ИНФОРМАТИКА

2008 октябрь-декабрь № 4

УДК 004.031

А.В. Головистиков

ЗАДАЧИ ДВУМЕРНОЙ ПРЯМОУГОЛЬНОЙ УПАКОВКИ И РАСКРОЯ: ОБЗОР

Приводится обзор результатов по решению задач двумерной прямоугольной упаковки и раскроя. Рассматриваются основные формулировки, математические модели и алгоритмы решения задач ортогональной упаковки прямоугольных предметов в контейнер или полубесконечную полосу и двумерного прямоугольного (в том числе гильотинного) раскроя. Основное внимание уделяется эвристическим алгоритмам и метаэвристикам (генетическим алгоритмам, поиску с запретами, метаэвристике муравьиной колонии).

Введение

Рассматриваются основные подходы к решению задач ортогональной упаковки прямоугольных предметов в контейнер или полубесконечную полосу (двумерной прямоугольной упаковки) и двумерного прямоугольного (в том числе гильотинного) раскроя. Под гильотинным раскроем понимается раскрой, полученный в результате последовательности сквозных резов, параллельных сторонам полосы.

Задачи двумерной прямоугольной упаковки и раскроя имеют широкое практическое применение: в полиграфическом производстве, машиностроении, при размещении элементов электронных схем, в мебельной и стекольной промышленности и т. д., к ним сводятся также задачи упаковки грузов рядами.

Задачи двумерной упаковки являются частным случаем *общей задачи упаковки*, которая состоит в том, что необходимо разместить имеющиеся малые элементы внутри больших объектов так, чтобы не было взаимного перекрытия, а некоторая заданная целевая функция достигала оптимума [1–6]. Сходными математическими моделями описываются задачи раскроя, возникающие при производстве изделий, когда материал раскраивается на части заданных размеров и конфигураций (на заготовки либо готовые детали) [7–9]. Классификация задач раскроя и упаковки дана в работе [10] и уточнена в [11].

1. Формулировки задач двумерной упаковки и раскроя

В задачах *двумерной прямоугольной упаковки* требуется без взаимного перекрытия разместить имеющиеся прямоугольные элементы внутри некоторых плоских объектов. В зависимости от вида объекта, внутри которого происходит размещение, задачи прямоугольной упаковки делятся на задачи упаковки в открытую область, в полубесконечную прямоугольную полосу или в заданные прямоугольники (контейнеры). В последних двух случаях обычно предполагается, что стороны размещаемых прямоугольников должны быть параллельны сторонам полосы или контейнера. Различают также задачи, в которых можно размещаемые прямоугольники поворачивать на 90° [12, 13] либо такой поворот запрещен [14]. На практике задачи с запрещением поворота возникают, например, при верстке материала в полиграфическом производстве или при раскрое декорированного либо структурированного материала в мебельной, стекольной или швейной промышленности. Задачи двумерной прямоугольной упаковки идентичны задачам двумерного прямоугольного *раскроя*, в которых требуется раскроить некоторые плоские объекты на заданные прямоугольные элементы (если рассматривать их математическую сторону и не учитывать некоторые технологические ограничения, такие, как ширина реза, припуск под прижим и пр.), поэтому в дальнейшем термины «упаковка» и «раскрой» будем рассматривать как взаимозаменяемые.

Перейдем к формальному описанию задач. Пусть необходимо упаковать прямоугольники, которые могут отличаться по типу, с заданным для каждого типа i, $i = 1, \dots, m$, количеством b_i и размерами (w_i, l_i) прямоугольников.

В задаче *двумерной упаковки в открытую область* (Two-dimensional Area Packing, 2DAP) предполагается, что заданные прямоугольники надо разместить без пересечения так, чтобы площадь прямоугольника, огибающего размещаемые элементы, была минимальной.

Задача *двумерной упаковки в полубесконечную полосу* шириной W (Two Dimensional Strip Packing, 2DSP) состоит в том, что требуется разместить прямоугольники в полосе минимальной длины (или минимальной высоты при вертикальной полосе) при следующих условиях:

– стороны прямоугольников параллельны сторонам полосы;

– прямоугольники не пересекаются друг с другом;

– прямоугольники не пересекаются со сторонами полосы.

Под сторонами полосы, кроме двух ее параллельных сторон, понимается также и торец полосы. Совпадение сторон прямоугольников между собой или со сторонами полосы не рассматривается как пересечение. На рис. 1 показана двумерная упаковка в горизонтальную полосу.

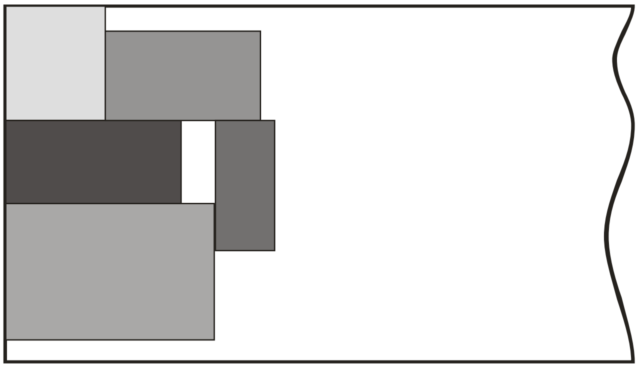

Рис. 1. Пример упаковки в горизонтальную полосу

Задача *гильотинного раскроя* (Guillotines Cutting, GC) состоит в том, что требуется найти такую последовательность сквозных резов, параллельных сторонам полосы, для разрезания полосы на заданные прямоугольники в указанном количестве, чтобы длина отрезанной части полосы была минимальной. При этом, естественно, будут выполняться условия параллельности сторон прямоугольников сторонам полосы и непересечения прямоугольников друг с другом и со сторонами полосы.

Очевидно, что, с одной стороны, гильотинный раскрой является частным случаем двумерного прямоугольного раскроя (упаковки) и, с другой стороны, не каждый прямоугольный раскрой является гильотинным.

На рис. 2 приведен пример раскроя полубесконечной полосы гильотинными резами, а рис. 1 можно рассматривать как пример негильотинного раскроя.

Задача *двумерной упаковки в контейнеры* (Two Dimensional Bin Packing, 2DBP) состоит в том, что требуется разместить заданные прямоугольники в наименьшее число одинаковых контейнеров шириной W и длиной L.

Следует отметить также задачу *двумерного раскроя* (Two Dimensional Cutting Stock), состоящую в нахождении размещения (без самопересечений) прямоугольников размера (w_i, l_i) в прямоугольной области (контейнере) шириной W и длиной L, которое максимизирует суммарную площадь размещенных прямоугольников. Другая разновидность этой задачи состоит в том, что размещаемые элементы имеют коэффициенты полезности (для элемента i, $i = 1,\dots,m$, задана полезность p_i) и необходимо максимизировать общую выгоду размещения всех элементов. Эта разновидность задачи была введена P.C. Gilmore и R.E. Gomory [9] под названием *квадратный рюкзак* (Two Dimensional Cutting Knapsack). Если ориентация размещаемых прямоугольников фиксирована и разрешены только гильотинные размещения (т. е. такие, для которых существует прямая, параллельная одной из сторон контейнера, разделяющая все размещения без пересечения внутренности прямоугольников на две части, каждая из которых обладает тем же свойством), то задача называется *гильотинным размещением на паллете* (Guillotine Pallet Loading) [15, 16].

