Уфа: УГАТУ, 2005 Вестник УГАТУ Т. 6, № 1 (12). С. 106–115

УДК 004.3

А. С. ФИЛИППОВА

БЛОЧНАЯ ТЕХНОЛОГИЯ КОНСТРУИРОВАНИЯ АЛГОРИТМОВ ДЛЯ РЕШЕНИЯ ЗАДАЧ ПРЯМОУГОЛЬНОЙ УПАКОВКИ ПОЛУБЕСКОНЕЧНОЙ ПОЛОСЫ

Рассматривается NP-трудная задача прямоугольной упаковки полубесконечного рулона (1.5 Dimensional Bin Packing, 1.5DBP). Предлагается новая технология блочных структур для решения задач 1.5DBP на основе вероятностных методов локального поиска оптимума. После небольших модификаций она может быть применена и для решения задач 2DBP, а также параллелепипедной упаковки, 3DBP. *Задача прямоугольной упаковки; технология блочных структур; локальный поиск оптимума; метаэвристики*

ВВЕДЕНИЕ

Под задачами раскроя-упаковки (Cutting and Packing, **C&P**) понимается широкий класс задач, допускающих различное прикладное толкование. Общим для этого класса является наличие двух групп объектов. Между элементами этих групп устанавливается и оценивается соответствие. Впервые качественная типология в области раскроя-упаковки проведена в 1990 г. немецким ученым *H. Dykhoff* [22]. Она принята в мировой практике и используется при изучении моделей и методов решения задач раскроя-упаковки. Разнообразие моделей определяется прежде всего фактором геометрии. Различают задачи линейного (одномерного), прямоугольного (двухмерного) и параллелепипедного (трехмерного) раскроя-упаковки. Среди этих задач выделяются гильотинный раскрой и упаковка. Особое значение имеют задачи нестинга, размещения деталей сложной геометрической формы в заданных областях. Для них на первый план выступают информационные проблемы задания фигур, учета и обеспечения их непересечения, кодировки и другие. Задачи **C&P** являются типичными представителями NP-трудных проблем комбинаторики, и для их решения (включая проблемы нестинга) применяются общие подходы: точные методы, простые эвристики и метаэвристики. Ввиду неполиномиальной сложности точных алгоритмов, авторами многих работ уделяется значительное внимание приближенным методам и эвристикам.

В течение 90-х годов прошлого столетия по теме раскроя-упаковки было выпущено несколько специальных изданий: под редакцией *H. Dyckhoff & G. Wascher* в 1990 г. [23], *S. Martello* в 1994 г. [31, 32], *Y. Lirov* в 1995 г. [29], *E. Bischoff & G. Wascher* в 1995 г. [21], *E. Mukhacheva* в 1997 г. [33], *H. Yanasse* в 1999 г. [36], *P. Wang & G. Waesher* в 2002 г. [35]. Более того, сотни статей опубликованы в международных и российских журналах: European Journal of Operational Research (EJOR), Computers & Operational Research, Computers & Industrial Engineering, Operations Research Letters, Pesquisa Operacional; «Информационные технологии», «Автоматика и телемеханика», «Дискретный анализ и исследование операций», «Вестник высшей школы», «Кузнечно-штамповочное производство» и другие центральные и ведомственные издания. При этом статьи и книги имеют как теоретический, так и прикладной характер.

Причина растущего интереса к задачам раскроя-упаковки состоит в их принадлежности к **NP**-трудным проблемам, в широкой применимости результатов, разнообразии и сложности задач реального мира, которые можно реализовать как **NP**-трудные.

Представленная здесь технология предназначена для решения задач упаковки в полубесконечную полосу (**1.5DBP**).

Термин **1.5**-размерной задачи ввел в 1980 г. *A. Hinxman* [26]. Остановимся подробнее на постановке этой задачи. Пусть имеются прямоугольная полоса заданной ширины W и неограниченной длины и набор из

Работа поддержана Фондом Президента Российской Федерации, проект № МК 145.2003.01

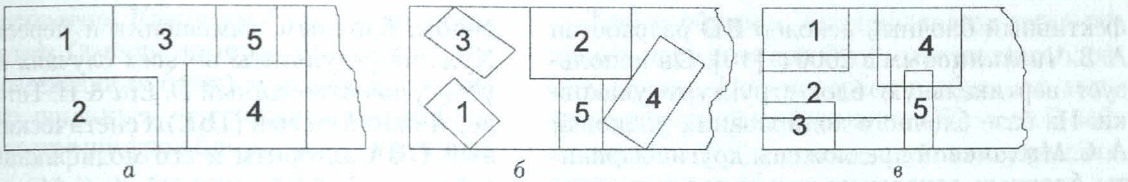

Рис. 1. Упаковка прямоугольников: *а* — допустимая ортогональная; *б* — неортогональная; *в* — недопустимая ортогональная

m прямоугольных предметов заданных размеров (w_i, l_i), $i = 1, \ldots, m$, где w_i — ширина, l_i — длина стороны, параллельной неограниченной грани полосы. Введем прямоугольную систему координат: оси Ox и Oy совпадают соответственно с нижней неограниченной и боковой сторонами полосы. Положение каждого прямоугольника P_i зададим координатами (x_i, y_i) его левого нижнего угла.

Задача 1.5DBP. Набор векторов (x_i, y_i), $i = 1, \ldots, m$ называется прямоугольной упаковкой (Rectangular Packing, **RP**), если для $i \neq j$; $i, j = 1, \ldots, m$

$$x_i \geqslant (x_j + l_j) \vee x_j \geqslant (x_i + l_i) \qquad (1)$$

или

$$y_i \geqslant (y_j + w_j) \vee y_j \geqslant (y_i + w_i); \qquad (2)$$

$$x_i \geqslant 0 \wedge y_i \geqslant 0 \wedge y_i + w_i \leqslant W. \qquad (3)$$

Условия (1) или (2) означают непересечение прямоугольников между собой, а (3) — с гранями полосы. На рис. 1 изображены допустимая и недопустимые упаковки. Если длина L занятой части полосы достигает минимума, то **RP** — оптимальная упаковка. Исходную информацию для **1.5DBP** принято задавать вектором $(W; m; w; l)$; $w = (w_1, \ldots, w_m)$; $l = (l_1, \ldots, l_m)$.

