# На правах рукописи

## Иконникова Надежда Сергеевна

ОПТИМИЗАЦИЯ ДВУМЕРНОЙ ОРТОГОНАЛЬНОЙ УПАКОВКИ НА ОСНОВЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

Направление 230100.68 «Информатика и вычислительная техника» магистерская программа «Автоматизированные системы обработки информации и управления»

АВТОРЕФЕРАТ

диссертации на соискание академической степени магистра

Нижний Новгород – 2013

Работа выполнена на кафедре «Информатика и системы управления» ФГБОУ ВПО «Нижегородский государственный технический университет им. Р.Е. Алексеева (НГТУ)»

**Научный руководитель:** кандидат технических наук,

доцент Тимофеева О.П.

**Рецензент:** кандидат технических наук,

доцент Копьева О.С.

Защита состоится «27» июня 2013 г. в 10 часов в аудитории 4403 в Нижегородском государственном техническом университете по адресу: 603600, г. Нижний Новгород, ГСП-41, ул. К.Минина, 24.

**ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ**

## Актуальность темы магистерской работы

Задачи упаковки и по сей день привлекают внимание ученых, исследователей и разработчиков. Причиной столь неугасающего интереса к данному классу задач является отсутствие простых и сравнительно быстрых способов их решения, а также разнообразие практических областей применения данной задачи. Точные методы для ее решения, такие, как методы ветвей и границ, локального поиска, не дают результата вследствие полиномиального роста времени обработки данных, то есть при большой размерности задачи неэффективны.

В данной работе предложен квазиоптимальный метод решения двумерной задачи упаковки, основанный на использовании эволюционного подхода.

Генетические алгоритмы представляют собой эвристические алгоритмы поиска, включающие механизмы случайного выбора, комбинирования и вариации искомых решений, которые хорошо зарекомендовали себя для решения задач оптимизации большой размерности, и позволяют получить качественное решение за приемлемое время. Отличительной особенностью адаптации генетического алгоритма к решению задачи двумерной упаковки является акцент на использование различных видов декодеров. Результат работы любого генетического алгоритма существенно зависит от выбранных параметров. Именно поэтому важно перед решением той или иной задачи выполнить настройку генетического алгоритма таким образом, чтобы получить наиболее оптимальное решение поставленной задачи. Цель данной работы – реализовать генетический алгоритм для решения двумерной задачи ортогональной упаковки, а затем в результате моделирования и анализа получить рекомендации по настройке параметров генетического алгоритма, которые гарантированно приведут к получению наиболее оптимального решения задачи.

## Постановка задачи

Существует множество вариаций задач двумерной упаковки. В рамках данной работы планируется реализовать алгоритм для решения двумерной задачи ортогональной упаковки. Исходными данными будет множество прямоугольников разной формы. Необходимо расположить их на полосе заданной ширины таким образом, чтобы достичь максимальной плотности упаковки. Данную задачу можно разделить на несколько подзадач, в зависимости от того, можно ли вращать заданные прямоугольники.

В результате проведенных исследований в области решения двумерной задачи ортогональной упаковки, в работе сформулированы следующие задачи:

- реализовать генетический алгоритм для решения двумерной задачи ортогональной упаковки,

- провести серию экспериментов для всех типов поставленных задач (основываясь на возможности вращения прямоугольников), учитывая форму прямоугольников для укладки,

- в результате моделирования и анализа получить рекомендации по настройке параметров генетического алгоритма, которые гарантированно приведут к получению наиболее оптимального решения задачи.

## Методы исследования

Для решения поставленной задачи в работе использовались понятия и методы решения задач оптимизации с помощью генетического алгоритма, а также эволюционного подхода к решению задач в целом, понятия применительно к задачам оптимизации упаковки.

Для практической апробации разработанных алгоритмов применено программное моделирование оптимизации двумерной ортогональной упаковки на основе генетического алгоритма, разработанное на языке C++ в интегрированной среде разработки *Microsoft Visual Studio 2010.*

## Научная новизна полученных результатов

Научная новизна заключается в разработке и реализации применения генетического алгоритма для решения двумерной задачи ортогональной упаковки, с целью достижения максимальной плотности упаковки.

