УДК 519.8

АЛГОРИТМ ИМИТАЦИИ ОТЖИГА В ЗАДАЧЕ ПЛОСКОГО ПРЯМОУГОЛЬНОГО РАСКРОЯ

Козин И.В.[1], Нарзуллаев У.Х.[2], Сардак О.В.[1],Сабиров З.Р.[3]
[1] Запорожский национальный университет, Запорожье, Украина
[2] Ташкентский университет информационных технологий имени Мухаммада ал-Хорезми, Ташкент, Узбекистан
[3] Ургенчский государственный университет, Ургенч, Узбекистан
ainc00@gmail.com, ulug1956_56@mail.ru, karnelcore@gmail.com, sabirovz9625@gmail.com

Аннотация. *Задача плоского прямоугольного раскроя принадлежит к классу NP-трудных задач, то есть для ее точного решения неизвестны алгоритмы полиномиальной трудоемкости. До сих пор не разработано эффективных и достаточно точных способов расчета нижних границ для данной задачи, позволяющих определить достижение оптимума. Таким образом, точные алгоритмы сводятся к полному перебору вариантов. В связи с этим, использование точных алгоритмов для решения задачи плоского прямоугольного раскроя часто оказывается нецелесообразным и невозможным по причине больших затрат времени. Поэтому большое значение уделяется разработке и исследованию эвристических методов оптимизации. В настоящей работе рассмотрен алгоритм имитации отжига и описан вариант этого алгоритма применительно к задаче оптимизации на множестве перестановок. Показано, что ряд классов задач плоского прямоугольного раскроя имеют фрагментарную структуру и, таким образом, поиск оптимальных(субоптимальных) решений этих задач можно свести к поиску оптимальной перестановки. Это позволило создать гибридный алгоритм отыскания субоптимальных решений задач плоского прямоугольного раскроя на основе комбинации алгоритма имитации отжига и фрагментарного алгоритма.*

Ключевые слова: *дискретная оптимизация, метаэвристика, фрагментарная структура, алгоритм имитации отжига, задача плоского прямоугольного раскроя.*

I. ВВЕДЕНИЕ

Для большинства классов задач дискретной оптимизации не известны точные алгоритмы поиска оптимального решения, сложность которых ограничена полиномом от длины условия задачи. В частности, к таким классам относятся различные варианты задачи плоского прямоугольного раскроя [1]. Задачи такого типа часто возникают на практике и поэтому необходимы методы, которые позволяют получить хотя бы приближенные (в каком-либо смысле) решения этих задач. Практически единственным инструментом отыскания приближенных решений подобных задач за приемлемое время являются метаэвристики.

Относительная простота метаэвристик [2,3], ограниченный объем памяти и высокая скорость делают их незаменимыми для решения прикладных задач, в которых требуется получить хотя бы допустимое решение, более или менее отвечающее требованием заказчика. Метаэвристики не имеют теоретического обоснования. Единственным подтверждением их качества является практика, то есть результаты, полученные на тестовых примерах, факты решения реальных прикладных задач и т.п. Но в задачах практики как правило имеется достаточно много ограничений, которые не позволяют создать универсальный метод решения для большого числа прикладных задач. Часто подобные трудности возникают при переходе от непрерывных задач к дискретным. В настоящей работе предлагается подход к дискретным задачам с использованием фрагмен-тарных структур. Такой подход, хотя и не является универсальным, позволяет достаточно просто строить гибридные алгоритмы на основе существующих метаэвристик для большого класса дискретных оптимизационных задач.

В работе предлагается гибридный алгоритм такого типа на основе комбинации фрагментарного алгорит-ма и алгоритма имитации отжига для некоторых классов задач плоского прямоугольного раскроя.