Различаются задачи прямоугольной упаковки и гильотинного раскроя. В том и другом случаях требуется разделить большой прямоугольник на малые так, чтобы стороны прямоугольников были параллельны сторонам большого прямоугольника и выбранная функция цели достигала минимума. Задачи упаковки и гильотинного раскроя различаются технологией разделения. Гильотинность предполагает возможность только сквозных резов, параллельных кромкам раскраиваемого материала. Задачи гильотинного раскроя берут свое начало от одномерного случая и рассматриваются как обобщение последней в первых работах *Л. В. Канторовича*, *В. А. Залгаллера* [2,3] и *P. Gilmore*, *R. Gomory* [25]. Методы, алгоритмы и технологии, разработанные для задач линейного раскроя, адаптированы на случай гильотинного (2- и 3-мерного) раскроя. Подробно эта задача в условиях массового и серийного производства описана в работах *Э. А. Мухачевой* [12,13] в России и *J. Terno*, *R. Lindeman & G. Scheithauer* [34] в Германии. Метод склейки *И. В. Романовского*, разработанный вначале для задач линейного раскроя, также был обобщен на случай гильотинного раскроя [18]. В современных работах авторы часто исходят из тщательной проработки задачи линейного раскроя. Например, метод отсекающих плоскостей, подробно разработанный и исследованный *Г. Беловым* и *G. Scheithauer*, обобщается *Г. Беловым* на случай гильотинного раскроя [20].

Среди простых алгоритмов решения задач **1.5DBP** и **2DBP** выделяются те, которые служат декодерами в многопроходных алгоритмах. Их разработка стала актуальной в связи с появлением и развитием вероятностных методов локального поиска оптимума. Они вычисляют значение целевой функции и восстанавливают эскиз упаковки. Для этого с помощью декодера достаточно найти прямую схему кодирования, которой является последовательность координат (x, y), удовлетворяющая условиям допустимости упаковки. В качестве исходных кодов часто используют приоритетные списки (перестановки прямоугольников) [10]. Затем с помощью декодера переходят к прямой схеме. Известны различные алгоритмы — декодеры, и между ними возникла некая конкуренция. Широкое распространение на Западе получил декодер нижний-левый (Bottom-Left, **BL**). Усовершенствованный нижний-левый (Improved Bottom-Left, **IBL**) предложен *D. Liu & H. Teng* в 1999 г. [30]. Различные варианты реализации этого алгоритма использует *И. П. Норенков* в рамках генетического алгоритма [17]. *E. Hopper & B. Turton* применяют **BL** в составе нескольких метаэвристик [27]. *I. Imahori* (Япония) в 1995 г. разработал схему парных последовательностей (Sequence Pair, **SP**) кодирования упаковок для комбинаторного поиска [28]. Эффективный блочный декодер **BD** разработан *А. В. Чиглинцевым* в 2000 г. [10]. Он использует вертикальную блок-структуру упаковки. На базе блочного кодирования упаковки *А. С. Мухачевой* предложены другие варианты блочных декодеров: поиск парных списков (Dublicity List Search, **DLS**) [6], замещения (**Sub**) и декодера перестройки (Reconstructions, **Rec**) [14].

Внесение в детерминированный алгоритм элементов случайности повышает его результативность. Так, например, повысилась эффективность алгоритмов *последовательного уточнения оценок* (Sequential Value Correction, **SVC**) и *динамического перебора* (Dinamic Sorting, **DS**) после внесения в них элементов стохастики [15, 11]. Бурное развитие вероятностных методов локального поиска оптимума началось 20 лет назад с появлением метаэвристик для решения **NP**-трудных задач, систематическое обоснование и характеристика которых приведены в 1996 г. в книге под редакцией *E. Aurts, J. Lenstra* [19]. В России обзор вероятностных методов локального поиска оптимума для **NP**-трудных задач сделан в 2001 г. *Ю. А. Кочетовым* [5]. В обзоре обсуждаются общие схемы алгоритмов *поиска с запретами, имитации отжига* и *генетических* алгоритмов. Показано, что эти разные по своей структуре алгоритмы используют общую математическую конструкцию конечных цепей Маркова. Это свойство гарантирует сходимость по вероятности наилучшего найденного решения к оптимальному решению задачи.

Первыми среди метаэвристик для задач раскроя-упаковки стали применяться генетические алгоритмы. В этой связи хорошо зарекомендовали себя работы 1992–2000 годов, например, *E. Folkenaur* [24]. Он ввел процедуру группирования в классическую схему. Основные генетические структуры (*ген, аллели, хромосома*) и операции (*кроссовер* и *мутация*) определены многими авторами, в том числе в России *Д. И. Батищевым* [1]. Возможны различные способы кодирования и приемы идентификации простейших структур. Это порождает различные классы генетических алгоритмов. Классический генетический алгоритм для решения задач **1.5DBPP** и **2DBPP** представлен, например, в работах *D. Liu & H. Teng* [30]; *А. С. Мухачевой, М. А. Смагина* [10]. Они различаются деталями и программной реализацией. Эффективность классического алгоритма зависит от используемого декодера. Исследован алгоритм *D. Liu & H. Teng* с родным декодером, блочным, замещения и перестройки. Худшие результаты во всех случаях показал усовершенствованный *D. Liu & H. Teng* декодер нижний-левый (**IBL**). Генетический блочный **GBA** алгоритм и его модификации разработаны *Э. А. Мухачевой, А. С. Мухачевой, А. В. Чиглинцевым* в 1999 г. [16]. В этом алгоритме генами являются блоки — прямоугольные фрагменты упаковки.