Результаты были получены на основе проведенного исследования, определяющего зависимость результатов решения задачи от параметров генетического алгоритма.

Промежуточные результаты работы были представлены на Международной научно-технической конференции “Информационные системы и технологии” ИСТ-2012г., посвященной 95-летию Нижегородского политехнического института, а также в журнале Научно-технический вестник Поволжья. №6 2012г., в статье О.П. Тимофеевой, Э.С. Соколовой, Н.С. Иконниковой«Применение генетического алгоритма для оптимизации двумерной ортогональной упаковки».

## Практическая ценность

Практическая ценность полученных результатов обусловлена наличием разработанной и реализованной программной системы, позволяющей пользователю оптимизировать процесс упаковки или раскроя в двухмерном пространстве наиболее быстрым способом, для нахождения графического представления порядка укладки объектов с максимальной плотностью упаковки.

Значимым результатом является получение оптимальных настроек алгоритма применительно к различным условиям задачи. Следование полученным указаниям приведет к получению наилучшего решения задачи.

Результаты реализации методов и алгоритмов были апробированы на Международной научно-технической конференции “Информационные системы и технологии” ИСТ-2012г., посвященной 95-летию Нижегородского политехнического института, а также в журнале Научно-технический вестник Поволжья. №6 2012г., в статье О.П. Тимофеевой, Э.С. Соколовой, Н.С. Иконниковой«Применение генетического алгоритма для оптимизации двумерной ортогональной упаковки».

## Основные положения, выносимые на защиту

* Алгоритм решения двумерной задачи ортогональной упаковки на основе генетического алгоритма.
* Исследование и анализ результатов зависимости решения поставленной задачи от настроек генетического алгоритма.
* Программная реализация разработанного алгоритма.

**Структура и объем работы**

Магистерская работа состоит из введения, шести глав, заключения, библиографического списка и приложений. Общий объём работы составляет 70 страниц текста, содержащего 30 рисунков и 4 таблицы. Список литературы содержит 15 наименований.

**ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ**

**Во введении** обосновывается актуальность темы диссертационной работы, формируются цели и направления исследований, выносимых на защиту, определена научная новизна и практическая ценность полученных результатов, определяются выносимые на защиту положения.

**В первой главе** описана математическая постановка задачи. Ее можно представить следующим образом.

Даны *n* прямоугольников , для которых заданы , , *isOrient* – ширина, высота и возможность вращения *i*-го прямоугольника соответственно, задана – ширина полосы упаковки. Требуется найти ортогональную упаковку (прямоугольники располагаются параллельно сторонам полосы) без перекрытий данного набора прямоугольников, имеющую максимальную плотность упаковки.

Математическая постановка задачи имеет следующий вид:

Высота упаковки не ограничена, однако, для вычисления общей площади упаковки используется максимальная высота, которую заняли прямоугольники - *H*, а в качестве ширины упаковки используется *W* – ширина полосы. Кроме того, введены следующие дополнительные параметры задачи: *ai* – коэффициент, показывающий, входит ли *i*-й прямоугольник в число тех, которые определяют ширину упаковки (*ai*=[0,1]); *bi* – коэффициент, показывающий, входит ли *i*-й прямоугольник в число тех, которые определяют высоту упаковки (*bi*=[0,1]); ρϵ[0,1] - плотность упаковки (функция пригодности) - отношение площади, занимаемой прямоугольниками *S*уп, к общей площади упаковки *HW*.

Из основной постановки задачи следует три типа задач, которые будут рассмотрены в работе. В задаче 1 не все прямоугольники можно вращать при упаковке (*isOrient=(true/false*)). В задаче 2 все прямоугольники можно вращать (*isOrient=true*). В задаче 3 прямоугольники вращать нельзя (*isOrient=false*).