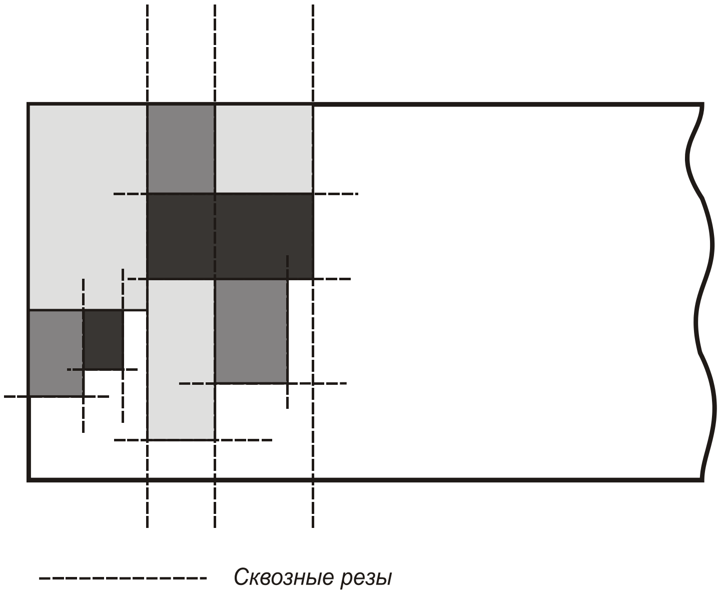

Рис. 2. Пример раскроя полосы гильотинными резами

Несмотря на разнообразие постановок задач двумерной упаковки и раскроя, все они NP-трудны в сильном смысле, поскольку легко построить полиномиальное сведение к ним следующей задачи одномерной упаковки (One Dimensional Bin Packing, 1BP), для которой известно, что она NP-трудна в сильном смысле [17]. Имеется *n* элементов размера h_i, которые необходимо разбить на минимальное число подмножеств так, чтобы сумма размера каждого подмножества не превышала заданного размера *H*.

Применяемые для решения задач двумерной упаковки и раскроя методы можно разделить на три основные группы:

– точные алгоритмы (для задач малой размерности или для частных случаев задач, которые оказываются полиномиально разрешимыми);

– простые эвристики;

– метаэвристические алгоритмы.

Поскольку задачи двумерной прямоугольной упаковки являются NP-трудными, для решения задач, возникающих на практике, обычно применяются эвристические алгоритмы полиномиальной сложности, что позволяет получать рациональное решение за приемлемое время.

2. Модели задач и точные алгоритмы решения

Первая попытка моделирования задачи рационального раскроя была предпринята Л.В. Канторовичем и В.А. Залгаллером в работе, проведенной в 1948-1950 гг. на Ленинградском вагоностроительном заводе. В их монографии, вышедшей в 1951 г. и переизданной в 1971 г. [18], обобщен опыт этой работы. Как отмечает Л.В. Канторович в своей автобиографии [19], представленной в Нобелевский комитет в связи с присуждением ему премии по экономике в 1975 г., в этой монографии «дается не только более систематическое изложение алгоритмов линейного программирования, но также используется для этих задач (независимо от Беллмана) идея динамического программирования (задача о раскрое) и комбинирование его с линейным». Фактически в ходе работы по раскрою был создан метод последовательного улучшения планов, позже названный модифицированным симплекс-методом линейного программирования, и попутно – идея шкалы, предвосхитившая динамическое программирование Беллмана.

В работе P.C. Gilmore и R.E. Gomory [9] предложена следующая модель целочисленного линейного программирования (ЦЛП) для задачи двумерной упаковки 2DBP, являющаяся обобщением подхода к задаче одномерной упаковки 1BP [20, 21]. Пусть A_j – бинарный вектор-столбец с компонентами a_{ij} $(i=1,\ldots,n)$, принимающими значение 1, если элемент *i* принадлежит *j*-му подмножеству элементов, которые попадают в один контейнер (*j*-му образцу), и 0 в противном случае. Множество всех возможных образцов представляется тогда матрицей *A* из всех возможных столбцов A_j, $j=1,\ldots,M$, а математическая модель задачи имеет вид

$$\min \sum_{j=1}^{M} x_j$$

при условиях

$$\sum_{j=1}^{M} a_{ij} x_j = 1, \; i=1,\ldots,n;$$

$$x_j \in \{0,1\}, \; j=1,\ldots,M,$$

где x_j принимает значение 1, если образец j принадлежит решению, и 0 в противном случае.

Ввиду огромного числа столбцов, которые могут быть в матрице *A*, единственным путем работы с этой моделью является динамическое генерирование столбцов по мере надобности. Для соответствующей задачи одномерной упаковки 1BP предложен подход динамического программирования [20, 21], в случае же задачи 2BP такого вида подходы неэффективны. В связи с этим в [9] предлагается решение методом динамического программирования в частном случае задачи, где имеет место поуровневое размещение элементов.

Другие формулировки задач двумерной прямоугольной упаковки в виде задач ЦЛП предложены в [22, 23]. Эти модели позволяют находить верхние границы оптимального решения с помощью лагранжевой релаксации и субградиентной оптимизации. Отметим, что модели [9, 22, 23] содержат неполиномиальное число переменных.

A. Lodi, S. Martello и D. Vigo [24] предлагают модель ЦЛП с полиномиальным числом переменных для такого частного случая задач двумерной упаковки 2DSP (в вертикальную полосу) и 2DBP (в контейнеры), в котором упаковка идет по уровням. Первый уровень пакуется (по какому-то способу, который определяется алгоритмом упаковки) на дно полосы (контейнера). Каждый следующий уровень пакуется на горизонтальной линии, проведенной по верху самого высокого упакованного элемента предыдущего уровня. Поскольку уровни и элементы можно перенумеровать, то без потери общности можно считать, что на каждом уровне элементы упакованы слева направо по невозрастанию высоты, уровни упорядочены снизу вверх тоже по невозрастанию высоты, и сами элементы пронумерованы по невозрастанию высоты. Пример такой перенумерации двухуровневой упаковки показан на рис. 3.

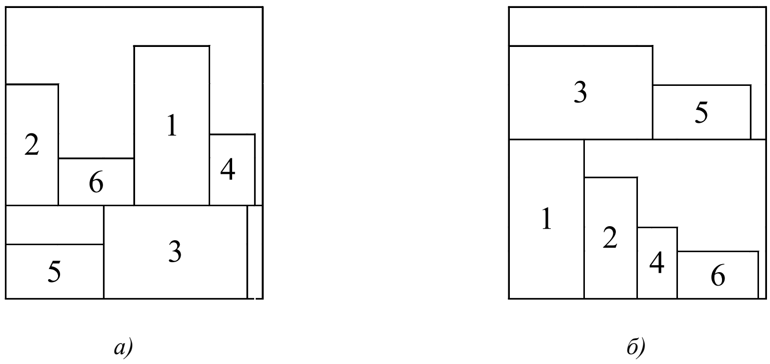

Рис. 3. Двухуровневая упаковка: *а)* до; *б)* после перенумерации и переупорядочивания элементов