II. АЛГОРИТМ ИМИТАЦИИ ОТЖИГА.

Название алгоритма «имитация отжига» (simulated annealing) предложено в работе [4-6], где рассмотрена метаэвристика, основан-ная на процессе, напоминающем процесс отжига, применяемый в металлургии. В этом процессе металл или сплав нагревается до высокой температуры и охлаждается по некоторому закону до комнатной температуры. Вероятность изменения положения атома в кристаллической решетке p связана с потенциальной энергией решетки E и температурой T согласно закону Максвелла-Больцмана. По этой причине при охлаждении металла общая потенциальная энергия решетки значительно уменьшается, что улучшает характеристики металла.

Для поиска решения оптимизационной задачи "нахождения точки максимума функции $F(x)$ на множестве произвольной природы" алгоритм имитации отжига может быть описан следующим образом:

На каждой итерации алгоритма по определенному правилу генерации генерируется новое решение $x \in X$. Сгенерированное решение x сравнивается с текущим (наилучшим) решением $x^* \in X$. Если решение x «лучше» текущего ($F(x) > F(x^*)$), то оно выбирается в качестве текущего решения для последующей итерации. Если это решение хуже текущего ($F(x) < F(x^*)$), то оно выбирается как текущее решение с вероятностью

$$p = \frac{1}{1 + \exp(\Delta E / T)}$$

Здесь положительный параметр T является аналогом температуры, а величина $\Delta E = F(x) - F(x^*)$ интерпретируется как изменение потенциальной энергии расплава.

Более часто используется приближенная формула для вероятности

$$p = \exp\left(-\frac{\Delta E}{T}\right) = \exp\left(\frac{F(x) - F(x^*)}{T}\right)$$

При использовании этой формулы величина p находится в промежутке (0,1), если выполняется условие $\Delta E < 0$. Возможность выбора в некоторых итерациях «менее оптимального» решения связана с необходимостью избежать преждевременной сходимости к локальному оптимуму. В начале работы алгоритма имитации отжига величина T относительно велика, что означает сравнительно высокую вероятность выбрать менее оптимальное решение. В процессе поиска T уменьшается по некоторым законам, и через несколько итераций менее оптимальное решение выбирается уже со значительно меньшей вероятностью. Функцию $T = T_k$, по которой температура зависит от номера итерации k, называют «функцией понижения температуры», или «функцией охлаждения», по аналогии с физичес-ким процессом отжига. Наиболее простой пример функции уменьшения температуры дает линейная функция вида $T_{k+1} = \beta T_k$, где β – коэффициент охлаждения. Конечно, $0 < \beta < 1$. Другие примеры функции уменьшения температуры и правил генерации приводятся ниже:

1) Больцмановский отжиг: $T_k = \dfrac{T_0}{\ln(1+k)}$, k>0, правило генерации задается нормальным распределением с математическим ожиданием x^* и дисперсией T_k, т.е. задается плотностью распределения

$$\varphi_k(x) = (2\pi T_k)^{-D/2} \exp(-|x - x^*|^2 / 2T_k),$$

где D – размерность пространства поиска, $|x - x^*|$ - расстояние между соответствующими точками. Пространство поиска предполагается метрическим пространством. Для больцмановского отжига подтверждено [5], что при достаточно большом значении параметра T_0 и достаточно большом количестве шагов правило генерации гарантирует нахождение глобального максимума с любой заданной степенью точности. К сожалению, в алгоритме больцмановского отжига температура уменьшается очень медленно. Более быстрые алгоритм рассмотрены ниже.

2) Отжиг Коши (быстрый отжиг) [7]: в этом алгоритме $T_k = \dfrac{T_0}{k}$, k>0, а правило генерации задается нормальным распределением Коши:

$$\varphi_k(x) = \frac{T_k}{\left(|x - x^*|^2 + T^2\right)^{(D+1)/2}}.$$

Есть и другие варианты алгоритма имитации отжига, как правило, определенные на непрерывном прос-транстве поиска.

