

УДК 004.023

О.П. Тимофеева, Т.Ю. Чернышева, О.Н. Корелин, А.В. Волков

**ГЕНЕТИЧЕСКИЙ АЛГОРИТМ В ОПТИМИЗАЦИИ
ТРЕХМЕРНОЙ УПАКОВКИ БЛОКОВ В КОНТЕЙНЕР**

Нижегородский государственный технический университет им. Р.Е. Алексеева

Цель: Построение модели и алгоритма решения задачи трехмерной упаковки блоков, имеющих форму параллелепипедов, в контейнер.

Методология / подход: Применение генетического алгоритма.

Результаты: В результате тестирования получено решение задачи упаковки блоков в контейнер, обладающее высокой плотностью упаковки.

Применение: Решение задачи раскроя-упаковки на производстве, в транспортной логистике.

Ключевые слова: генетический алгоритм, методы оптимизации, трехмерная упаковка параллелепипедов в контейнер, настройка генетического алгоритма, декодер.

Задачи упаковки представляют собой важный прикладной раздел комбинаторной оптимизации. Высокий интерес к задачам упаковки среди исследователей связан с широким спектром применения их решений в различных сферах экономической деятельности. К задаче упаковки сводится множество практических задач автоматизации и управления, в частности, задачи эффективного размещения объектов, составления расписаний, планирования распределения ресурсов в вычислительных сетях, геометрического проектирования, управления процессами обработки данных и т.д.

От качества получаемого решения данной задачи зависят:

- во-первых, процент задействованного материала при раскрое;
- во-вторых, продуктивность использования раскройного оборудования;
- в-третьих, время на процесс проектирования и производительность труда.

В связи с этим, важной и актуальной задачей является разработка новых методов и алгоритмов решения задачи раскроя-упаковки.

Сложность решения задачи упаковки обусловлена её принадлежностью к классу NP-полных задач, для решения которых не существует детерминированных алгоритмов полиномиальной сложности. Практическое применение точных методов для решения NP-полных задач упаковки большой размерности, использующих полный перебор вариантов, невозможно из-за больших затрат временных и вычислительных ресурсов.

Ситуация осложняется отсутствием математических моделей и алгоритмов, гарантирующих получение оптимального решения для большинства задач упаковки. В связи с этим, одним из наиболее перспективных направлений исследований являются разработка и совершенствование различных приближенных, в том числе эвристических методов решения задач упаковки, позволяющих получать квазиоптимальные решения за полиномиальное время. Наиболее эффективными и хорошо зарекомендовавшими себя при решении таких задач являются эволюционные алгоритмы, в частности генетические [1–3].

Постановка задачи

Рассмотрим область трехмерного пространства шириной W , глубиной D и высотой H , называемую контейнером – $M = W \times D \times H$. Имеется множество из N блоков, имеющих форму параллелепипедов, которые необходимо разместить в контейнере M таким образом, чтобы значение целевой функции было наилучшим с точки зрения используемого критерия оптимальности. Результат упаковки будем представлять в виде последовательности параллелепипедов в списке, представляющем очередность их укладки в контейнер.

Каждый блок описывается кортежем из шести элементов: $\{x_0, y_0, z_0, w_i, d_i, h_i\}$, где x_0, y_0, z_0 - координаты расположения самого близкого к началу координат угла блока, w_i, d_i, h_i - ширина, глубина и высота блока соответственно ($i = \overline{1, N}$).

Введем ограничения [4], которые следует учитывать при реализации алгоритма упаковки:

1. Ни один блок не может выходить за границы заданного контейнера, следовательно, его координаты должны удовлетворять системе неравенств

$$\begin{cases} x_{0i} \geq 0, \\ y_{0i} \geq 0, \\ z_{0i} \geq 0, \\ x_{0i} + w_i \leq W, \\ x_{0i} + d_i \leq D, \\ x_{0i} + h_i \leq H, \end{cases}$$

где $\{x_{0i}, y_{0i}, z_{0i}\}, \{x_{0i} + w_i, y_{0i} + d_i, z_{0i} + h_i\}$ - координаты углов блока - самого близкого к началу осей координат и самого удаленного соответственно.