И. П. Норенков в 1999 г. предлагает использовать в качестве генов простые эвристики [17]. Результаты тестирования оказались сопоставимы с блочным алгоритмом при количестве прямоугольников $m \leqslant 250$ и превосходят все другие тестируемые алгоритмы при $250 < m < 1000$. Эта технология использовалась *А. С. Мухачевой* и *А. В. Чиглинцевым* в 2000 г. для задач гильотинного раскроя [9].

Среди эволюционных алгоритмов особое место занимает специальная блочная технология конструирования алгоритмов локального поиска оптимума в задачах **1.5DBP** и **2DBP**. Она развита в работах *А. С. Мухачевой* и другими авторами [6–8, 14, 16]. В работе [16] положено начало в области блочных технологий. Там используется только одна, вертикальная блок-структура. С 2002 г. для конструирования алгоритмов *А. С. Мухачевой* и *В. М. Картаком* предлагается схема парных списков, которая базируется на двух структурах: вертикальной и горизонтальной. *А. С. Мухачева* использует язык блок-структур для конструирования алгоритмов локального оптимума [8], а *В. М. Картак* — связных матриц для алгоритма типа ветвей и границ [4].

Здесь делается попытка обобщения накопленного опыта в виде блочной технологии конструирования алгоритмов решения задач двухмерной упаковки.

1. БЛОК-СТРУКТУРЫ УПАКОВКИ И ИХ СВОЙСТВА

Пусть имеется прямоугольная упаковка **RP**. Проведем через правые стороны прямоугольников вертикальные резы, они разбивают **RP** на прямоугольные *вертикальные блоки* одной и той же ширины W и различной длины. Пусть длина **RP** равна N. Проведем через верхние стороны прямоугольников горизонтальные линии. Тогда **RP** разобьется на *горизонтальные блоки* одной и той же длины N и различной ширины η_j, причем $\sum_j \eta_j \leqslant W$. Таким образом, мы получаем две блок-структуры для **RP**, вертикальную и горизонтальную. Каждому блоку j сопоставим кортеж (запись номеров прямоугольников, пересекающих блок) и длину χ_j вертикального, ширину η_j — горизонтального блока. В качестве шифров блок-структур используем пары списков $(S, N)\left((\tilde{S}, \tilde{N})\right)$, где

$$S = \{1(j), 2(j), ..., i(j), ...\}\, \chi_j,$$
$$j = \overline{1, r}, \quad N = \sum_{j=1}^{r} \chi_j;$$
$$\tilde{S} = \{1(j), 2(j), ..., i(j), ...\}\, \eta_j, \tag{4}$$
$$j = \overline{1, q}, \quad \tilde{N} = \sum_{j=1}^{q} \eta_j,$$

$i(j)$ — номер прямоугольника в позиции i, пересекающего блок j; r — количество вертикальных, q — горизонтальных блоков.

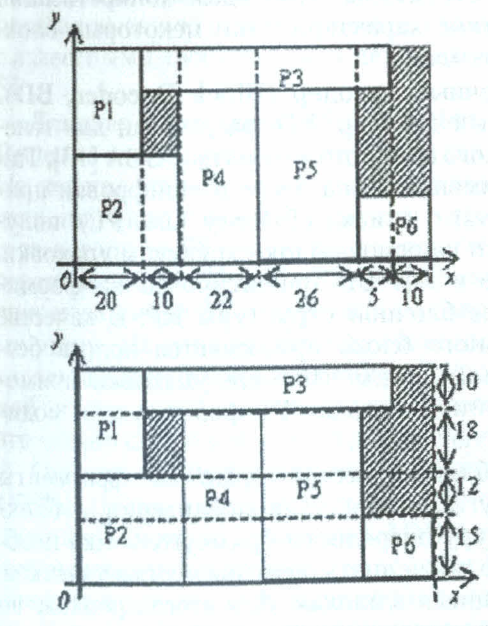

Рис. 2. Блок-структуры упаковки: *а* — вертикальная; *б* — горизонтальная

Пример 1. На рис. 2 изображены вертикальная и горизонтальная блок-структуры упаковки с $m = 6$. Разбиения на блоки указаны штриховыми линиями. Там же указаны длины (ширины) блоков. Легко проверить, что этим блок-структурам отвечают следующие списки:

$S = \{(2,1)20; (2,3)10; (4,3)22; (5,3)26; (6,3)5; (6)10\}$;
$\tilde{S} = \{(2,4,5,6)15; (2,4,5)12; (1,4,5)18; (1,3)10\}$.

Длина $N(\mathbf{RP}) = 20 + 10 + 22 + 26 + 5 + 10 = 93$; ширина $\tilde{N}(\mathbf{RP}) = 15 + +12 + 18 + 10 = 55 = W$.

Таким образом, имея упаковку, легко получить пару списков S и $\tilde{S}$, отвечающих блок-структурам. Нас интересует обратная задача: по исходной информации $(W; m; w; l)$ найти списки $S(\tilde{S})$, соответствующие упаковке. Обозначим: $I_j(\tilde{I}_j)$ — множество прямоугольников, пересекающих j-й вертикальный (горизонтальный) блок; $I_j^+(\tilde{I}_j^+)$ — множество прямоугольников $i \in I_j (i \in \tilde{I}_j)$, заканчивающихся в j-м блоке; $I_j^-(\tilde{I}_j^-)$ — множество прямоугольников $i \in I_j (i \in \tilde{I}_j)$, начинающихся в j-м блоке. Важными являются следующие необходимые свойства соответствия:

Лемма 1. Пусть список $S(\tilde{S})$ построен на основании исходной информации $(W; m; w; l)$ задачи **1.5DBP**. Если этот список соответствует допустимой упаковке **RP**, то он удовлетворяет следующим свойствам:

1°. *Разнородность* прямоугольников. Элементы $i(j)$ каждого кортежа $I_j(\tilde{I}_j)$ различные.

