP|R), и SM также будет решать этот вариант задачи. Это позволяет обеспечить широкий круг решаемых задач и вариативность свойств за счет выбора базовых эвристик.

Несмотря на порядок приоритетов в последовательности G, в процедуры упаковки передаются не подгруппы R_{ik} , а группы R_{jk} . Это делается для того, чтобы обеспечить заполнение пустого пространства прямоугольниками более низкого приоритета в том случае, если ни один прямоугольник соответствующего приоритета p не может быть упакован. В связи с этим необходима проверка мультимножества R_{jk} на пустоту, поскольку все его элементы могут быть упакованы ранее.

```
Algorithm 1 SM algorithm
```

```
1: procedure SM(L, W, R)
           L_{bins} \leftarrow \{0, j \leftarrow \overline{1, m}\}
           R' \leftarrow \{\}
 3:
           G \leftarrow \{(h_j, p_{jk}), k \leftarrow \overline{1, q_j}, j \leftarrow \overline{1, m}\}
 4:
           sort G descending h_i and increasing p_{ik}
 5:
           for all (h, p) \in G do
 6:
 7:
                if R_{ik} = \emptyset then
                      continue
 8:
 9:
                end if
10:
                y_c \leftarrow L_{bins}[h]
                \frac{g_c}{L} \leftarrow \frac{Lh_{bins}}{h}
11:
                (L', R'_i) \leftarrow \text{PackStrip}(W, R_i, y_c)
12:
13:
                if L' > \overline{L} then
                      (L', R'_i) \leftarrow \operatorname{PackBin}(W, \overline{L}, R_j, y_c)
14:
                end if
15:
                if R'_i = \emptyset or L' = 0 then
                      continue
17:
18:
                L_{bins}[h] \leftarrow L_{bins}[h] + L'
19:
                R' \leftarrow R' + R'_i
20:
                L \leftarrow \overline{L} - L'
21:
22:
                h_{bins} \leftarrow h
                if L=0 then
23:
                      break
24:
                end if
25:
26:
           end for
           return R', L_{bins}
28: end procedure
```

Рис. 2. Алгоритм SM

SM может «дробить» одну группу R_j на несколько частей и упаковывать их по отдельности. Несмотря на это, обеспечивается последовательность схемы раскроя каждой группы.

Наглядная демонстрация работы алгоритма приводится на рис. 3. Здесь видно, что размеры незаполненной части контейнера преобразуются с помощью (2) и выступают в качестве контейнера для упаковки более тонких прямоугольников.

Результаты исследования и их обсуждение

Последовательный метаэвристический (SM) алгоритм с использованием приоритетной эвристики (PH) был протестирован на 10 примерах, составленных специально для задачи BPP|P|M|IC|G. Примеры содержат от 3 до 14 несовместимых категорий и от 3 до 5 приоритетов. Примеры составлены таким образом, чтобы включать задачи с числом элементов из широкого диапазона. Так, наборы 1, 8 и 10 содержат относительно немного элементов (n < 1000), а крупномасштабные наборы 6 и 7 включают более 10 000 элементов.

Для возможности сопоставимости результатов работы алгоритма оценка качества упаковки каждой группы проводится аналогично [2] с помощью показателя $Gap_j = 100(L_j - L_j^*) / L_j^*$, где L_j^* — известная оптимальная длина контейнера или нижняя граница (LB), а L_j — длина контейнера, полученная с помощью SM, j=1,...,m. При этом длина контейнера не ограничивается, т.е. решается SPP. Результирующий показатель определяется как среднее взвешенное:

$$Gap = \sum_{j=1}^{m} Gap_{j} S_{j} / \sum_{j=1}^{m} S_{j} , \qquad (4)$$

где S_j — площадь всех прямоугольников j-й группы. На рис. 3 представлены результаты работы SM на SPP|P|M|IC|G. Из рис. 4 видно, что влияние метода сортировки алгоритма PH сохраняется.