2. Суммарный объем блоков не может превышать объема контейнера:

$$\sum_{i=1}^N (w_i d_i h_i) \leq WDH.$$

3. Блоки не могут накладываться друг на друга (в одну область контейнера нельзя поместить несколько блоков):

$$\begin{aligned} & \left(x_{0i} + w_i \geq x_{0j} \ \& \ y_{0i} + d_i \geq y_{0j} \ \& \ z_{0i} + h_i \geq z_{0j} \right) \vee \\ & \left(x_{0i} + w_i \leq x_{0j} \ \& \ y_{0i} + d_i \geq y_{0j} \ \& \ z_{0i} + h_i \geq z_{0j} \right) \vee \\ & \left(x_{0i} + w_i \geq x_{0j} \ \& \ y_{0i} + d_i \leq y_{0j} \ \& \ z_{0i} + h_i \geq z_{0j} \right) \vee \\ & \left(x_{0i} + w_i \geq x_{0j} \ \& \ y_{0i} + d_i \geq y_{0j} \ \& \ z_{0i} + h_i \leq z_{0j} \right) \vee \\ & \left(x_{0i} + w_i \leq x_{0j} \ \& \ y_{0i} + d_i \leq y_{0j} \ \& \ z_{0i} + h_i \geq z_{0j} \right) \vee \\ & \left(x_{0i} + w_i \leq x_{0j} \ \& \ y_{0i} + d_i \leq y_{0j} \ \& \ z_{0i} + h_i \leq z_{0j} \right) \vee \\ & \left(x_{0i} + w_i \leq x_{0j} \ \& \ y_{0i} + d_i \geq y_{0j} \ \& \ z_{0i} + h_i \leq z_{0j} \right); \forall i \leq N, j \leq N (i \neq j). \end{aligned}$$

В качестве критерия оптимизации будем использовать отношение полезного объема, занимаемого блоками (с учетом образовавшихся между блоками пустот) к общему объему контейнера:

$$F = \frac{V_{\text{пол}}}{V_{\text{кон}}} \rightarrow \min,$$

где $V_{\text{пол}}$ - полезный объем, $V_{\text{кон}}$ - объем контейнера.

Таким образом, целью решения задачи является уменьшение пустот в области упаковки. В идеальном случае, когда целевая функция принимает минимальное значение, полезный объем равен суммарному объему блоков.

На контейнер накладывается ограничивающее условие: контейнер имеет фиксированные размеры основания: ширину и глубину (W, D) и теоретически неограниченную высоту (H). Практически эта высота ограничена очень большим, недостижимым размером (суммарной высотой блоков, которые предстоит упаковать в контейнер). Полученная таким образом высота используется при расчете объема контейнера ($V_{\text{кон}}$).

Методика кодирования-декодирования решений генетического алгоритма

Для решения поставленной задачи за основу был взят классический генетический алгоритм. Определим основные элементы генетического алгоритма в терминах рассматриваемой задачи. В качестве хромосомы будет выступать приоритетный список $PL = \{p_1, p_2, \dots, p_n\}$, представляющий собой последовательность блоков, соответствующую очередности, в которой они будут упаковываться в контейнер. Хромосомы состоят из более мелких элементов – генов. В данном случае элемент p_i (ген) хранит параметры p_i -го блока (габаритные размеры и координаты самого ближнего от начала координат угла). Все генетические операции алгоритма производятся над набором хромосом, который называется популяцией.

При решении задачи с помощью генетического алгоритма при заданной кодировке генов для определения пригодности полученного решения (шаблона упаковки) необходимо использовать декодер – оператор, позволяющий получить графическое представление решения задачи по приоритетному списку. С помощью декодера можно получить координаты блоков, найти значение целевой функции полученного решения.