2°. *Продолженность* прямоугольников. Если некоторый элемент $i(j) \notin I_j^+(\tilde{I}_j^+)$, то $i(j) \in I_{j+1}(\tilde{I}_{j+1})$.

3°. *Непересечение*. Для любого вертикального и любого горизонтального блока выполняется одно из следующих условий:

(a) для любой пары $(i_1, i_2) \in I_k$, $k \in S$ и любого кортежа $j \in \tilde{S}$: если $i_1 \in \tilde{I}_j$, то $i_2 \notin \tilde{I}_j$ или если $i_2 \in \tilde{I}_j$, то $i_1 \notin \tilde{I}_j$;

(b) для любой пары $(i_1, i_2) \in \tilde{I}_j$, $j \in \tilde{S}$ и любого кортежа $k \in S$: если $i_1 \in I_k$, то $i_2 \notin I_k$ или если $i_2 \in I_k$, то $i_1 \notin I_k$.

4°. *Размещаемость*. Если некоторые элементы $i \in I_j^+(\tilde{I}_j^+)$, $j \in S(\tilde{S})$, то вместо них в $(j+1)$-м кортеже должны разместиться все его новые элементы.

5°. *Непересечение с границами полосы:*

$$\sum_{j=1}^{r} \chi_j = N; \quad \sum_{j=1}^{q} \eta_j = \tilde{N} \leqslant W.$$

6°. *Координаты* $(x_i; y_i)$, $i = \overline{1, m}$ вычисляются по формулам:

$$x_{i(k)} = \sum_{j=1}^{k-1} \chi_j, \; i(k) \in I_k^-;$$
$$y_{i(k)} = \sum_{j=1}^{k-1} \eta_j, \; i(\tilde{k}) \in \tilde{I}_{\tilde{k}}^-. \tag{5}$$

Лемма 2. Свойство 3° является достаточным условием непересечения прямоугольников в упаковке.

Пусть имеет место условие (a) свойства 3°. Тогда если прямоугольники P_{i_1} и P_{i_2} пересекают один и тот же вертикальный блок, то они не могут пересечь один и тот же горизонтальный блок, т. е. содержатся в различных горизонтальных блоках. А это означает раздвинутость прямоугольников по оси Oy, т. е. (2). Аналогично выполнение условия (b) означает раздвинутость прямоугольников по оси Ox. Выполнение (a) или (b) означает непересечение прямоугольников по определению.

Лемма 3. Свойства 3° и 4° эквивалентны. При выполнении любого из них прямоугольники в упаковке **RP** не пересекаются.

Утверждение 1. *(Достаточные условия соответствия)*. Блок-структуры (S, N) и $(\tilde{S}, \tilde{N})$, удовлетворяющие свойствам 1°, 2°, 5° и свойствам 3° или 4°, соответствуют допустимой упаковке **RP** с координатами $(x_i; y_i)$, $i = \overline{1, m}$, вычисленными по формулам (5).

Доказательство. Пусть блок-структуры (S, N) и $(\tilde{S}, \tilde{N})$ удовлетворяют свойствам 1°, 2°, 5° и условию (3° или 4°). Условие 1° обеспечивает неразрывность прямоугольников по вертикали, условие 2° — продолженность (неразрывность) прямоугольников по горизонтали. Таким образом, 1°, 2° гарантируют целостность прямоугольников в блок-структурах. Выполнение 3° или 4° означает непересечение прямоугольников между собой ((1) или (2)), а выполнение 5° — непересечение с гранями полосы (3). Тогда блок-структурам соответствует допустимая упаковка по определению. Очевидно, что координаты $(x_i; y_i)$ прямоугольников P_i, $i = \overline{1, m}$, удовлетворяют условиям (5).

2. ТЕХНОЛОГИЯ БЛОЧНЫХ СТРУКТУР: ДЕТЕРМИНИРОВАННЫЕ БЛОК-АЛГОРИТМЫ

На базе блочных структур рассматриваются две технологии: *вертикальной блок-структуры и двойственной блок-структуры*. В первой технологии описывается только одна, вертикальная блок-структура.

Технология базируется на *обобщенной структуре методов локального поиска*, получивших широкое распространение в последние десятилетия в виде метаэвристик. Локальный поиск составляют следующие основные шаги:

1. *инициализация*. Выбрать начальное допустимое решение и вычислить для него значение критерия оптимальности;

2. *генерация соседнего решения*. Выбрать допустимое решение из окрестности начального решения и вычислить для него значение критерия;

3. *анализ перехода*. Проверить, следует ли совершать переход к новому решению. Если да, то применить новое решение в качестве текущего, в противном случае предыдущее решение остается текущим;

4. *конец*. Завершить работу алгоритма и вывести решение.

Блок-структуры используются на втором и четвертом шагах. Могут быть использованы и при построении первого решения. При этом применяются только *вертикальные* блок-структуры (Upright, UBS) или вертикальные и *горизонтальные* (Horizontal, HBS) структуры. Приведем конкретизации и краткие характеристики некоторых блок-алгоритмов.

Блочный декодер (Block Decoder, BD). Блочный декодер, BD, разработан для генетического блочного алгоритма, GBA [16]. Там он применяется на этапе расшифровки приоритетного списка (Priority List, PL), полученного на основе блок-структуры упаковки. Вместе с тем BD ориентирован на формирование блочной структуры RP. В качестве начального блока принимается полоса бесконечной длины. По мере упаковки прямоугольников согласно PL формируют и модифицируют блоки.

В блок-структуре хранятся фрагменты прямоугольников. Для добавления в блок-структуру очередного прямоугольника необходимо также знать размеры пустых участков, остающихся в блоках. Для этого предложено хранить ширину такого участка в самой блок-структуре, но со знаком «*», чтобы отличать от номеров прямоугольников, а его длина равна длине блока, в котором он находится.

Под модификацией блок-структуры понимается корректное (в соответствии с размерами размещаемых прямоугольников) изменение длины блоков, ширины пустых участков, а также добавление новых блоков.