III. АЛГОРИТМ ИМИТАЦИИ ОТЖИГА ДЛЯ ЗАДАЧИ КОММИВОЯЖЕРА.

Рассмотрим классическую задачу коммивояжера на полном графе K_n с n вершинами [8-11]. Перенумеруем вершины графа числами $1,2,\ldots,n$. Каждому ребру графа (i, j) сопоставлено неотрицательное число $\rho(i, j)$ - вес ребра. Путь коммивояжера с началом и концом в вершине i_1 задается произвольной перестановкой вершин в графе $x = i_1, i_2, \ldots, i_n$. Весом(длиной) пути x будем называть сумму весов ребер этого пути, то есть

$$\rho(x) = \rho(i_1, i_2) + \rho(i_2, i_3) + \ldots + \rho(i_{n-1}, i_n) + \rho(i_n, i_1).$$

Алгоритм имитации отжига для этой задачи будет состоять из следующих шагов:

Шаг 0. На этом шаге задается коэффициент охлаждения $\beta \in (0,1)$, величина максимальной температуры $T_0 > 0$ и максимальное число шагов K алгоритма. Выбирается случайным образом начальная перестановка вершин, задающая маршрут $x^* = (i_1, i_2, ..., i_n)$ и вычисляемая значение целевой функции на этом маршруте:

$$\rho(x^*) = \sum_{l=1}^{n-1} \rho(i_l, i_{l+1}) + \rho(i_n, i_1)$$

….

Шаг k $(1 \le k \le K)$. Случайным образом выбираются две позиции s и t в текущей перестановке $x^* = (i_1, i_2, ..., i_n)$ такие, что $1 \le s < t \le n$. Строим новую перестановку

$$x^k = (i_1, i_2, ..i_{s-1}, i_t, i_{t-1}...i_s, i_{t+1}, ..., i_n'),$$

в которой вершины $i_s, i_{s+1}, ..., i_{t-1}, i_t$ переставляются в обратном порядке. Вычисляется значение целевой функции $\rho(x^k)$ на перестановке x^k. Вычисляется новое значение температуры $T_k = \beta T_{k-1}$. Если $\rho(x^k) \le \rho(x^*)$, то полагаем $x^* = x^k$ и переходим к очередному шагу алгоритма. Если $\rho(x^k) > \rho(x^*)$, то вычисляем вероятность $p_k = \exp\left(\frac{\rho(x^*) - \rho(x)}{T_{k-1}}\right)$. Если $p_k > rand()$, где – $rand()$ - функция, возвращающая равномерно распределенное случайное число в диапазоне [0,1], то полагаем $x^* = x^k$ и переходим к очередному шагу. В противном случае просто переходим к шагу с номером k+1.

Алгоритм заканчивает работу, когда проведено заданное число шагов K. Текущая перестановка x^*, определенная на последнем шаге, берется в качестве оптимального решения задачи.

В модификациях алгоритма могут быть изменены формула для функции понижения температуры и способ генерации решения на очередном шаге.

Заметим, что описанный выше алгоритм решает задачу поиска оптимальной перестановки из n элементов на множестве всех таких перестановок с целевой функцией $\rho(x)$, которая задана на множестве перестановок. Далее будет показано, что этот алгоритм может быть обобщен на широкий класс дискретных оптимизационных задач, а именно задач, имеющих фрагментарную структуру.

IV. ЗАДАЧИ С ФРАГМЕНТАРНОЙ СТРУКТУРОЙ

Фрагментарной структурой [12] (X, E) на конечном множестве X называется семейство его подмножеств $E = \{E_1, E_2, ..., E_n\}$ такое, что $\forall E_i \in E$, $E \ne \varnothing$ $\exists e \in E_i : E_i \setminus \{e\} \in E$.

Элементы из множества E будем называть допустимыми фрагментами. Таким образом, для любого допустимого фрагмента E_i существует нумерация его элементов $E_i = \{e_{i1}, e_{i2}, ...e_{is_i}\}$ такая, что $\forall k = 1, 2, ...s_i$ $\{e_{i1}, e_{i2}, ...e_{ik}\} \in E$. Элементарным фрагментом будем называть допустимый фрагмент, состоящий из одного элемента. Максимальный фрагмент – допустимый фрагмент, который не является подмножеством никакого другого фрагмента.