При условии поуровневой упаковки модель ЦЛП для задачи 2DBP строится в [24] в предположении, что имеется *n* пакуемых элементов, *n* потенциальных уровней (причем *i*-й уровень ассоциируется с элементом *i*, с которого начинается упаковка на этом уровне; будем говорить, что *i*-й элемент инициализирует *i*-й уровень) и имеется *n* потенциальных контейнеров (причем *k*-й контейнер ассоциируется с *k*-м потенциальным уровнем, с которого начинается упаковка в этом контейнере; будем говорить, что *k*-й уровень инициализирует *k*-й контейнер). Вводятся двоичные переменные y_i (и соответственно q_k), принимающие значение 1, если элемент *i* инициализирует уровень *i* (и соответственно уровень *k* инициализирует контейнер *k*), и значение 0 в противном случае. Тогда модель ЦЛП имеет вид

$$\min \sum_{k=1}^{n} q_k$$

при условиях

$$\sum_{i=1}^{j-1} x_{ij} + y_j = 1 \quad (j = 1, \ldots, n),$$

$$\sum_{j=i+1}^{n} w_j x_{ij} \le (W - w_i) y_i \quad (i = 1, \ldots, n-1),$$

$$\sum_{k=1}^{i-1} z_{ki} + q_i = y_i \quad (i = 1, \ldots, n),$$

$$\sum_{i=k+1}^{n} h_i z_{ki} \le (H - h_k) q_k \quad (k = 1, \ldots, n-1),$$

$$y_i, x_{ij}, q_k, z_{ki} \in \{0, 1\} \quad \forall i, j, k,$$

где x_{ij} (при $i \ne n,\ \ j > i$) и соответственно z_{ki} (при $k \ne n$, $i > k$) принимают значение 1, если элемент *j* попадает в уровень *i* и соответственно уровень *i* попадает в контейнер *k*, а в противных случаях принимают значение 0. Первое условие обеспечивает упаковку каждого из элементов только в один из уровней. Второе условие гарантирует, что суммарная ширина элементов в каждом возникшем уровне не превосходит ширины контейнера. Третье условие гарантирует, что каждый из возникающих уровней попадает только в один из контейнеров (и, возможно, инициализирует контейнер). Четвертое условие гарантирует, что суммарная высота уровней в каждом контейнере не превосходит высоты контейнера. В работе [24] показано, как эта модель трансформируется в случае задачи 2DSP, и приведены результаты вычислительного эксперимента для обеих моделей, позволяющие говорить об их применимости для разумных с практической точки зрения размерностей задач. Естественно, обе модели можно использовать и для получения нижних границ оптимальных решений для задач большей размерности, если отбросить условие целочисленности переменных.

И.В. Романовским [25] предлагается следующая модель задачи гильотинного раскроя плоского листа на прямоугольные заготовки в виде задачи, представляющей собой часть одной итерации симплекс-метода (и решение ее методом переработки списка состояний). Задан плоский лист размером $W \times L$ и *m* деталей размером $w_i \times l_i$, $i = 1, \ldots, m$. Стоимость *i*-й детали c_i. Требуется найти наименьшее неотрицательное решение функционального уравнения

$$f(w,l) = \max\{\max_{0 \le x \le w}(f(x,l) + f(w-x,l)), \max_{0 \le x \le l}(f(w,x) + f(w,l-x))\}$$

с дополнительными условиями

$$f(w,0) = f(0,l) = 0\,,\ f(w_i,l_i) \ge c_i\,.$$

Функция $f(w,l)$, являющаяся наименьшим решением этого уравнения, допускает представление в виде $f(w,l) = \max_{j:\, w_j \le w,\, l_j \le l} f_j$, где $\{(w_j, l_j, f_j)\}$ – некоторый набор точек и $w_j \le W$, $l_j \le L$. Идея состоит в последовательном пересчитывании этого набора для итерирования решения функционального уравнения. Это оказывается проще и экономнее, чем запоминание решения функционального уравнения в узлах сетки при сеточном методе решения.

В работе [26] предлагается теоретико-графовая модель для задачи двумерной прямоугольной упаковки, в которой для представления упаковки в контейнер вводятся интервальные графы $G_w = (V, E_w)$ и $G_l = (V, E_l)$ так, что вершина v_i соответствует элементу i упаковки, а ребро между вершинами v_i и v_j имеет место тогда и только тогда, когда проекции элементов i и j на вертикальную ось (для графа G_w) и горизонтальную ось (для графа G_l) пересекаются. При этом если упаковка допустима, то $E_w \cap E_l = \varnothing$ и для каждого независимого множества S графа G_w (соответственно графа G_l) выполняется $\sum_{v_j \in S} w_j \le W$ (и соответственно $\sum_{v_j \in S} l_j \le L$ для графа G_l). В работе [27] показано, что одному интервальному графу соответствует некоторое семейство упаковок, и предложен алгоритм, осуществляющий перебор указанных графов.

И.В. Романовский предложил общую идею перебора с отсечением неперспективных вариантов при решении задач упаковки методом ветвей и границ [28]. Различные версии метода ветвей и границ рассмотрены в работах S. Martello и P. Toth и обобщены в их монографии [29]. Для получения точного решения использовались и методы отсечения в сочетании с эвристикой последовательного уточнения оценок [30]. Точный метод решения, рассчитанный на небольшое число прямоугольников и основанный на понятии «зоны», был предложен А.И. Липовецким [31]. Метод сводится к перебору допустимых зон, его реализация с использованием ряда правил отсечений бесперспективных ветвей приведена в работе [32].

В работе [33] предлагается новый переборный метод на базе матричного представления прямоугольной упаковки для задачи упаковки в полубесконечную полосу (задачи 2DSP). Первоначально упаковка разбивается на вертикальные и горизонтальные полосы. Далее каждому разбиению сопоставляется бинарная матрица с элементами $a_{ij} = 1$, если i-й прямоугольник пересекается с j-й полосой, в противном случае $a_{ij} = 0$. Алгоритм трансформируется также на задачу упаковки n-мерных параллелепипедов.

Два точных метода решения задачи двумерной прямоугольной упаковки предложены в работе [34]. Первый метод является улучшением классического метода ветвей и границ, предложенного в [35]. Вместо упаковки прямоугольников в каждую возможную позицию предлагаемый алгоритм предусматривает упаковку в левую нижнюю позицию, а также проверяет возможность не упаковывать никакой прямоугольник в эту позицию. Второй метод основан на новом способе релаксации задачи, отличном от релаксаций, предложенных в работах [36, 37]. В этом методе на первом шаге фиксируются только x-координаты размещаемых элементов с использованием точного алгоритма. Для каждой найденной конфигурации на втором шаге находят список y-координат размещаемых элементов. Вычислительный эксперимент показал, что предлагаемые методы сравнимы по эффективности с графовым подходом [27, 38] и превосходят остальные известные точные алгоритмы. Тем не менее, несмотря на многообразие различных подходов к созданию точных алгоритмов, с их помощью удается решать задачи с небольшим числом (≤ 20) размещаемых прямоугольников.

Для отдельных специальных случаев задач упаковки и раскроя предложены полиномиальные точные алгоритмы решения, позволяющие решать задачи значительной размерности. Здесь следует отметить работы А.Г. Тарновского по двумерному гильотинному раскрою в случае, когда все прямоугольники, на которые производится раскрой, имеют одинаковую размерность при различной ортогональной ориентации, либо в случае двух типов прямоугольников при фиксированной ориентации [15, 16, 39]. Эти результаты применимы также для решения некоторых задач теории расписания с параллельными приборами [40].