Эксперименты показали, что **GBA** дает лучшие результаты, если в декодере на этапе размещения добавить простые эвристики, смещающие прямоугольник в пределах пустого участка. Эвристики применяются для смещения или нет в зависимости от случайной величины, разыгрываемой для каждого прямоугольника.

Были предложены следующие две эвристики:

• *сравнение*: перед размещением прямоугольника сравниваются возможные потери площади выше и ниже прямоугольника, сдвиг делается в сторону меньшей площади; таким образом, увеличивается вероятность того, что в большей области смогут разместиться еще несколько предметов;

• *сдвиг*: в зависимости от положения прямоугольника относительно средней линии полосы, он сдвигается к ближайшему краю; таким образом, прямоугольники притягиваются к краям полосы, образуя в середине пустое пространство, в котором могут разместиться другие прямоугольники из приоритетного списка.

Трудоемкость **BD** — $O(m^2)$. Блочный декодер разработан А. В. *Чиглинцевым* [10], он показал высокую эффективность, сравнимую с известными «восточными» декодерами типа **SP**.

Декодер замещение (Substitution, Sub). Решение задачи осуществляется в два этапа. На первом этапе решается вертикальная задача линейного раскроя. Для этого применяется алгоритм **FF** с учетом дополнительных ограничений 1°, 2° и 4°. Затем формируется список запретов **TL** и переходят к следующему этапу. На втором этапе необходимо найти любое допустимое решение горизонтальной задачи линейного раскроя. Покажем, что это можно сделать за один проход.

Однопроходный алгоритм решения RCS* (FF*). Пусть имеются: исходная информация вида $(W; m; w; l)$ задачи **1.5DBP** и решение вертикальной задачи в виде пары $(S; N)$. Причем это решение удовлетворяет условиям 1°, 2° и 4°. Существуют эвристики (декодеры), генерирующие **RP** по этой информации. Здесь же мы опишем декодер на базе двойственной схемы, который вычисляет горизонтальную блок-структуру длины $N* \leqslant W$ за один проход. После этого остается по формулам (5) вычислить координаты прямоугольников и изобразить эскиз упаковки.

Горизонтальный алгоритм *первый подходящий*, **FF***, основан на связи двух структур, генерируемой $(\tilde{S}, \tilde{N})$ с уже имеющейся $(S; N)$. Алгоритм **FF*** включает выполнение следующих процедур:

1. *Построенние первого кортежа*.

1.1. *Выбор первого элемента*. В качестве первого элемента i_1 первого кортежа $\tilde{S}$ полагаем первый элемент первого кортежа списка S и удаляем его из всех кортежей списка. Элемент i_1 продолжается в следующих кортежах списка $\tilde{S}$, пока $b_{i_1} \neq 0$.

1.2. *Выбор следующего элемента*. Каждый следующий элемент текущего кортежа выбирается как первый возможный из списка S с учетом его длины, свободного места в кортеже и списка запретов **TL**. Как только элемент выбран, он удаляется (отмечается) в списке S.

2. *Переход к следующему кортежу*. Если $S = \varnothing$, то пара $(\tilde{S}, \tilde{N})$ построена и для нее выполняется условие $\tilde{N} \leq W$, так как алгоритм не нарушает свойств 4^0 вертикальной блок-структуры. Иначе — переход на 3.

3. *Построение следующего кортежа*. Если $b_{i_1} \neq 0$, то в качестве первого остается первый элемент предыдущего кортежа. Если $b_{i_1} = 0$, то в качестве первого выбирается первый элемент из первого непустого кортежа списка S. Далее элементы выбираются по правилу 1.2.

4. *Вычисление координат* прямоугольников по формулам (2.5).

5. *Построение эскиза упаковки*.

Замечание. Можно реализовать работу **Sub** на одном вертикальном этапе, так как при решении линейной задачи размещение элементов жестко фиксируется. В таком случае в качестве фиктивных прямоугольников фигурируют в списке S остатки с признаком (*). Номера им присваиваются начиная с $(m+1)$ по мере появления в блоках. Обозначим через $i(\nu(k))$ номер прямоугольника, занимающего в блоке k-ю позицию ν. Тогда в примере 1 $S = \{(2;1)20; (2;7*;3)10; (4;3)22; (5;3)26; (6;8*;3)5(6;9*)10\}$, а $2(5) = \underline{8*}$; $3(2) = 3$. Тогда координаты $(x_i; y_i)$, $i = \overline{1,m}$ прямоугольников вычисляются по формулам:

$$\begin{gathered} x_{i(\nu(k))} = \sum_{j=1}^{k-1} \chi_j; \\ y_{i(\nu(k))} = \sum_{j=1}^{\nu-1} w_{i(j(k))}, \\ i(\nu(k)) \in I_k^-. \end{gathered} \tag{6}$$

Метод перестройки (Reconstructions, Rec). Топологические свойства упаковок. Предположим, что задача **RCS** решена и известно ее оптимальное решение N. Это решение представлено списком кортежей вида (4). Каждому кортежу отвечает блок, удовлетворяющий свойствам 1° и 2°. Определяющим для **RP** с $L = N$ является выполнение свойства 4° размещаемости прямоугольь-

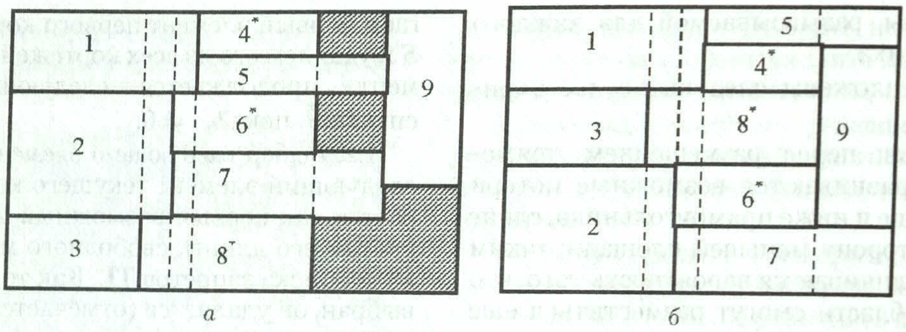

Рис. 3. Прямоугольная упаковка до (*а*) и после (*б*) «перестройки»