Максимальный фрагмент может быть построен с помощью следующего "жадного" алгоритма:

а) элементы множества X линейно упорядочиваются;

б) на начальном шаге выбирается пустое множество $X_0 = \varnothing$;

в) на шаге с номером k + 1 выбирается первый по порядку элемент $x \in X \setminus X_k$, такой, что $X_k \bigcup \{x\} \in E$;

г) алгоритм заканчивает работу, если на очередном шаге не удалось найти элемент $x \in X \setminus X_k$ с требуемым свойством.

Результат работы алгоритма определяется заданным линейным порядком на множестве *X*. Таким образом, любой максимальный фрагмент может быть описан некоторой перестановкой элементов множества *X*. Пусть $A \in E$. Условие для элемента $x \in X$, при котором $A \bigcup \{x\} \in E$, будем называть условием присоединения элемента *x*.

Пусть теперь каждому фрагменту приписан вес, то есть задана функция $\rho : E \to R^1$. Будем предполагать, что функция ρ монотонна по включению (возрастающая или убывающая). Если. $A, B \in E$ и $A \subseteq B$, то $\rho(A) \leq (\geq) \rho(B)$. Задача оптимизации на фрагментарной структуре, это задача отыскания допустимого фрагмента максимального (минимального) веса. Очевидно, что для монотонных весов оптимальное решение будет являться максимальным фрагментом.

Любой максимальный фрагмент определяется заданным линейным порядком просмотра элементарных фрагментов. Этот порядок определяет результат работы фрагментарного алгоритма, который и построит требуемый максимальный фрагмент.

Каждый линейный порядок определяется некоторой перестановкой $s \in S_n$ элементарных фрагментов (*n* – число элементарных фрагментов). Сопоставим каждой перестановке максимальный фрагмент, который ей порождается. Обозначим это отображение через $\varphi : S_n \to E$. Таким образом, имеет место естественная коммутативная диаграмма отображений

$$\begin{array}{c} S_n \\ \varphi \downarrow \quad \square \quad F \circ \varphi, \\ E \to R^1 \end{array}$$

которая превращает задачу оптимизации на фрагментарной структуре в задачу оптимизации на множестве перестановок. Причем любая перестановка является допустимой. Для больших значений n задача поиска оптимальной перестановки, как правило, является трудной в вычислительном смысле. Поэтому для таких задач оправдано применение метаэвристик. В частности, алгоритм имитации отжига, который был описан для задачи коммивояжера автоматически переносится на любую задачу с фрагментарной структурой. Изменяется лишь правило отыскания значений целевой функции. Для получения этих значений используется фрагментарный алгоритм, который является индивидуальным для каждого класса дискретных оптимизационных задач с фрагментарной структурой. В качестве примера использования алгоритма имитации отжига рассмотрим задачу плоского прямоугольного раскроя.

V. ЗАДАЧИ ПЛОСКОГО ПРЯМОУГОЛЬНОГО РАСКРОЯ

Рассмотрим следующую постановку задачи плоского прямоугольного раскроя [13-16]:

В качестве матрицы-основы берется прямоугольник размерности $H \times W$, где *H*, *W* – положительные целые числа – высота и ширина прямоугольника. Задано множество прямоугольных заготовок, стороны которых также выражаются положительными целыми числами. Требуется разместить прямоугольные заготовки на матрице-основе таким образом, чтобы площадь части основы, не занятой заготовками, была минимальной.

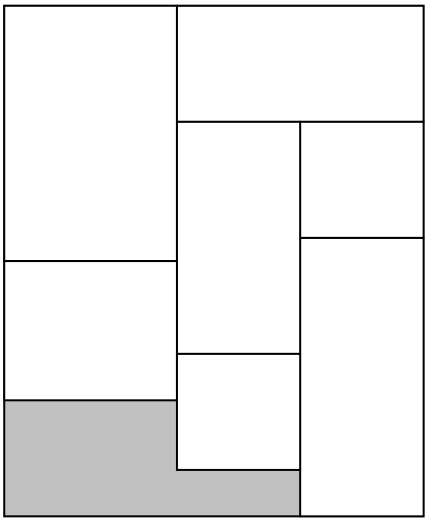

Рис. 1. Карта раскроя

Заготовки могут пересекаться лишь по границе. Пример такого размещения приведен на рис. 1. Незанятая часть основы выделена серым цветом. Пусть множество заготовок представлено набором пар $\{(h_i, w_i)\}$, где h_i, w_i - высота и ширина соответствующей заготовки, $i = 1, 2, \ldots, N$, N – число заготовок. Допустимое решение задачи можно представить набором троек $\{(\delta_i, t_i, l_i)\}_{i=1}^{N}$. Здесь