**Во второй главе** описана история развития генетических алгоритмов, основные термины. Сделан краткий обзор существующих методов, позволяющий выявить преимущества генетических алгоритмов. Описаны шаги стандартного генетического алгоритма.

**В третьей главе** рассматриваются все этапы модифицированного генетического алгоритма для решения двумерной задачи ортогональной упаковки.

Рассмотрим, как происходит кодирование возможного решения задачи. В качестве структуры данных, несущей информацию об упаковке, будем использовать последовательность номеров прямоугольников, представляющих очередность их укладки , где *n* – количество прямоугольников для упаковки, *pi* – номер прямоугольника, укладывающегося *i*-м в очереди. Такую последовательность назовем **приоритетным списком**. Каждый приоритетный список представляет собой кодированное решение, где элемент *pi* хранит параметры прямоугольника: *pi = {wi, hi, isOrienti, xi, yi}*. Отметим также, что за координаты прямоугольника (*xi*, *yi*) будем принимать координаты его нижней левой точки.

Каждый приоритетный список в терминах генетического алгоритма представляет собой конкретную особь популяции (**хромосому**). Каждый элемент *pi* –**ген**, который и составляет с совокупности всю особь.

Таким образом, приоритетный список является косвенной схемой кодирования решения. Однако, в такой схеме решения параметры *xi* и *yi* изначально не известны. Их можно вычислить с помощью перехода от косвенной к прямой схеме кодирования решения.

**Декодер –** оператор, позволяющий перейти от косвенной (числовой) схемы кодирования решения задачи к прямой (графической) схеме.

Декодер создает карту раскроя по приоритетному списку. Таким образом, с помощью декодера можно получить координаты прямоугольников, а значит, получить графическое представление решения задачи и вычислить значение целевой функции.

В работе предлагается использовать два вида декодеров для дальнейшего сравнительного анализа работы каждого из них: Декодер «Нижний левый» и декодер «Усовершенствованный нижний левый».

Декодер «Нижний левый» (BL) представлен на рис.1. Шаг *i* (**): последовательно передвинуть прямоугольник *pi* из *PL*, начиная от верхнего правого угла упакованной площади полосы вниз настолько, насколько это возможно, а затем влево, снова вниз и т.д., до тех пор, пока его можно сдвинуть влево или вниз. Преимущество здесь имеет движение влево.

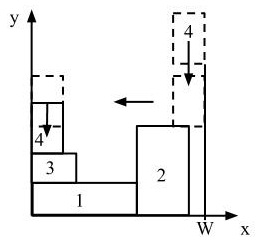


Рисунок 1. Декодер «Нижний левый»

Декодер «Усовершенствованный нижний левый» (IBL) представлен

на рис.2. Шаг *i* (**): последовательно передвинуть прямоугольник *pi* из *PL*, начиная от верхнего правого угла упакованной площади полосы вниз настолько, насколько это возможно, а затем влево, снова вниз и т.д., до тех пор, пока его можно сдвигать влево или вниз. При этом передвижение вниз имеет преимущество.

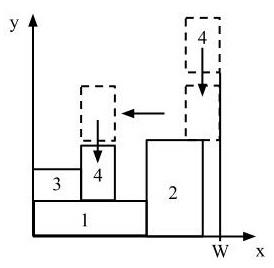
****

Рисунок 2. Декодер «Усовершенствованный нижний левый»

При решении поставленной задачи приходится учитывать ее специфику, а значит модифицировать стандартные генетические операторы.

**Функцию пригодности** можно вычислить по формуле:

,

где *n* – число прямоугольников, *Si*– площадь *i*-го прямоугольника, *H* – высота, которую занимают прямоугольники, *W* – ширина упаковки (ограничена условиями задачи).

В алгоритме используются три вида **селекции**: пропорциональная, ранговая, турнирная с возвращением. Каждая из них определяет *N* пар для скрещивания, где *N* – размер популяции.

Также будем проводить два вида **скрещивания**: двухточечное и одноточечное. Их приходится модифицировать, так как при использовании стандартных операторов могли бы получиться приоритетные списки, содержащие одинаковые номера прямоугольников, что противоречит условию задачи.