Для решения поставленной задачи было решено использовать основную идею *BL*- и *IBL*-декодеров [5] для задач двумерного раскроя-упаковки и модифицировать их алгоритмы в соответствии со спецификой решаемой задачи для трехмерного пространства. В результате был разработан и реализован декодер *BFL* (*Bottom-Far-Left*, Нижний-Дальний-Левый). По приоритетному списку он формирует карту упаковки блоков в контейнер. Первый блок всегда помещается в начало координат. Также принимается условие, что перед началом упаковки каждый последующий блок будет расположен в верхнем правом ближнем углу (если смотреть напротив начала координат) выше контейнера, координаты ее ближнего к началу координат угла равны $\{W - x_{0i}, D - y_{0i}, H\}$. Затем этот блок перемещается вниз (заполняем сначала первый нижний слой контейнера) насколько это возможно (т.е. пока не столкнется с уже размещенными блоками или не достигнет дна контейнера), затем вдаль насколько возможно и наконец влево.

После того, как на первый слой не поместится очередной блок, начинается укладка следующего, второго, слоя. Это продолжается до тех пор, пока все блоки не будут помещены в контейнер.

Определение столкновения очередного блока происходит в соответствии с заданными ограничениями, регулирующими тот факт, что блоки не могут находиться в одной и той же области контейнера: реализовано сравнение координат самого дальнего угла уже размещенного блока и самого ближнего угла размещаемого (нового блока).

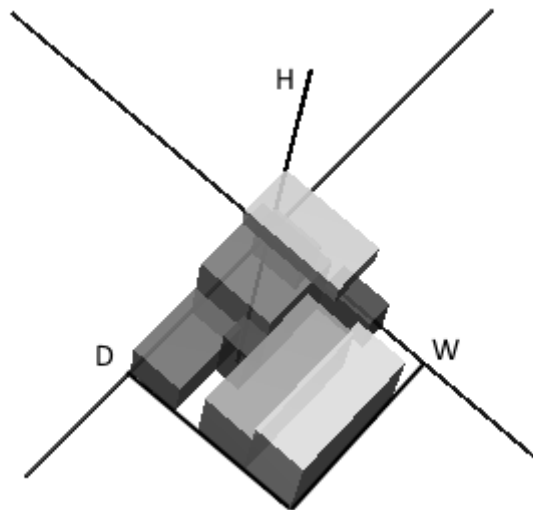


Рис. 1. Пример результата работы декодера

На рис. 1 представлен пример результата декодирования приоритетного списка в трехмерном пространстве шириной W , глубиной D и неограниченной высотой H .

Описание модифицированного генетического алгоритма

Учитывая специфику поставленной задачи, был реализован генетический алгоритм, адаптированный к трехмерной задаче раскроя-упаковки. Рассмотрим его подробнее.

На начальном этапе происходит генерация особей популяции. Создается множество блоков, размеры которых формируются случайным образом. Алгоритм позволяет задать количество блоков и максимальный размер блока по любой из координат. Затем из них формируются хромосомы (особи) - списки блоков в различных последовательностях. Количество особей в популяции задается отдельным параметром.

Перед началом основного цикла алгоритма происходит вычисление целевой функции для каждой особи. Целевая функция задается как отношение полезного объема, занимаемого блоками, к объему контейнера. Полезный объем рассчитывается по наибольшим координатам самых удаленных от начала координат блоков, что позволяет учесть объем пустот, образовавшихся при упаковке. Также для настройки и тестирования параметров и генетических операторов алгоритма было введено понятие плотности, которая выражается как отношение суммарного объема блоков (так называемого идеального) к занимаемому объему после операции декодирования:

$$P = \frac{\sum_{i=1}^N V_{\text{бл}}^i}{V_{\text{пол}}},$$

где $V_{\text{бл}}^i$ – объем i -го блока, $V_{\text{пол}}$ – занимаемый объем после декодирования (полезный объем). В наилучшем случае плотность будет равна 1, т.е. будет равна суммарному объему блоков.