$$\delta_i = \begin{cases} 1, \text{ если } i\text{-я заготовка входит в покрытие,} \\ 0, \text{ если } i\text{-я заготовка не входит в покрытие,} \end{cases}$$

t_i, l_i – соответственно координаты левого верхнего угла заготовки относительно матрицы основы.

Покажем, что рассматриваемая задача может быть представлена, как задача на фрагментарной структуре. Каждый элементарный фрагмент соответствует заготовке и координатами ее левого верхнего угла на матрице основе. Условие присоединения – размещаемая на матрице основе заготовка не принадлежит к множеству уже выбранных заготовок и не пересекается с уже размещенными на матрице заготовками. Принцип выбора места размещения Top-Left – верхний левый свободный угол.

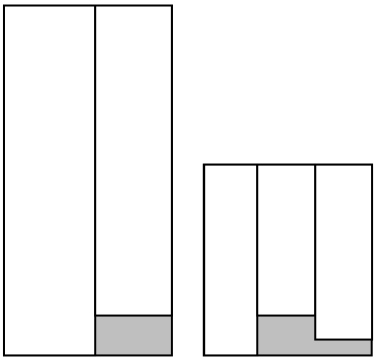

Рис. 2. Вертикальные слои

Для этого варианта задачи раскроя предложенная выше фрагментарная модель задачи не обладает свойством достижимости. А именно: не всегда существует перестановка элементарных фрагментов, для которой фрагментарный алгоритм дает оптимальное решение задачи раскроя. Однако в частном случае, когда существует оптимальное решение задачи раскроя, в котором площадь, не занятая заготовками равна 0, всегда найдется такая перестановка фрагментов, которая реализует оптимальное решение задачи.

В качестве следующего примера рассмотрим задачу гильотинного раскроя. Здесь заданный набор прямоугольных заготовок должен быть уложен на бесконечной полосе заданной ширины W таким образом, чтобы при раскрое на элементы использовались лишь гильотинные разрезы «от края до края».

На первом этапе заготовки близкие по ширине или длине укладываются в вертикальные или горизонтальные слои (рис.2, 3). Слой рассматривается как прямоугольник минимального размера, содержащий все входящие в него заготовки.

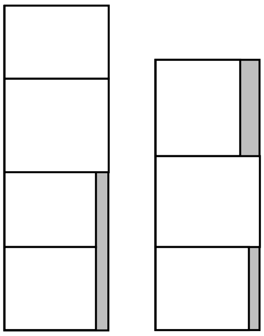

Рис. 3. Горизонтальный слой

Затем слои размещаются на матрице основе (рис.4). Алгоритм размещения предлагается следующий: слои укладываются в заданной последовательности по правилу **Top-Left** с фиксированным верхним краем до достижения границы основы.