В случае ВРР|Р|М|IС|G размер контейнера ограничен, что, несомненно, приводит к ухудшению качества работы вследствие влияния приоритета прямоугольников. Показатель *Gap* (4) для ВРР|Р|М|IС|G представлен на рис. 5, а. Можно заметить, что в некоторых случаях *Gap* (4) становится отрицательным. Это происходит, когда длина контейнера, выделенная алгоритмом SM для какой-либо группы, меньше оптимальной, соответственно некоторые прямоугольники не могут быть упакованы. Существенное отклонение наблюдается для примера 10.

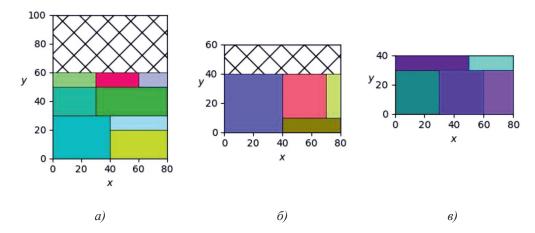
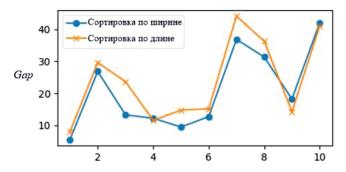


Рис. 3. Упаковка прямоугольников толщиной 3 мм (а), 2 мм (б) и 1 мм (в) на исходный лист с размерами $L_0=100,\,W_0=80,\,H_0=3$



 $Puc.\ 4.\ Peзультаты\ paбomы\ SM\ c\ PH\ npu\ paзных\ методах\ copmupoвки\ нa\ SPP|P|M|IC|G$

Он содержит несколько низкоприоритетных прямоугольников больших размеров. Несмотря на это, плотность заполнения каждого контейнера высока. На рис. 5, б приведена средняя относительная площадь для каждого примера:

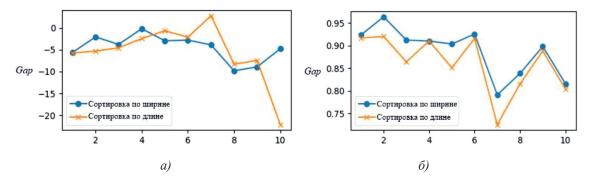
$$\overline{S} = \sum_{j=1}^{m} \overline{S_j} / m, \ \overline{S_j} = 100 \frac{S_j'}{L_j W}.$$

Здесь S_j — отношение площади размещенных прямоугольников к площади контейнера, где S_j — площадь размещенных прямоугольников, а L_j — длина контейнера. На рис. 5, а и б можно видеть, что метод сортировки по длине показывает результат хуже относительно сортировки по ширине.

Выводы

В работе ставится новая задача приоритетной упаковки прямоугольников с несовместимыми категориями и динамически изменяющимися размера контейнеров (ВРР|Р|М|IС|G). Эта задача определяется тем, что размеры исходного контейнера изменяются в процессе упаковки, при этом необходимо учитывать приоритетность

деталей, их несовместимость между собой и гильотинные ограничения. Для решения представленной задачи предлагается новый последовательный метаэвристический алгоритм (SM), который в процессе работы разделяет группы прямоугольников по их приоритетам и категориям. Полученные группы размещаются последовательно, при этом остается возможность заполнения свободного пространства прямоугольниками более низкого уровня. В качестве базового алгоритма используется однопроходной детерминированный эвристический алгоритм PH, показывающий хорошие результаты среди аналогов. Но SM не ограничивается рассмотренной эвристикой. В качестве базового алгоритма можно использовать любые алгоритмы, позволяющие решать ВРР и SPP. Свойства этих алгоритмов сохраняются. Так, в приведенных примерах показано, что при использовании РН его зависимость от варианта сортировки унаследована SM. Стоит отметить гибкость SM относительно выбираемых алгоритмов. Их можно комбинировать для достижения необходимых свойств.



 $Puc. 5. \ Peзультаты \ paбomы \ SM \ c \ PH \ npu \ paзных методах \ copmupo<math>6$ ки на SPP[P]M[IC]G

Качество работы SM было продемонстрировано на наборе примеров. Наборы различаются как количеством элементов, так и их распределением по категориям и приоритетам. При решении BPP|P|M|IC|G и SPP|P|M|IC|G алгоритм показывает хорошие результаты, а свойства решения зависят от свойств используемого базового алгоритма.