3. Приближенные алгоритмы

Основной вывод, который можно сделать из предыдущего раздела, состоит в том, что в силу NP-трудности большинства задач упаковки и раскроя известные точные алгоритмы (ветвей и границ, последовательного анализа вариантов и динамического программирования) не позволяют решать за приемлемое время задачи, размерность которых соответствует требованиям практических приложений. Альтернативой точным алгоритмам могут служить так называемые градиентные алгоритмы или, точнее, градиентные аналоги дискретных алгоритмов. Эти алгоритмы дают возможность быстро и с довольно высокой точностью решать задачи большой размерности. Эффективность градиентных алгоритмов заметно увеличивается, если использовать их в виде серий конкурирующих алгоритмов [41].

Погрешностью приближенного алгоритма *A* решения задачи на максимум называется величина

$$\Delta_A = \frac{F^* - F_A}{F^*},$$

где F^* и F_A – значения целевой функции соответственно на оптимальном и на построенном с помощью алгоритма *A* решениях задач.

В качестве примера рассмотрим оценки погрешностей градиентных алгоритмов при решении задачи двумерного гильотинного раскроя, т. е. задачи определения такого размещения (укладки) заданных прямоугольников в прямоугольном листе, которое максимизирует суммарную площадь уложенных прямоугольников. Напомним, что при гильотинном размещении существует прямая, параллельная одной из сторон листа, разделяющая все размещение на две части, каждая из которых обладает тем же свойством. Назовем разбиением гильотинной укладки на фрагменты такое разбиение листа на прямоугольные части (фрагменты), при котором каждый фрагмент содержит ровно по одному прямоугольнику из укладки и не пересекает внутренность других прямоугольников укладки. Нетрудно видеть, что для любой гильотинной укладки существует ее разбиение на фрагменты. Действительно, если удалить из множества отрезков гильотинной укладки те, которые отделяют части листа, не содержащие прямоугольников, и продлить некоторые из оставшихся отрезков, получим разбиение укладки на фрагменты. Совершенной укладкой (или плотным размещением) называют гильотинную укладку, в любом разбиении которой в каждый фрагмент невозможно без изменения местоположения содержащегося в нем прямоугольника разместить дополнительно такой же прямоугольник. Всякий «разумный» алгоритм гильотинного раскроя строит размещения, в которые невозможно добавить ни одного прямоугольника, не изменив местоположения уже размещенных, и, следовательно, строит совершенную укладку. В работе [42] показано, что для погрешности любого алгоритма, строящего совершенную укладку, справедливы следующие оценки:

1) $\Delta_A < \frac{8}{9}$, если прямоугольники во фрагментах разбиения располагаются произвольно;

2) $\Delta_A < \frac{5}{6}$, если не менее одной стороны любого фрагмента некоторого разбиения инцидентны сторонам содержащегося в нем прямоугольника;

3) $\Delta_A < \frac{3}{4}$, если не менее двух сторон любого фрагмента некоторого разбиения инцидентны сторонам содержащегося в нем прямоугольника;

4) $\Delta_A < \frac{2}{3}$, если не менее двух сторон любого фрагмента некоторого разбиения инцидентны противоположным сторонам содержащегося в нем прямоугольника;

5) $\Delta_A < \frac{1}{2}$, если не менее трех сторон любого фрагмента некоторого разбиения инцидентны сторонам содержащегося в нем прямоугольника;

6) $\Delta_A = 0$ (получено точное решение), если все четыре стороны любого фрагмента некоторого разбиения совпадают со сторонами содержащегося в нем прямоугольника.

Доказательство всех утверждений проводится по единой схеме, например, если не менее двух сторон любого фрагмента инцидентны сторонам содержащегося в нем прямоугольника, то в худшем случае этот прямоугольник содержится во фрагменте с двумя смежными сторонами. Следовательно, площадь любого фрагмента не может быть более чем в четыре раза больше площади содержащегося в нем прямоугольника, иначе в нем можно было бы уложить хотя бы еще один прямоугольник. Это означает, что $F_A > \frac{1}{4}LW$, где L – длина, а W – ширина листа. Учитывая, что $F^* \le LW$, получаем $\Delta_A < \frac{3}{4}$.

В работе [9] изучается задача, называемая двухстадийной задачей раскроя: в листе длиной L и шириной W необходимо разместить прямоугольники размера $l_i \times w_i$, $i = 1, \dots, n$, в две стадии, сначала разбивая лист на полосы ширины l_i или w_i прямыми, параллельными некоторой стороне листа, а потом разрезая полосы на одинаковые прямоугольники размера $l_i \times w_i$ или $w_i \times l_i$.

Для решения этой задачи в [42, 43] предложена серия градиентных алгоритмов с параметром α, имеющих трудоемкость $O(n^2)$, которая строится следующим образом. Для каждого прямоугольника $l_i \times w_i$ определяют градиенты полос $l_i \times W$ и $L \times w_i$ по правилам

$$grad_i^W = (W - W \bmod w_i) l_i \left[\left\lfloor \frac{L}{l_i} \right\rfloor + \alpha \left(\frac{L}{l_i} \right) \right]$$

и

$$grad_i^L = (L - L \bmod w_i) w_i \left[\left\lfloor \frac{W}{w_i} \right\rfloor + \alpha \left(\frac{W}{w_i} \right) \right].$$

Пусть индексы i_W и i_L отвечают максимальным градиентам полос. Затем отрезают от левого нижнего угла листа $\left\lfloor \frac{L}{l_{i_L}} \right\rfloor$ полос размера $l_{i_L} \times W$ (соответственно $\left\lfloor \frac{W}{w_{i_W}} \right\rfloor$ полос размера $L \times w_{i_W}$). Полученные полосы раскраиваются на прямоугольники соответствующего размера. К оставшейся части листа рекурсивно применяют описанную процедуру. Для погрешности этой серии алгоритмов установлены следующие оценки [42]:

1) $\Delta < \frac{25}{61}$ при $\alpha = 0$;

2) $\Delta < \frac{3}{7}$ при $\alpha = \frac{1}{2}$;

3) $\Delta < \frac{1}{2}$ при $\alpha = 1$;

4) $\Delta < \frac{1}{3}$ для серии из двух алгоритмов при $\alpha = 0$ и $\alpha = 1$.

В работе [42] предложены также приближенные алгоритмы с оценками для решения задачи многостадийного раскроя, в которой число стадий более двух и раскраиваются также остатки полос. Приближенные алгоритмы положены в основу промышленной системы раскроя с использованием персональных ЭВМ [44].

4. Эвристики и метаэвристики

В силу NP-трудности задачи двумерной прямоугольной упаковки для ее решения применяются эвристические алгоритмы полиномиальной сложности, что позволяет получить рациональное решение за приемлемое время. Обзоры эвристических методов решения задач прямоугольной упаковки представлены в статьях A. Lodi, S. Martello и D. Vigo [12, 13], А.С. Филипповой [45], Э.Ф. Мухачевой и др. [46]. Отметим разработанные на базе уровневой стратегии детерминированные эвристики: алгоритмы размещения в соответствии с приоритетным списком по принципу *следующий подходящий* (Next Fit, NF), *первый подходящий* (First Fit, FF), *лучший подходящий* (Best Fit, BF) [13]. Основная безуровневая стратегия *нижний левый* (Bottom Left, BL) состоит в упаковке элемента в самую нижнюю позицию с выравниванием по левому краю [47]. E.G. Coffman и др. [48] проанализировали стратегии NF и FF при выборе элементов по убыванию их длины и определили асимптотическое поведение в худшем случае. При условии, что длины нормализованы так, что $\max_j l_j = 1$, получаем $F_{NF} = 2F^* + 1$ и $F_{FF} = 1{,}7F^* + 1$, где F_{NF} и F_{FF} – значения целевой функции, полученные соответствующим алгоритмом, а F^* – оптимальное значение. Для стратегии BL в [49] показано, что при выборе элементов по невозрастанию ширины имеет место оценка $F_{BL} = 3F^*$.