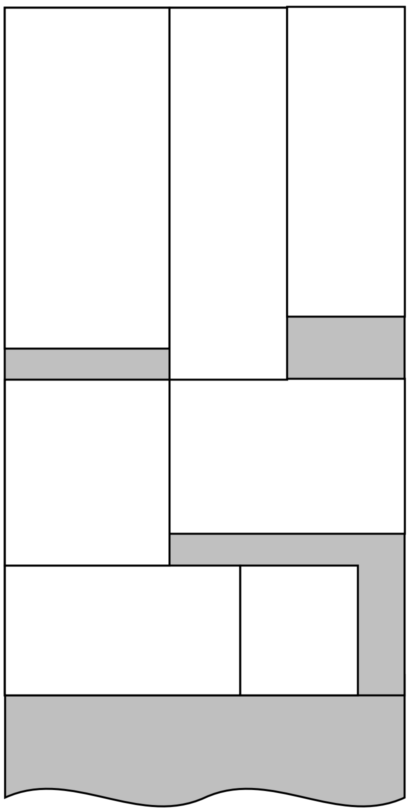

Рис. 4. Гильотинный раскрой

Если очередной слой не помещается в основу, то он переносится вниз и выравнивается по верхнему краю, граница которого определяется максимальной нижней границей уже упакованных слоев.

Критерием оптимальности в этой задаче является ордината максимальной из нижних границ уложенных в полосу слоев.

В качестве элементарного фрагмента выбираем тройку – слой и координаты его левого верхнего угла на матрице основе.

В такой модели гильотинного раскроя свойство достижимости отсутствует, однако «жадные алгоритмы», построенные по предложенной схеме, часто применяется в прикладных задачах гильотинного раскроя [13,16].

Поскольку задача плоского прямоугольного раскроя может быть сведена к задаче оптимизации на множестве перестановок размерности $N(N-1)/2$ (N-число заготовок), то к ней практически без изменений может быть применен алгоритм имитации отжига, который был описан выше для задачи коммивояжера. Это позволяет достаточно эффективно (по времени) получать субоптимальные решения задачи плоского прямоугольного раскроя.

VI. ЗАКЛЮЧЕНИЕ

В работе было показано, что известная метаэвристика, основанная на алгоритме имитации отжига, который хорошо зарекомендовал себя в ряде прикладных задач, может с успехом применятся для различных классов дискретных оптимизационных задач, имеющих фрагментарную структуру. Для решения задачи размещения производства приведен простой гибридный алгоритм, основанный на комбинации алгоритма имитации отжига и фрагментарного алгоритма. Предлагаемая методика построения гибридных алгоритмов легко может быть расширена и на другие варианты метаэвристик, которые разработаны для поиска субоптимальных решений классичес-кой задачи коммивояжера.

ЛИТЕРАТУРА

[1] *Гэри М., Джонсон Д.,* Вычислительные машины и труднорешаемые задачи -М.: Мир, 1982. 416 с.

[2] *Щербина О.А.,* Метаэвристические алгоритмы для задач комбинаторной оптимизации (обзор) // Таврический вестник информатики и математики. – 2014. – № 1. –С. 56-72.

[3] *Карпенко А.П.* Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие для вузов - Москва: Изд-во МГТУ им. Н. Э. Баумана, 2014, 446с.

[4] *Лопатин А. С.,* Метод отжига / Стохастическая оптимизация в информатике. СПб., Изд-во СПбГУ, 2005. Вып. 1. С. 133–149.

[5] *Kirkpatrick S., Gelatt C.D., and Vecchi M.P.,* Optimization by simulated annealing Science. –1983. – 220(4598). – P. 671-680.

[6] *van Laarhoven P.J.M. and Aarts E.H.L.,* Simulated Annealing: Theory and Applications. – Dordrecht:Springer, 1987.

[7] *Szu H. H., Hartley R. L.,* Fast Simulated Annealing / Physical Letters A. 122. 1987. P. 157-162.

[8] *Кристофидес Н.* Теория графов. Алгоритмический подход. / -М: Мир, 1978, 432с.

[9] *Deineko V., Tiskin A.,* Fast minimum-weight double-tree shortcutting for metric TSP: is the best one good enough? / ACM Journal on Experimental Algorithmics. 2009. Vol. 14. N 4.

[10] *Сигал И.Х., Иванова А.П.,* Введение в прикладное дискретное программирование: модели и вычислительные алгоритмы. - М.:ФИЗМАТЛИТ, 2007, 304с.

[11] *Пападимитриу Х., Стайглиц К.,* Комбинаторная оптимизация. Алгоритмы и сложность. - М.: Мир, 1985, 510с.