Для решения трехмерной задачи раскроя-упаковки необходимо также модифицировать стандартные генетические операторы.

Селекция хромосом заключается в выборе (по рассчитанным значениям целевой функции) тех хромосом, которые будут участвовать в создании потомков для следующей популяции, т.е. для очередного поколения. Такой выбор производится согласно принципу естественного отбора, и с учетом смысла критерия оптимальности наибольшие шансы на участие в создании новых особей имеют хромосомы с наименьшими значениями целевой функции. Существуют различные методы селекции [6]. В работе исследуются два метода селекции: метод колеса рулетки и метод турнирной селекции.

Идея метода колеса рулетки заключается в том, что для каждой особи выделяется некий сектор колеса рулетки, размер которого пропорционален значению целевой функции. Чем больше значение целевой функции, тем больше сектор на колесе рулетки. Все колесо рулетки соответствует сумме значений целевых функций всех хромосом рассматриваемой популяции. Затем случайным образом выбирается некоторое число. В результате селекции выигрывает та особь, чей сектор содержит данное число.

Таким образом отбирается несколько особей из общего количества особей в популяции для дальнейшего участия в алгоритме. Вероятность селекции задается как параметр алгоритма. Слабая сторона этого метода заключается в том, что особи с очень малым значением целевой функции слишком быстро исключаются из популяции, что может привести к преждевременной сходимости генетического алгоритма.

При отборе методом турнирной селекции все особи популяции разбиваются на подгруппы с последующим выбором в каждой из них особи с наилучшей целевой функцией. Различаются два способа такого выбора: детерминированный выбор и случайный выбор. Детерминированный выбор осуществляется с вероятностью, равной 1, а случайный выбор – с вероятностью, меньшей 1. В общем случае подгруппы могут иметь произвольный размер, в работе используется размер подгруппы, равный 2-3 особям.

Таким образом создается временная популяция, состоящая из хромосом, отобранных в

результате селекции и предназначенных для дальнейших преобразований операторами скрещивания и мутации с целью формирования новой популяции.

Для скрещивания были исследованы одно- и двухточечный кроссинговер.

Идея метода одноточечного скрещивания заключается в том, что выбираются две особи-родителя. Затем случайным образом выбирается место для разрыва между двумя позициями генов в обеих хромосомах – точка скрещивания. В результате скрещивания пары особей-родителей получаются два потомка, хромосомы каждого из которых состоят из части хромосом одного из родителей до точки скрещивания и части хромосомы другого родителя после точки скрещивания.

Основная идея двухточечного метода скрещивания схожа с одноточечным методом, однако вместо одной точки скрещивания используются две.

Оба этих метода для данной кодировки решения задачи приходится модифицировать, так как при использовании стандартных операторов могли бы получиться приоритетные списки, содержащие по несколько одинаковых блоков, что противоречит условию задачи. Поэтому во вторую часть потомка дописываются недостающие номера блоков (гены) в том порядке, в каком они встречаются во втором родителе. Аналогично получаем и второго потомка. В результате работы оператора скрещивания в новой популяции остаются оба потомка.

Мутация – редкое изменение случайного гена в хромосоме – в классической интерпретации предполагает изменения значения выбранного гена на противоположное. Но применимо к реалиям данной задачи оператор мутации был модифицирован. В рассматриваемом случае происходит перемещение случайно выбранного блока (гена) в начало приоритетного списка.

Тестирование

Для реализации разработанного алгоритма была написана программа в среде IntelliJ IDEA 2016.3 на языке Java для 64 разрядной операционной системы Windows 7, которая позволяет пользователю решать задачу раскроя материала. Отладка и тестирование разработанных алгоритмов выполнялись на компьютере с процессором Intel Core i5-2410M CPU с ОЗУ- 6Гб. Проведен вычислительный эксперимент.