При построении эвристических алгоритмов упаковок часто выполняются следующие процедуры: кодирование схемы упаковки (представление решения), декодирование (получение упаковки из кодированного решения), осуществление стратегии поиска (поиск лучшего кодированного решения) [45]. Прямой код в виде последовательности координат размещаемых прямоугольников однозначно определяет упаковку. Близким к прямому является блочный код [50], причем различают кодирование одной блочной структурой и парой – вертикальной и горизонтальной структурами, когда через правые стороны прямоугольников проводят вертикальные линии, разбивающие упаковку на прямоугольные вертикальные блоки, а через верхние стороны прямоугольников проводят горизонтальные линии, разбивающие упаковку на горизонтальные блоки. Популярными схемами кодирования являются перестановки элементов [45], когда каждому прямоугольнику присваивают позицию в перестановке, и кодирование парой последовательностей, предложенное в работе [51]. Алгоритмы-декодеры [45, 52, 53] строят упаковку по ее коду, при этом каждой перестановке (в отличие от прямой схемы) соответствует несколько различных упаковок и однозначный выбор упаковки осуществляет тот или иной декодер.

В последнее время для решения задач раскроя и двумерной прямоугольной упаковки стали применять различные метаэвристики: генетические алгоритмы, поиск с запретами, имитацию отжига, алгоритм муравьиной колонии (Ant Colony). Генетические алгоритмы представляют собой мощное средство решения задач дискретной оптимизации и широко распространены при решении задач теории расписаний, экстремальных задач на графах, задач проектирования, компоновки и размещения, при решении систем логических уравнений, маршрутизации перевозок и др. [54–57]. В генетических алгоритмах решение задачи представляется *хромосомой* (ее иногда называют *записью*, *фреймом*), состоящей из *генов* (*полей*, *слотов*). Местоположение определенного гена в хромосоме является *локусом*, а альтернативные формы одного и того же гена, расположенные в одинаковых локусах хромосомы, называются *аллелями*. Аллели определяют значения генов (параметров задачи). С помощью генетических операторов (*выбора родителей*, *кроссовера*, *мутации*, *селекции*, *переупорядочения*) осуществляется направленный перебор решений. Согласно хромосомной теории наследственности передача качественных принципов, закодированных в генах, осуществляется через хромосомы от родителей к потомкам. Хромосома, содержащая в своих локусах конкретные значения аллелей, является *генотипом*. Конечное множество всех допустимых генотипов образует *генофонд*.

При решении задачи генетическим алгоритмом прежде всего формируется исходное поколение, состоящее из некоторого числа хромосом, затем случайным образом выбираются пары родителей среди хромосом данного поколения, при этом увеличивается вероятность выбора хромосом, соответствующих лучшим значениям целевой функции. Так начинается цикл эволюции. Следующее поколение образуется из перспективных дочерних хромосом, являющихся результатом операций кроссовера. Кроссовер заключается в разрыве двух родительских хромосом и комбинировании получившихся хромосомных отрезков (при этом не любой результат кроссовера является допустимой хромосомой, поэтому для корректировки неправильных хромосом используются операторы переупорядочения). Мутации (случайные изменения некоторых аллелей) служат для исключения застревания поиска в локальных минимумах. Мутация заключается в замене значений некоторых генов на случайные значения. Иногда аллелями служат не значения параметров, а имена эвристик, используемых для определения этих значений [57]. Например, макромутацией называют присвоение избранным генам случайных значений из диапазона номеров эвристик при некоторых вероятностях задания эвристик. В качестве эвристик обычно используют простые полиномиальные алгоритмы, например упомянутые ранее алгоритмы NF, FF, BF, BL и т. п. [47].

Для задач двумерной упаковки в полубесконечную полосу (2DSP) шириной W в [58] предложен генетический алгоритм блочной структуры, в котором упаковка разбивается на прямоугольные блоки ширины W и различной длины; при этом генами являются блоки, хромосомами – блок-структуры, аллели генов формируют путем перестановки прямоугольников внутри блока, кроссовер фиксирует родительские гены и осуществляет скрещивание соответствующих им блоков, мутация изменяет блок-структуру, начальную блок-структуру получают с помощью алгоритма [59] последовательного уточнения оценок (Sequencial Value Correction method, SVC). Популяция соответствует совокупности всех допустимых решений. Степенью приспособленности особи является значение длины занятой части полосы прямоугольной упаковки: чем меньше эта длина, тем лучше особь приспособлена к внешней среде. В работе [57] предлагается другая интерпретация генетического алгоритма для решения задач упаковки: ген здесь означает способ упаковки очередного прямоугольника, аллели – последовательности алгоритмов для его упаковки. Эта идея использована в генетическом алгоритме для решения задач гильотинного раскроя [60], в котором структура гильотинного раскроя представляется как последовательность вложенных друг в друга прямоугольных блоков, а последовательность аллелей – это различные алгоритмы типа FF, NF, BF (например, в алгоритме FF – *первый подходящий* – в блок укладывается один или несколько равных прямоугольников, а затем формируется новый блок). Возможно также использование эвристики *худший подходящий* (Worst Fit, WF) или «жадного» алгоритма (в котором наилучшим образом заполняется боковая грань очередного блока); при этом можно задавать вероятность применения той или иной эвристики. Таким образом, в основе генетического гильотинного алгоритма лежит идея разбиения задачи на p подзадач, где p – число заполненных так называемых *корзин* (вложенных друг в друга прямоугольных блоков). Начальной корзиной является весь рулон, в качестве подзадач рассматриваются однородные размещения прямоугольников с помощью одной из вышеприведенных эвристик. Выбор способа раскроя сводится к задаче поиска алгоритма из заданных эвристик, при этом случайный перебор решений осуществляется генетическим алгоритмом. Его главной особенностью является то, что аллелями служат не значения параметров, а имена эвристик, использованных для определения этих значений, т. е. аллелями являются номера эвристик, а ген соответствует подзадаче.

В работе [61] предложен генетический алгоритм в применении к задаче упаковки в контейнеры, а в работе [62] – в применении к задаче двумерной упаковки с запретом ротаций размещаемых прямоугольников. Различные простые эвристики, применяемые в генетических методах, описаны в [63]. Сравнение эффективности различных подходов к построению генетических алгоритмов для решения задач двумерной упаковки приведено в работе [64].