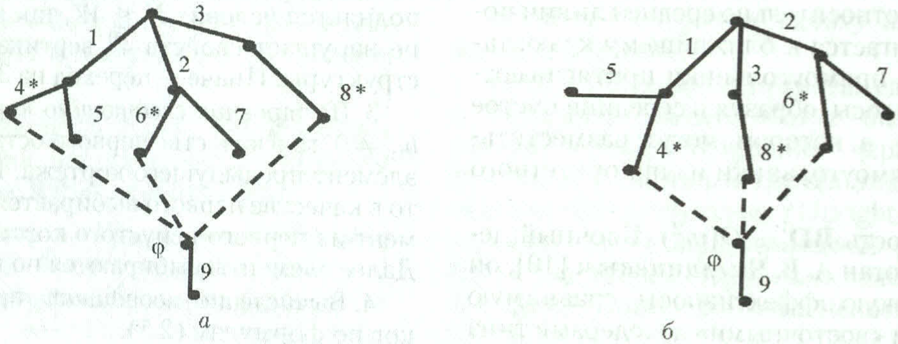

Рис. 4. Интерпретация d.ROLC на графах до (*а*) и после (*б*) «перестройки»

ников. Предположим, что для некоторой пары соседних кортежей $(j, j+1)$ нарушено это свойство. Тогда будем говорить, что при переходе с кортежа j к кортежу $(j+1)$ возникла ситуация *перестройки*, а соответствующие области $(j+1)$-го кортежа назовем *критическими*.

Рассмотрим интерпретацию указанной ситуации на графах. Рассмотрим вначале наглядный пример. На рис. 3 изображены упаковки **RP1** и **RP2** в полосу одних и тех же прямоугольников и $L(\mathbf{RP1}) > L(\mathbf{RP2})$. Четыре первых блока в этих упаковках различаются только перестановкой прямоугольников, входящих в них. В случае *а* в блоке № 5 не удается разместить девятый прямоугольник. Здесь мы встретились с ситуацией *перестройки*. Продолжения областей 4-го, 8-го и 6-го прямоугольников на 5-й блок *критические*. Если удастся их объединить в одну общую область, то перестройка преодолена. В случае б, полученном после перестановки прямоугольников в блоках №№ 1–4 случая *а*, освободилось место для размещения девятого прямоугольника в блоке № 5 (выполнено свойство размещаемости). Процедуру, реализующую нужную перестановку прямоугольников в блоках (элементов в кортежах), назовем ***перестройкой***.

Непрерывный отрезок в **RP**, параллельный боковой стороне полосы, являющийся объединением левых вертикальных сторон упакованных в полосу прямоугольников, назовем *опорной гранью* этих прямоугольников.

Сопоставим первым четырем блокам упаковок, изображенных на рис. 3, неориентированные графы. Вершины графа отвечают опорным граням начальных элементов, а ребра графа – прямоугольникам. На рис. 4, *а* и б сплошными линиями изображены фрагменты графов, соответствующих случаям *а* и б рис. 3.

Граф, отвечающий прямоугольной упаковке, назовем ***RP-графом***.

Известно, что граф, для которого существует ему изоморфный с плоским изображением, называется *планарным*. Плоское изображение графа разбивает плоскость на связные компоненты, одна из которых является неограниченной.

Очевидными являются следующие утверждения:

Утверждение 2.2. **RP**-граф является планарным, и в его плоском изображении корневая и висячие вершины расположены в неограниченной области.

Утверждение 2.3. Для объединения освободившихся областей в блоке j необходимо и достаточно, чтобы граф, отвечающий первым

j блокам, после размещения рядом отмеченных вершин оставался **RP**-графом.

Вершины, инцидентные ребрам 5 и 7, являются висячими для графов, изображенных на рис. 4. В случае a они расположены во внутренней грани графа, а в случае б — во внешней. Таким образом, задача объединения свободных областей сводится к построению плоского изображения **RP**-графа, у которого висячие вершины расположены во внешней неограниченной грани.

Декодер парных списков (Doublicity List Search, DLS). Он использует обе блок-структуры, **UBS** и **HBS**. На первом этапе строится допустимое решение задачи линейного раскроя, удовлетворяющее условиям 1° и 2°. Пусть найденное решение равно N и оно превосходит рекорд. На втором этапе строится допустимое решение горизонтальной задачи линейного раскроя прутков длины N, удовлетворяющее условиям 1°, 2° и 3°. Если полученное решение $\tilde{N} \leq W$, то для построения упаковки остается подсчитать по формулам (5) координаты (x_i, y_i) прямоугольников. В противном случае — алгоритм допустимое решение не нашел.

Список блочных декодеров можно и продолжить. Применялись различные модификации **DLS** [7].

3. ТЕХНОЛОГИЯ БЛОЧНЫХ СТРУКТУР: ВЕРОЯТНОСТНЫЕ БЛОК-АЛГОРИТМЫ ЛОКАЛЬНОГО ПОИСКА

Двойственный метод локального поиска оптимальной упаковки. Рандомизированная двойственная схема генерирует новый список π путем перестановки всех элементов. Это так называемый *наивный эволюционный алгоритм*. Используя полученный список π, решаем задачу прямоугольно ориентированного линейного раскроя алгоритмом первый подходящий **(FF)** с учетом 1° и 2° и получаем пару (S, N). **RCSP*** решается с помощью **FF** с учетом 1°, 2° и 3°. После получения каждого варианта решения $(\tilde{S}; \tilde{N})$ возможны следующие случаи:

1. Выполнено условие $\tilde{N} \leqslant W$, переходим к расчету (x_i, y_i) по формуле (5).