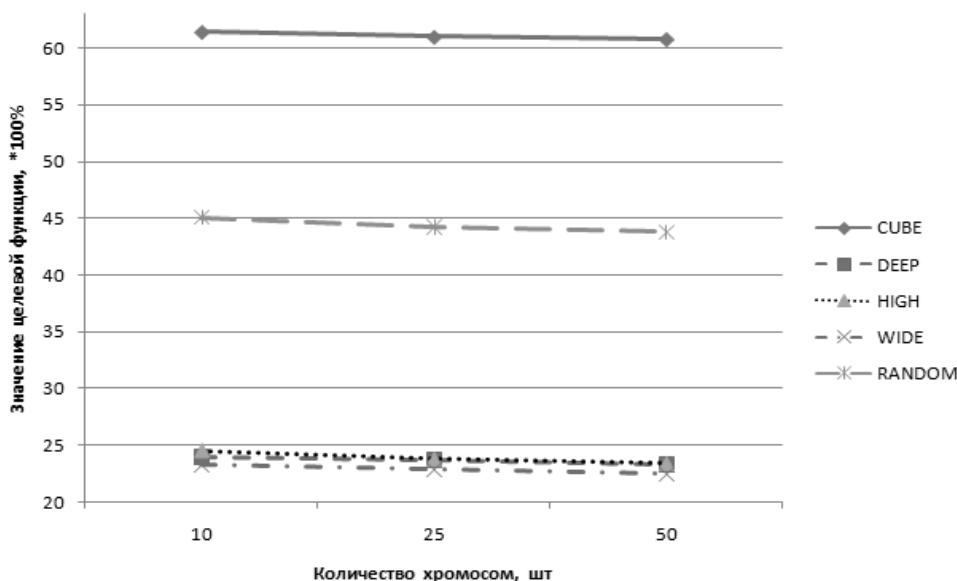


Рис. 2. График зависимости значения целевой функции от размера популяции для различных типов блоков

Для определения эффективности разработанного подхода были проведены исследования времени и качества решения для разного набора тестовых примеров, различающихся

количеством и типом блоков, размерами контейнера. Были выбраны следующие типы блоков: CUBE - блоки кубической формы, RANDOM - блоки произвольной формы; WIDE, DEEP и HIGH - блоки, у которых размер одного из параметров был значительно больше (по ширине x , глубине y и высоте z соответственно). Вычислительный эксперимент позволил провести настройку параметров генетического алгоритма и выбрать наиболее эффективное сочетание для каждого типа блоков. Критерием остановки выступало количество итераций алгоритма. Для каждого типа примеров алгоритму отводилось определенное количество итераций и размер популяции. Результаты проведенных исследований представлены в табл. 1 и на рис. 2.

Как видно из графика (рис. 2), наилучшим образом укладываются блоки, удлиненные по одному из параметров; а наихудшим – блоки, имеющие кубическую форму, что не противоречит здравому смыслу.

Таблица 1
Результаты вычислительного эксперимента

Размер популяции	R+OP	R+TP	T+OP	T+TP
	CUBE			
50	52,44418	51,90319	51,68638	52,28019
25	53,3695	53,14123	52,53	53,0802
10	53,20403	53,32949	53,86519	53,5805
RANDOM				
50	48,84218	49,00539	49,6889	48,9609
25	49,84583	49,6616	49,24949	49,65528
10	50,44537	49,99519	50,21232	50,16539
WIDE				
50	32,74189	32,35419	32,56698	32,4078
25	32,71248	32,91715	33,22148	32,8256
10	33,26711	32,95298	33,41968	33,35641
DEEP				
50	31,6328	31,82212	31,86514	32,14167
25	32,04756	31,94267	32,08467	32,17055
10	32,70844	32,87156	32,72581	32,90807
HIGH				
50	34,07616	34,42231	34,09106	34,23434
25	34,46524	34,73497	34,5102	34,69937
10	34,87977	34,90026	34,84237	34,83577