Поиск с запретами (Tabu Search) является метаэвристикой, позволяющей осуществлять эффективный поиск на множестве решений в задачах оптимизации [65]. Основными элементами поиска с запретами являются процедуры перехода от текущего решения к соседним и оценки полученного решения, а главное – *списки запретов*, которые позволяют избегать локальных минимумов, отсекать неперспективные ветви поиска, обеспечивать неповторение исследования уже изученных областей. Существенными моментами являются *интенсификация* и *диверсификация* поиска; целью первой является обеспечение поиска в наиболее перспективных областях, где с большей вероятностью может содержаться глобальный оптимум, а второй – избежание зацикливания и обеспечение исследования всего пространства решений [66]. Обычно интенсификация заключается в возврате к одному из лучших ранее полученных решений, а диверсификация – в осуществлении нескольких случайных переходов или в уменьшении ценности переходов к часто посещаемым областям поиска.

В работе [67] предлагается применение поиска с запретами к задаче двумерного гильотинного раскроя. Процедура оценки решений базируется на использовании коэффициента раскроя, равного отношению полезной площади ко всей использованной. В качестве исходного используется раскрой, полученный модифицированным алгоритмом FF. Процедура перехода от одного решения к другому заключается в замене двух деталей в различных группах друг на друга. Имеется два списка запретов: в *глобальном* содержатся пары деталей, замена которых в текущем лучшем раскрое является началом ветви преобразований, в *локальном* содержится информация о последних произведенных заменах. Содержащиеся в локальном списке пары деталей не могут участвовать в дальнейших преобразованиях, за исключением случая, когда такое преобразование приводит к улучшению раскройного плана по сравнению с текущим. Интенсификация заключается в возврате к лучшему из ранее полученных планов раскроя после определенного числа операций, а диверсификация – в случайной замене двух деталей без учета оценок и списков запретов. В работе [68] предложен алгоритм поиска с запретами, в котором процедуры интенсификации и диверсификации основываются на долговременной памяти. Опыт использования поиска с запретами показал, что при росте числа деталей он предпочтительнее генетических алгоритмов, но при увеличении числа деталей до 500 затраченное время становится слишком большим и в этом случае приходится прибегать к простым эвристикам.

Метаэвристика муравьиной колонии появилась в результате наблюдения и анализа поведения реальных муравьиных колоний [69]. Муравьи способны находить кратчайшие пути между источником пищи и муравейником, помечая путь капельками пахучей жидкости (феромоном), привлекающими других муравьев. Встретив феромонный след, муравей с большей вероятностью пойдет по нему, тем самым усиливая след собственным феромоном. Муравьи, выбирающие короткий путь к пище, быстрее возвращаются в муравейник, в результате чего концентрация феромона на коротком пути увеличивается, и все большее количество муравьев начинают использовать этот путь. Вероятностная модель хорошо описывает особенности организации муравьиной колонии. В метаэвристике муравьиной колонии составная часть решения задачи называется фрагментом, аналогом муравьев являются независимые агенты – алгоритмы, итеративно строящие из множества фрагментов допустимое решение задачи. Феромон (численная характеристика фрагмента или перехода между фрагментами) показывает, насколько часто данный фрагмент (или переход) входил в лучшие решения на предыдущих итерациях алгоритма. Локальная информация показывает полезность фрагмента для построения хорошего решения. Феромон обновляется на каждой итерации, при этом на фрагментах и переходах хороших решений уровень феромона растет, а на остальных уменьшается. На каждой итерации все агенты строят решения, пытаясь улучшить решения, найденные на предыдущих итерациях. Выбор фрагмента решения осуществляется агентом на основе следов феромона и локальной информации. После достижений условий останова (по заданному числу итераций, например) получаем множество рекордных решений с наилучшим значением целевой функции.

В работе [70] предлагается применение метаэвристики муравьиной колонии к задачам двумерной упаковки. Фрагментом здесь является заданный размещаемый прямоугольник, а локальная информация перехода от последнего фрагмента, добавленного агентом в свое решение, к фрагменту, который является кандидатом на добавление в решение, равна площади фрагмента-кандидата. Феромон, величина которого связана с коэффициентом раскроя полученных решений, наносится на фрагменты и переходы рекордных решений, а также на фрагменты (переходы) лучших решений итерации. Чтобы излишнее накопление феромона на фрагментах (переходах) рекордных решений не приводило к ранней стагнации поиска (т. е. к ситуации, когда все агенты строят одинаковые решения), введены два параметра $\tau_{\min}$ и $\tau_{\max}$, ограничивающие возможные значения феромона интервалом между минимальным $\tau_{\min}$ и максимальным $\tau_{\max}$ значениями. Численный эксперимент с использованием генератора задач раскроя и упаковки [71, 72] и тестовых задач библиотеки OR-library [73] показал достаточную эффективность подхода при решении задач размером до 200 размещаемых прямоугольников.

Заключение

При выборе подходов к решению задач двумерной упаковки и раскроя, особенно при использовании эвристических алгоритмов и метаэвристик, существенную роль играет проведение численных экспериментов. Экспериментальному исследованию эффективности алгоритмов посвящено небольшое число специальных статей, например [59, 72, 74, 75]. При проведении таких экспериментов интересно выявить «плохие» задачи, для которых любые алгоритмы дают низкие показатели (например, задачи с длинными прямоугольниками), и специфические задачи, для которых более пригоден тот или иной тип алгоритмов. Показателем эффективности при проведении таких экспериментов обычно является коэффициент раскроя. При решении практических задач большой размерности чаще предпочтительнее простые эвристики, но учитывающие специфические особенности задач. Выявление таких специфических особенностей задач, например, для решения задач укладки при компьютерной верстке или при размещении элементов электронных схем представляет особый интерес.

Список литературы

1. Dyckhoff, H. Cutting and Packing in Production and Distributing / H. Dyckhoff, U. Finke. – Heidelberg : Physica Verlag, 1992. – 248 p.
2. Dowsland, K.A. Packing problems / K.A. Dowsland, W.B. Dowsland // European Journal of Operational Research. – 1992. – Vol. 56. – P. 2–14.
3. Dyckhoff, H. Cutting and packing (C&P) / H. Dyckhoff, G. Scheithauer, J. Terno // Annotated Bibliographies in Combinatorial Optimization. – Chichester : Wiley, 1997. – P. 393–413.
4. Стоян, Ю.Г. Методы и алгоритмы размещения плоских геометрических объектов / Ю.Г. Стоян, Н.И. Гиль. – Киев : Наукова думка, 1976. – 247 с.
5. Стоян, Ю.Г. Размещение геометрических объектов / Ю.Г. Стоян. – Киев : Наукова думка, 1975. – 239 с.
6. Стоян, Ю.Г. Периодическое размещение геометрических объектов / Ю.Г. Стоян, А.А. Панасенко. – Киев : Наукова думка, 1978. – 175 с.
7. Мухачева, Э.А. Рациональный раскрой промышленных материалов. Применение АСУ / Э.А. Мухачева. – М. : Машиностроение, 1984. – 176 с.
8. Стоян, Ю.Г. Математические модели и оптимизационные методы геометрического проектирования / Ю.Г. Стоян, С.В. Яковлев. – Киев : Наукова думка, 1986. – 266 с.
9. Gilmore, P.C. Multistage cutting problems of two or more dimensions / P.C. Gilmore, R.E. Gomory // Operations Research. – 1965. – Vol. 13. – P. 94–119.
10. Dyckhoff, H. A typology of cutting and packing problems / H. Dyckhoff // European Journal of Operational Research. – 1990. – Vol. 44. – P. 145–149.
11. Wascher, G. An improved typology of cutting and packing problems / G. Wascher, H. Haussner, H. Schumann // European Journal of Operational Research. – 2007. – Vol. 183. – P. 1109–1130.