2. Оказалось, что $\tilde{N} > W$, переходим к другому списку π в окрестности, отбраковывая ранее полученную вертикальную блок-структуру.

Процедура *анализ перехода* оставляет лучшую из найденных упаковок и оценивает в процентах отклонение ее длины от лучшего решения линейной задачи. Для найденного лучшего решения по формулам (5) подсчитывают координаты прямоугольников и изображают эскиз упаковки.

Генетический блочный алгоритм (Genetic Block Algorithm, GBA). Пусть задана исходная информация (W, m, w, l) для решения **1.5DBP.** Затем формируется исходная информация (Z, m, λ, b) для задачи **1DCSP**. Решая эту задачу с учетом свойств 1° и 2°, строим пару (S, N). Полученная блок-структура, вообще говоря, не является упаковкой. Ее длина N может использоваться в качестве квазиграницы **RP** в рассматриваемой окрестности.

Генетический алгоритм решения задачи **1.5DBP** интерпретируется как эволюционный процесс, связанный с перестановкой элементов в кортежах. Каждой допустимой упаковке **RP** отвечает ее блочное представление (4), кортежи в нем расположены в установленном порядке и связаны друг с другом (свойство 2°). Это позволяет интерпретировать их как гены, а блок-структуру можно интерпретировать *хромосомой*, содержащей сцепленные между собой гены. Местоположение гена в хромосоме является *локусом*, а альтернативные формы одного и того же гена, расположенные в одинаковых локусах хромосомы, интерпретируются *аллелями*. Хромосома, содержащая в своих локусах конкретные значения аллелей, представляет *генотип*. Конечное множество всех допустимых генотипов образует *генофонд*. Фиксированная блок-структура со значением N функции цели определяет окрестность, в которой реализуется локальный поиск оптимума. *Степенью* $\mu(rp)$ *приспособленности* особи rp является значение L длины занятой части полосы прямоугольной упаковки **RP**. *Оценочной функцией* в окрестности с фиксированным значением N является величина $\Delta = (L - N)/L$. Множество эквивалентных структур образует *ареал*, заданный окрестностью $\{N\}$, а совокупность особей (допустимых упаковок), принадлежащих ареалу, образует *популяцию* P. Численность генофонда популяции в множестве эквивалентных блок-структур определяется параметром $k = \sum\limits_{\nu=1}^{n} r_\nu!$, где r_ν — количество начатых в блоке ν элементов, n - количество блоков. В общем случае экстремальной задачи **1.5DBP** популяция соответствует совокупности допустимых решений. С помощью основных генетических процедур *кроссовер* и *селекция* может быть найдена особь с показателем $L = N$. На этом заканчивается работа алгоритма. Иначе, выполняя заданное количество генетических итераций, находят **RP** с $L < N$ и вычисляют значения оценочной функции. Далее ареал расширяется за счет применения мутации, перехода к новой окрестности. Тогда численность генофонда $k = n! \sum_{\nu=1}^{n} r_\nu!$.

Перечислим основные процедуры **GBA**.

Хромосома — вычисление блок-структуры с помощью модификаций алгоритма **SVC** [22].

Декодер — построение допустимой прямоугольной упаковки **RP**, пользуясь алгоритмом *перестройки* в сочетании с блочным декодером [24].

Генофонд — построение аллелей путем перестановок элементов в кортежах.

Кроссовер — выбор случайной хромосомы (родителя), выбор гена (блока) хромосомы, перестановка двух случайных элементов в блоках.

Мутация — построение новой начальной хромосомы.

Алгоритм **GBA** состоит из выполнения следующих шагов:

G1. Построение блок-структуры начальной хромосомы.

G2. Построение начальной популяции. Выполняются процедуры *генофонд* и *декодер*. Повторяется до получения заданного количества особей в популяции.

G3. *Кроссовер* и занесение в популяцию наиболее приспособленных особей.

G4. *Мутация* и переход на G1.

Выполняется алгоритм до тех пор, пока не достигнута *квазиграница*, полученная **SVC**, или не выполнено заданное количество шагов.

ЗАКЛЮЧЕНИЕ

Статья посвящена блочной технологии конструирования алгоритмов прямоугольной упаковки: введены понятия блок-структур и блочные способы кодирования упаковок; сформулированы и обоснованы основные свойства блок-структур, в том числе необходимые и достаточные условия эквивалентности блок-структур и прямоугольной упаковки.

На этой основе разработана аппроксимация прямоугольной упаковки линейным раскроем специального вида и предложены две группы алгоритмов. Первая использует вертикальную и горизонтальную структуры. Вторая — вертикальную блок-структуру. Основным достоинством является открытая методология разработки блочных алгоритмов. На этой базе могут создаваться более эффективные эвристики, приближенные и точные алгоритмы. Более того, технология может применяться и для разработки методов расчета параллелепипедной упаковки.