[12] *I. V. Kozin, N. K. Maksyshko, V. A. Perepelitsa,* Fragmentary Structures in Discrete Optimization Problems, //Cybernetics and Systems Analysis November 2017, Volume 53, Issue 6, pp 931–936 DOI:https://doi.org/10.1007/s10559-017-9995-6

[13] *Мухачева Э.А.,* Рациональный раскрой промышленных материалов. Применение АСУ: монография -М.: Машиностроение, 1984. - 176 с.

[14] *Козин И. В., Полюга С.И.,* Использование ЭВФ-алгорит-мов для решения задачи прямоугольного раскроя / Питання прикладної математики і математичного моделювання: зб. наук. праць; [ред. кол. … О. М. Кисельова (голов. ред.) та ін.]. – Д. : Вид-во Дніпропетр. нац. ун-ту ім. Олеся Гончара,. – Дніпропетровськ, 2009. – С. 199-208.

[15] *Ермаченко А.И.* Модели и методы решения задач прямоугольного раскроя и упаковки

на базе метаэвристики «Поиск с запретами» [Электронный ресурс] : дис. … канд. тех. наук : 05.13.18 / Александр Иванович Ермаченко. - Уфа, 2004. – 95 с. – Режим доступа: http://www.dissercat.com/content/modeli-i-metody-resheniya-zadach-pryamougolnogo-raskroya-i-upakovki-na-baze-metaevristiki-po.

[16] *Чеканин В.А.* Модифицированные эволюционные алгоритмы и программные решения задачи ортогональной упаковки объектов [Электронный ресурс] : дис. … канд. тех. наук : 05.13.17 / Владислав Александрович Чеканин. - Москва, 2011. – 161 с. – Режим доступа: http://www.dissercat.com/content/modifitsirovannye-evolyutsionnye-algoritmy-i-programmnye-resheniya-zadachi-ortogonalnoi-upak

Поступила в редакцию 21.12.2022

Цитирование: *Козин И.В., Нарзуллаев У.Х., Сардак О.В., Сабиров З.Р.* (2023). Алгоритм имитации отжига в задаче плоского прямоугольного раскроя. *Международный Журнал Теоретических и Прикладных Вопросов Цифровых Технологий*, 1(3), –С. 16-24.

SIMULATION OF ANNEALING ALGORITHM FOR THE FLAT RECTANGULAR CUTTING PROBLEM

Kozin I.V.[1], *Narzullaev U.Kh.*[2], *Sardak O.V.*[1], *Sabirov Z.R.*.[3]

[1] Zaporizhzhia National University, Zaporizhzhia, Ukraine,

[2] Samarkand branch of Tashkent University of information technologies named after Muhammad al-Khwarizmi, Samarkand, Uzbekistan,

[3] Urgench state university, Urgench, Uzbekistan,

ainc00@gmail.com, ulug1956_56@mail.ru, karnelcore@gmail.com, sabirovz9625@gmail.com

Abstract. *The problem of flat rectangular cutting belongs to the class of NP-hard problems, that is, for its exact solution, algorithms of polynomial complexity are unknown. Until now, there have not been developed effective and sufficiently accurate methods for calculating the lower bounds for this problem, which make it possible to determine the achievement of the optimum. Thus, exact algorithms are reduced to a complete enumeration of options. In this regard, the use of exact algorithms for solving the problem of flat rectangular cutting often turns out to be inappropriate and impossible due to the large time costs. Therefore, great importance is given to the development and research of heuristic optimization methods. In this paper, we consider an annealing simulation algorithm and describe a variant of this algorithm as applied to an optimization problem on a set of permutations. It is shown that a number of classes of problems of flat rectangular cutting have a fragmented structure and, thus, the search for optimal (suboptimal) solutions to these problems can be reduced to the search for an optimal permutation. This made it possible to create a hybrid algorithm for finding suboptimal solutions to problems of flat rectangular cutting based on a combination of an annealing simulation algorithm and a fragmentary algorithm.*

Keywords: *discrete optimization, metaheuristics, fragmentary structure, annealing simulation algorithm, flat rectangular cutting problem.*