Для каждого типа блоков был проведен расчет с помощью всех возможных комбинации методов селекции и скрещивания (R - метод рулетки, T - метод турниров, OP - одноточечный и TP - двухточечный методы скрещивания). Результаты вычислительного эксперимента приведены в табл. 1. Анализ показал, что для блоков типа WIDE, DEEP или HIGH предпочтительнее применять комбинацию из метода рулетки и одноточечного кроссовера (R+OP, R+TP); для блоков типа CUBE - комбинацию метода турниров и одноточечного кроссовера (T+OP), а для блоков произвольной формы (RANDOM) - можно воспользоваться любой из комбинаций. Кроме того, анализ данных вычислительного эксперимента показывает, что целевая функция напрямую зависит от размера популяции: лучшее значение имеют расчеты, проведенные с большим размером популяции (рис. 2), (табл.1).

Выводы

Разработан подход для решения задачи упаковки блоков, имеющих форму параллелепипедов, в трехмерный контейнер ограниченных размеров. Отличительной особенностью данного подхода является использование модифицированного генетического алгоритма, а также разработанная процедура кодирования-декодирования. Подход позволяет получать квазиоптимальные решения задач упаковки за полиномиальное время.

Разработано приложение на языке Java, реализующее адаптированный к задаче упаковки генетический алгоритм с возможностью настройки параметров в зависимости от исходных данных. Проведен вычислительный эксперимент, который позволил выяснить, какое сочетание параметров генетического алгоритма дает лучший вариант упаковки блоков определенной формы.

Библиографический список

1. **Курейчик, В.В.** Применение генетического алгоритма решения задачи трехмерной упаковки / В.В. Курейчик, Д.В. Заруба, Д.Ю. Запорожец // Известия ЮФУ. Технические науки. – 2012. – № 7 (132). – С. 8–14.
2. **Тимофеева, О.П.** Генетический алгоритм в оптимизации упаковки контейнеров / О.П. Тимофеева, Э.С. Соколова, К.В. Милов // Труды НГТУ им. Р.Е. Алексеева. – 2012. – № 4 (101). – С. 167–172.
3. **Заруба, Д.В.** Использование методов эволюционной оптимизации для решения задач трехмерной упаковки / Д.В. Заруба, Д.Ю. Запорожец, Ю.А. Кравченко // Информатика, вычислительная техника и инженерное образование. – 2012. – № 2 (9). – С. 1–6.
4. **Гладков, Л.А.** Решение задачи трехмерной упаковки разногабаритных объектов с использованием бионических методов / Л.А. Гладков, Н.В. Гладкова, Е.С. Скубриева // Известия ЮФУ. Технические науки. – 2013. – № 7 (144). – С. 35–41.
5. **Тимофеева, О.П.** Использование модифицированных BL- и IBL-декодеров при решении задачи фигурного нерегулярного ортогонального раскроя / О.П. Тимофеева, К.В. Милов, А.Е. Миндров // Системы управления и информационные технологии. – 2015. – № 1.1 (59). – С. 177–180.
6. Генетические алгоритмы [Электронный ресурс] // Портал искусственного интеллекта: сайт. – URL: <http://www.aiportal.ru/articles/genetic-algorithms/genetic-algorithms.html> (дата обращения 18.11.2016)

*Дата поступления
в редакцию 13.04.2017*

O.P. Timofeeva, T.Yu. Chernysheva, O.N. Korelin, A.V. Volkov

GENETIC ALGORITHM IN OPTIMIZATION OF THREE-DIMENSIONAL PACKAGING OF BLOCKS INTO CONTAINER

Nizhny Novgorod state technical university n.a. R.E. Alexeyev

Purpose: Creation of model and algorithm to resolve the problem of three-dimensional packing of of parallelepiped-shaped blocks into a container.

Design/methodology/approach: Application of genetic algorithm.

Findings: Solution for problem of packing blocks into a container has been found as a result of testing. This decision possesses the high density of containers packing.

Research limitations/implications: The solution of cutting-packing problem in manufacturing, in transport logistics.

Key words: genetic algorithm, optimization methods, three-dimensional packing of parallelepipeds in container, genetic algorithm adjustment, decoders.