СПИСОК ЛИТЕРАТУРЫ

1. **Батищев Д. Ю.** Генетические алгоритмы решения экстремальных задач: Учебное пособие / Под ред. Я. Е. Львовича. Воронеж: Воронежск. гос. техн. ун-т; Нижегородск. ун-т, 1995. 96 с.
2. **Канторович Л. В., Залгаллер В. А.** Рациональный раскрой промышленных материалов. Новосибирск: Наука, СО, 1971. 299 с.
3. **Канторович Л. В., Залгаллер В. А.** Расчет рационального раскроя материалов. Л.: Лениздат, 1951.
4. **Картак В. М.** Оптимальная упаковка N-мерных параллелепипедов в полубесконечность // Методы оптимизации и их приложения: 12-я Байкальск. междунар. конф. Иркутск, 2001. С. 18–22.
5. **Кочетов Ю. А.** Вероятностные методы локального поиска для задач дискретной оптимизации // Дискретная математика и ее приложения: Сб. лекц. молодежн. и науч. шк. по дискретной математике и ее приложениям. М.: Изд-во ЦПИ при мех.-мат. фак-те МГУ, 2000. С. 87–117.
6. **Мухачева А. С., Ширгазин Р. Р.** Задачи упаковки прямоугольников: рандомизированная эвристика на базе двойственной схемы локального поиска оптимума // Информационные технологии. 2003. № 5. С. 18–23.
7. **Мухачева Э. А., Мухачева А. С.** Задача прямоугольной упаковки: методы локального поиска оптимума на базе блочных структур // Автоматика и телемеханика. 2004. № 2. С. 101–112.
8. **Мухачева А. С., Мухачева Э. А.** Конструирование алгоритмов локального поиска оптимума прямоугольной упаковки на базе двойственных задач линейного раскроя // Информационные технологии. 2002. № 6. С. 25–30.
9. **Мухачева А. С., Чиглинцев А. В.** Генетический алгоритм поиска минимума в задачах двухмерного гильотинного раскроя // Информационные технологии. 2001. № 3. С. 27–32.
10. **Мухачева А. С., Чиглинцев А. В., Смагин М. А., Мухачева Э. А.** Задачи двухмерной упаковки: развитие генетических алгоритмов на базе смешанных процедур локального поиска оптимального решения // Информационные технологии. 2001. № 9. Приложение. 24 с.

11. **Мухачева Э. А., Валеева А. Ф.** Метод динамического перебора в задаче двухмерной упаковки // Информационные технологии. 2000. № 5. С. 30–37.
12. **Мухачева Э. А.** Рациональный раскрой промышленных материалов. Применение в АСУ. М.: Машиностроение, 1984. 176 с.
13. **Мухачева Э. А., Верхотуров М. А., Мартынов В. В.** Модели и методы расчета раскроя-упаковки геометрических объектов // Уфа: УГАТУ, 1998. 216 с.
14. **Мухачева Э. А., Мухачева А. С.** Метод перестройки для решения задачи прямоугольной упаковки // Информационные технологии. 2000. № 4. С. 30–36.
15. **Мухачева Э. А., Мухачева А. С., Белов Г. Н.** Метод последовательного уточнения оценок: алгоритм и численный эксперимент для задачи одномерного раскроя // Информационные технологии. 2000. № 2. С. 11–17.
16. **Мухачева Э. А., Мухачева А. С., Чиглинцев А. В.** Генетический алгоритм блочной структуры в задачах двухмерной упаковки // Информационные технологии. 1999. № 11. С. 13–18.
17. **Норенков И. П.** Эвристики и их комбинации в генетических методах дискретной оптимизации // Информационные технологии. 1999. № 1. С. 2–7.
18. **Романовский И. В.** Решение задачи гильотинного раскроя методом переработки списка состояний // Кибернетика. 1969. № 1. С. 102–104.
19. **Aurts E., Lenstra J.** (Edt). Local Search in Combinatorial Optimization. John Wiley & Sons, 1996.
20. **Belov G.** A branch-and price algorithm for one-and-two dimensional two-stage cutting (stock) problems // Techn. Rep. MATH-NM-03-03. Dresden Univ, 2003. URL: www/math.tu-dresden.de/ capad.
21. **Bischoff E., Waescher G.** (Edt). Special issue: Cutting and packing // Europ. J. of Operat. Res. 1995. 84 (2).
22. **Dykhoff H.** A typology of cutting and packing problems // Europ. J. of Operat. Res. 1990. V. 44, No. 2. P. 145–159.
23. **Dyckhoff H., Wascher G.** (Edit). Special issue: Cutting and packing // Europ. J. of Operat. Res. 1990. V. 44, No. 2.
24. **Folkenauer E.** A hybrid grouping genetic algorithm for bin packing // J. of Heuristics. 1998. 2 (1). P. 5–30.
25. **Gilmore P. C., Gomory R. E.** A linear programming approach to the cutting-stock problem // Operat. Res. 1961. 9. P. 849–859.
26. **Hinxman A.** The trim-loss and assortment problems: a survey // Europ. J. of Operat. Res. 1980. No. 11. P. 863–888.
27. **Hopper E., Turton B.** An empirical investigation of meta-heuristic and heuristic algorithms for a 2D packing problem // EJOR. 128. 2001. P. 34–57.
28. **Imahori S., Yaguira M., Ibaraki T.** Local search heuristics for the rectangle packing problem with general spatial costs // MIC'2001 - 4th Metaheuristics Int. Conf. P. 471–476.
29. **Lirov Y.** (Edt). Special issue: Geometric resource allocation // Math. and Comp. Modelling. 1995. 16 (1).
30. **Liu D., Teng H.** An improved BL-algorithm for genetic algorithm of the orthogonal packing of rectangles // Europ. J. of Operat. Res. 1999. 112. P. 413–420.
31. **Martello S.** (Edt). Special issue: Knapsack, packing and cutting. Part I. One dimensional knapsack problem // INFOR. 1994. 32 (3).
32. **Martello S.** (Edt). Special issue: Knapsack, packing and cutting. Part II. Multidimensional knapsack and cutting stock problems // INFOR. 1994. 32 (4).
33. **Mukhacheva E.** (Edt). Special issue: Decision making under conditions of uncertainty (cutting-packing problems) // The Int. Scient. Collect. Ufa, Russia, 1997.
34. **Terno J., Lindeman R., Scheithauer G.** Zuschnitprobleme und ihre praktische Losung. Leipzig, 1987.
35. **Wang P., Wascher G.** (Edt). Special issue: Cutting packing problems // Europ. J. of Operat. Res. 2002. 141.
36. **Yanasse H.** (Edt). Special issue: Cutting and packing problems // Pesquisa Operacional. 1999. 19 (2).

ОБ АВТОРЕ

Филиппова Анна Сергеевна, доц., каф. выч. мат. и киберн. Дипл. инж.-системотехник (УГАТУ, 1996). Канд. физ.-мат. наук по примен. выч. тех-ки, мат. моделир. и мат. методов в науч. иссл. (защ. в БГУ, 1999). Тр. в обл. прикл. задач иссл. операций.