12. Lody, A. Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems / A. Lody, S. Martello, D. Vigo // INFORMS Journal on Computing. – 1999. – Vol. 11. – P. 345–357.

13. Lody, A. Recent advances on two-dimensional bin packing problems / A. Lody, S. Martello, D. Vigo // Discrete Applied Mathematics. – 2002. – Vol. 123/124. – P. 373–380.

14. Lody, A. Two-dimensional packing problems: a survey / A. Lody, S. Martello, M. Monaci // European Journal of Operational Research. – 2002. – Vol. 141. – P. 241–252.

15. Tarnowski, A.G. Advanced polynomial time algorithm for guillotine generalized pallet loading problem / A.G. Tarnowski // Decision Making under Conditions of Uncertainty (Cutting-Packing Problems); ed. E.A. Mukhacheva. – Ufa, 1997. – P. 93–124.

16. Tarnowski, A. A polynomial time algorithm for the guillotine pallet loading problem / A. Tarnowski, J. Tern, G. Scheithauer // INFOR. – 1994. – Vol. 32. – P. 275–286.

17. Гэри, М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. – М. : Мир, 1982 .– 416 с.

18. Канторович, Л.В. Рациональный раскрой промышленных материалов / Л.В. Канторович, В.А. Залгаллер. – Новосибирск : Наука, 1971. – 299 с.

19. Автобиография Леонида Витальевича Канторовича // Оптимизация : сб. науч. тр. СО АН СССР. – Новосибирск, 1982. – Вып. 28. – С. 50–57.

20. Gilmore, P.C. A linear programming approach to the cutting stock problem / P.C. Gilmore, R.E. Gomory // Operations Research. – 1961. – Vol. 9. – P. 849–859.

21. Gilmore, P.C. A linear programming approach to the cutting stock problem. Part II / P.C. Gilmore, R.E. Gomory // Operations Research. – 1963. – Vol. 11. – P. 863–888.

22. Beasley, J.E. An exact two-dimensional non-guillotine cutting tree search procedure / J.E. Beasley // Operations Research. – 1985. – Vol. 33. – P. 49–64.

23. Hadjiconstantnou, E. An exact algorithm for the orthogonal, 2-D cutting problems using guillotine cuts / E. Hadjiconstantnou, N. Christofides // European Journal of Operational Research. – 1995. – Vol. 83. – P. 21–38.

24. Lodi, A. Models and bounds for two-dimensional level packing problems / A. Lody, S. Martello, D. Vigo // Journal of Combinatorial Optimization. – 2004. – Vol. 8. – P. 363–379.

25. Романовский, И.В. Решение задачи гильотинного раскроя методом переработки списка состояний / И.В. Романовский // Кибернетика. – 1969. – № 1. – С. 102–103.

26. Fekete, S.P. On more-dimensional packing I: Modelling / S.P. Fekete, J. Schepers. – Köln, 1997. – 15 p. – (Technical paper ZPR97-288 / Matematisches Institut, Universität zu Köln).

27. Fekete, S.P. A combinatorial characterization of higher dimensional orthogonal packing / S.P. Fekete, J. Schepers // Mathematics of Operations Research. – 2004. – Vol. 29. – P. 353–368.

28. Романовский, И.В. Алгоритмы решения экстремальных задач / И.В. Романовский. – М. : Наука, 1977. – 352 с.

29. Martello, S. Knapsack Problems: Algorithms and Computer Implementations / S. Martello, P. Toth. – Chichester : Wiley, 1990. – 296 p.

30. Belov, G. A cutting plane algorithm for the one-dimensional cutting stock problem with multiple stock lengths / G. Belov, G. Scheithauer // European Journal of Operational Research. – 2002. – Vol. 141. – P. 274–294.

31. Липовецкий, А.И. К оптимизации свободного размещения прямоугольников / А.И. Липовецкий // Автоматическое проектирование в машиностроении. – Минск : Ин-т техн. кибернетики АН БССР, 1985. – С. 80–87.

32. Бухвалова, В.В. Задачи прямоугольного раскроя: метод зон и другие алгоритмы / В.В. Бухвалова. – СПб. : СПбГУ, 2001. – 96 с.

33. Картак, В.М. Матричный алгоритм поиска оптимального решения для задачи упаковки прямоугольников в полубесконечную полосу / В.М. Картак // Информационные технологии. – 2008. – № 2. – С. 24–30.

34. Clautiaux, F. A new exact method for the two-dimensional orthogonal packing problem / F. Clautiaux, J. Carlier, A. Moukrim // European Journal of Operational Research. – 2007. – Vol. 183. – P. 1196–1211.

35. Martello, S. Exact solution of the two-dimensional finite bin packing problem / S. Martello, D. Vigo // Management Sciences. – 1998. – Vol. 44. – P. 388–399.

36. Boschetti, M. New upper bounds for the two-dimensional orthogonal non guillotine cutting stock problem / M. Boschetti, E. Hadjiconstantinou, A. Mingozzi // IMA Journal of Management Mathematics. – 2002. – Vol. 13. – P. 95–119.

37. Martello, S. An exact approach to the strip-packing problem / S. Martello, M. Monaci, D. Vigo // INFORMS Journal on Computing. – 2003. – Vol. 15. – P. 310–319.

38. Fekete, S.P. A general framework for bounds for higher- dimensional orthogonal packing problems / S.P. Fekete, J. Schepers // Mathematical methods of Operations Research. – 2004. – Vol. 60. – P. 311–329.

39. Тарновский, А.Г. Полиномиальный алгоритм решения задачи монораскроя / А.Г. Тарновский // Системы программного обеспечения решения задач оптимального планирования : тез. Одиннадцатой Всесоюз. школы (г. Кострома, 20–21 мая 1990 г.). – М. : ЦЭМИ, 1990. – С. 128–129

40. Girlich, E. On polynomial solvability of two multiprocessor scheduling problems / E. Girlich, A.G. Tarnowski // Mathematical Methods of Operations Research. – 1999. – Vol. 50. – P. 27–51.

41. Наумович, Н.А. Экспериментальное исследование серий градиентных методов / Н.А. Наумович // Методы, алгоритмы и программы решения экстремальных задач : сб. науч. тр. – Минск : Ин-т техн. кибернетики АН БССР, 1985. – С. 98–111.

42. Ковалев, М.М. Эффективность градиентных алгоритмов раскроя / М.М. Ковалев, А.Г. Тарновский // Комбинаторно-алгебраические методы в дискретной оптимизации : межвуз. тематич. сб. науч. тр. – Нижний Новгород : НГУ, 1991. – С. 22–36.

43. Ковалев, М.М. Анализ эффективности серий градиентных алгоритмов раскроя / М.М. Ковалев, А.Г. Тарновский // Математическое обеспечение рационального раскроя в системах автоматизированного проектирования : тез. Всесоюзн. конф. – Уфа, 1988. – С. 62–65.

44. Тарновский, А.Г. САПР «РАСКРОЙ» – промышленная система раскроя материалов на персональных ЭВМ / А.Г. Тарновский // Методы решения экстремальных задач и смежные вопросы : сб. науч. тр. – Минск : Ин-т техн. кибернетики АН БССР, 1990. – С. 142–158.