Поскольку, как нам известно, это первое исследование BPP|P|M|IC|G, может быть много возможных вариантов дальнейшего развития. Так, остается открытым вопрос о возможности изменения направления деформации контейнера. Возможности применения стохастических и детерминированных алгоритмов локальной и глобальной оптимизации и их комбинаций также широки. Одним из вариантов является распределение длин субконтейнеров с помощью глобальной оптимизации.

Список литературы

- 1. Wascher G., Haußner H., Schumann H. An improved typology of cutting and packing problems. European Journal of Operational Research. 2007. Vol. 183. № 3. P. 1109–1130.
- 2. Zhang D., Shi L., Leung S.C.H., Wu T. A priority heuristic for the guillotine rectangular packing problem. Information Processing Letters. 2016. V. 116(1). P. 15–21. DOI: 10.1016/j. ipl.2015.08.008.
- 3. Coffman E.G. et al. Bin Packing Approximation Algorithms: Survey and Classification. Handbook of Combinatorial Optimization / ed. Pardalos P.M., Du D.-Z., Graham R.L. New York, NY: Springer New York, 2013. Vol. 1–5. P. 455–531.
- 4. Guinet A., Chaabane S. Operating theatre planning. International Journal of Production Economics. 2003. Vol. 85. N_2 1. P. 69–81.

- 5. Jebali A., Hadj Alouane A.B., Ladet P. Operating rooms scheduling. International Journal of Production Economics. 2006. Vol. 99. № 1–2. P. 52–62.
- 6. Vijayakumar B., Parikh P.J. , Scott R., Barnes A., Gallimore J. A dual bin-packing approach to scheduling surgical cases at a publicly-funded hospital. European Journal of Operational Research. 2013. Vol. 224. № 3. P. 583–591.
- 7. Baldi M.M., Manerba D., Perboli G., Tadei R. A Generalized Bin Packing Problem for parcel delivery in last-mile logistics. European Journal of Operational Research. 2019. Vol. 274. № 3. P. 990–999.
- 8. Liang X., Zhou S., Chen H., Xu R. Pseudo transformation mechanism between resource allocation and bin-packing in batching environments. Future Generation Computer Systems. 2019. Vol. 95. P. 79–88.
- 9. Dósa G., Epstein L. The tight asymptotic approximation ratio of First Fit for bin packing with cardinality constraints. Journal of Computer and System Sciences. 2018. Vol. 96. № 2018 P 33–49
- 10. Khanafer A., Clautiaux F., Talbi E.G. Tree-decomposition based heuristics for the two-dimensional bin packing problem with conflicts. Computers and Operations Research. 2012. Vol. 39. N₂ 1. P. 54–63.
- 11. Muritiba A.E.F., Iori M., Malaguti E., Toth P. Algorithms for the bin packing problem with conflicts. INFORMS Journal on Computing. 2010. Vol. 22. № 3. P. 401–415.
- 12. Santos L.F.O.M., Yoshizaki H., Cunha C.B.D. A variable neighborhood search algorithm for the bin packing problem with compatible categories. Expert Systems with Applications. 2019. Vol. 124. P. 209–225.
- 13. Laurent A., Klement N. Bin Packing Problem with priorities and incompatibilities using PSO: Application in a health care community. IFAC-PapersOnLine. 2019. Vol. 52. № 13. P. 2596–2601.
- 14. Leung S.C.H., Zhang D. A fast layer-based heuristic for non-guillotine strip packing. Expert Systems with Applications. Elsevier Ltd, 2011. Vol. 38. № 10. P. 13032–13042.
- 15. Zhang D., Lijun Wei, Stephen C. H. Leung, Qingshan Chen. A binary search heuristic algorithm based on randomized local search for the rectangular strip-packing problem. INFORMS Journal on Computing. 2013. Vol. 25. № 2. P. 332–345.