45. Филиппова, А.С. Проблемы декодирования прямоугольных упаковок: краткий обзор современных технологий / А.С. Филиппова // Информационные технологии. – 2005. – № 12. – С. 13–19.

46. Методы локального поиска в задачах ортогонального раскроя и упаковки: аналитический обзор и перспективы развития / Э.А. Мухачева [и др.] // Информационные технологии. – 2004. – № 5. – С. 2–17.

47. Liu, D. An improved BL-algorithm for genetic algorithm of the orthogonal packing of rectangles / D. Liu, H. Teng // European Journal of Operational Research. – 1999. – Vol. 112, № 2. – P. 413–420.

48. Performance bounds for level-oriented two-dimensional packing algorithms / E.G. Coffman Jr. [et al.] // SIAM Journal on Computing. – 1980. – Vol. 9. – P. 801–826.

49. Baker, B.S. Orthogonal packing in two dimensions / B.S. Baker, E.G. Coffman Jr., R.L. Riverst // SIAM Journal on Computing. – 1980. – Vol. 9. – P. 846–855.

50. Мухачева, А.С. Технология блочных структур локального поиска оптимума в задачах прямоугольной упаковки / А.С. Мухачева // Информационные технологии. – 2004. – № 5. Приложение. – С. 18–32.

51. Rectangle-packing-based module placement / H. Murata [et al.] // Proc. IEEE/ACM Int. Conf. on Computer-Aided Design. – San Jose, California, 1995. – P. 472–479.

52. Филиппова, А.С. Задача двумерной упаковки в полубесконечную полосу: численный эксперимент с алгоритмами локального поиска и с декодерами блочной структуры / А.С. Филиппова // Информационные технологии. – 2005. – № 6.– С. 32–45.

53. Мухачева, Э.А. Задача прямоугольной упаковки: методы локального поиска оптимума на базе блочных структур / Э.А. Мухачева, А.С. Мухачева // Автоматика и телемеханика. – 2004. – № 2. – С. 101–112.

54. Goldberg, D. Genetic Algorithms in Search, Optimization and Machine Learning / D. Goldberg. – Boston : Adison-Wesley, 1989. – 372 p.

55. Батищев, Д.И. Генетические алгоритмы решения экстремальных задач / Д.И. Батищев. – Воронеж : ВГТУ, 1995. – 69 с.

56. Aarts, E.H.L. Local search in combinatorial optimization / E.H.L. Aarts, J.K. Lenstra. – Chichester : Wiley, 1997. – 528 p.

57. Норенков, И.П. Эвристики и их комбинации в генетических методах дискретной оптимизации / И.П. Норенков // Информационные технологии. – 1999. – № 1. – С. 2–7.

58. Мухачева, Э.А. Генетический алгоритм блочной структуры в задачах двумерной упаковки / Э.А. Мухачева, А.С. Мухачева, А.В. Чиглинцев // Информационные технологии. – 1999. – № 11. – С. 13–18.

59. Linear one-dimensional cutting-packing problems: numerical experiments with sequencial value correction method (SVC) and a modified branch-and-bound method (MBB) / E.A. Mukhacheva [et al.] // Pescuisa Operational. – 2000. – Vol. 20. – P. 153–168.

60. Мухачева, А.С. Генетический алгоритм поиска минимума в задачах двумерного гильотинного раскроя / А.С. Мухачева, А.В. Чиглинцев // Информационные технологии. – 2001. – № 3. – С. 27–31.

61. Gehring, H. Genetic algorithm for solving the container loading problem / H. Gehring, A. Bortfeldt // International Transactions in Operational Research. – 1997. – Vol. 4. – P. 401–418.

62. Hadjiconstantnou, E. A hybrid genetic algorithm for the two-dimensional single large object placement problem / E. Hadjiconstantnou, M. Iori // European Journal of Operational Research. – 2007. – Vol. 183. – P. 1150–1166.

63. Scholl, A. BISON: A fast hybrid procedure for exactly solving the one-dimensional Bin-Packing Problem / A. Scholl, R. Klein, C. Yuergens // Computers & Operations Research. – 1997. – Vol. 24. – P. 627–645.

64. Задачи двумерной упаковки: развитие генетических алгоритмов на базе смешанных процедур локального поиска оптимального решения / А.С. Мухачева [и др.] // Информационные технологии. – 2001. – № 9. Приложение. – С. 1–24.

65. Glover, F. Tabu search and adaptive memory programming – advances, applications and challenges / F. Glover // Interfaces in Computer Science and Operations Research. – Kluwer Academic Publishers, 1996. – P. 1–75.

66. Hertz, A. Tabu Search / A. Hertz, E. Taillard, D. de Werra // Local Search in Combinatorial Optimization. – Chichester : Wiley, 1997. – P. 121–136.

67. Метод поиска минимума с запретами в задачах двумерного гильотинного раскроя / Э.А. Мухачева [и др.] // Информационные технологии. – 2001. – № 6. – С. 25–31.

68. Alvarez-Valdes, R. A tabu search algorithm for two-dimensional non-guillotine cutting problem / R. Alvarez-Valdes, F. Parreno, J.M. Tamarit // European Journal of Operational Research. – 2007. – Vol. 183. – P. 1270–1277.

69. Dorigo, M. Ant algorithms for discrete optimization / M. Dorigo, G. Di Caro, L. Gambardella // Artificial Life. – 1999. – Vol. 5. – P. 137–172.

70. Валеева, А.Ф. Применение метаэвристики муравьиной колонии к задачам двумерной упаковки / А.Ф. Валеева // Информационные технологии. – 2005. – № 10. – С. 36–43.

71. Schwerin, P. A new lower bound for the bin-packing problem and its integration to MTP / P. Schwerin, G. Wascher // Pescuisa Operational. – 1999. – Vol. 9. – P. 111–131.

72. Schwerin, P. The bin-packing problem: a problem generator and some numerical experiments with FFD packing and MTP / P. Schwerin, G. Wascher // International Transactions in Operational Research. – 1997. – Vol. 4. – P. 337–389.

73. The Management School, Imperial College, UK [Electronic resource]. – Mode of access: http://people.brunel.ac.uk/~mastjjb/jeb/info.html. – Date of access: 16.06.2008.

74. Филиппова, А.С. Задача ортогональной упаковки в полубесконечную полосу: численный эксперимент с безотходными задачами E. Hopper / А.С. Филиппова, Э.А. Мухачева, А.В. Чиглинцев // Информационные технологии. – 2005. – № 7. – С. 45–54.

75. Hopper, E. An empirical investigation of metf-heuristic and heuristic algorithms for a 2D packing problem / E. Hopper, B. Turton // European Journal of Operational Research. – 2001. – Vol. 128. – P. 34–57.

Поступила 28.05.08

Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: andrei.golovistikov@life.com.by

A.V. Golovistikov

THE ORTHOGONAL TWO-DIMENSIONAL CUTTING-PACKING PROBLEMS: A SURVEY

A survey of the results achieved on solving two-dimensional cutting-packing problems of different kinds is provided. The main problem formulations, mathematical models and approaches for solving orthogonal two-dimensional packing problems of the rectangle-shaped objects into bins or strips and the cutting stock problem (including the guillotine cutting) are discussed. The main attention is given to the heuristic algorithms as well as to the meta-heuristic approaches utilizing genetic algorithms, the search with prohibitions and the ant colony algorithms.