**Одноточечное скрещивание**. Пусть имеется два родителя (два приоритетных списка) *PL*1 и *PL*2. Случайным образом выберем место для разрыва между двумя позициями генов в обеих хромосомах. После этого перепишем в хромосому потомка первую часть первого родителя. А затем допишем недостающие гены в том порядке, в котором они встречаются у второго родителя. Аналогично получаем и второго потомка. Из них выбирается случайно один потомок, который и передается в качестве результата оператора скрещивания.

**Двухточечное скрещивание**. Пусть имеется два родителя (два приоритетных списка) *PL*1 и *PL*2. Случайным образом выберем два места для разрыва между двумя позициями генов в обеих хромосомах После этого перепишем в хромосому первого потомка выбранную в результате разрыва область. Остальные гены заполним в соответствии с тем, как они следуют в хромосоме второго родителя. Аналогично получается второй потомок. Из них выбирается случайно один потомок, который и передается в качестве результата оператора скрещивания.

**Мутация** проходит в два этапа. Сначала смена порядка прямоугольников в приоритетном списке, затем поворот прямоугольников. Поворот осуществляется в зависимости от параметра *isOrient* и типа решаемой задачи (возможности вращения прямоугольников). Вероятность мутации берем от двух до восьми процентов.

После проведения операций скрещивания и мутации имеем *N* родителей и *N* потомков, где *N* – число особей популяции. Параметр *Kgen,*, задаваемый пользователем, представляетсобой число лучших потомков (в процентах), которое перейдет в новое поколение. Чтобы размер популяции не менялся от итерации к итерации, дополним популяцию случайными особями из родительского поколения.

**В четвертой главе** описывается процесс разработки программы, реализующей генетический алгоритм, а именно: обзор среды разработки, основные идеи представления данных в программе, описание функций.

**В пятой главе** приведены результаты применения разработанного метода. Выполнена настройка параметров алгоритма для решения трех видов поставленных задач в зависимости о соотношения сторон прямоугольников. Было проведено тестирование описанного выше алгоритма для трёх типов задач на основе следующих параметров: тип декодера, тип селекции, тип кроссовера, *Kst* – коэффициент отношения сторон прямоугольников (отношение *wi* к *hi*), возможность вращения прямоугольников, *Kgen*. В качестве входных данных были взяты 100 прямоугольников, ширина полосы, равная 500, максимальное количество поколений, равное 25.

Ниже представлены графики зависимостей плотности упаковки от вышеперечисленных параметров для задачи третьего типа, когда лишь часть прямоугольников можно вращать.

Рисунок 3. Зависимость плотности упаковки от типа декодера для задач третьего типа

Рисунок 4. Зависимость плотности упаковки от типа селекции для задач третьего типа

Рисунок 5. Зависимость плотности упаковки от типа кроссовера для задач третьего типа

Рисунок 6. Зависимость плотности упаковки от коэффициента *Kgen* для задач третьего типа

Все результаты экспериментов для трех типов задач можно свести в одну таблицу (Таблица 1).

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Задача первого типа *(isOrient = false)*** | | |
|  |  |  |  |
| Тип декодера | *IBL* | *IBL* | *IBL* |
| Тип селекции | Пропорциональная | Ранговая | Турнирная |
| Тип кроссовера | Одноточечный | Двухточечный | Одноточечный |
| *Kgen* | 20 | 60 | 40 |
|  | **Задача второго типа *(isOrient = true)*** | | |
|  |  |  |  |
| Тип декодера | *IBL* | *IBL* | *IBL* |
| Тип селекции | Турнирная | Пропорциональная | Турнирная |
| Тип кроссовера | Двухточечный | Двухточечный | Двухточечный |
| *Kgen* | 20 | 20 | 40 |
|  | **Задача третьего типа *(isOrient = true/false)*** | | |
|  |  |  |  |
| Тип декодера | *IBL* | *IBL* | *IBL* |
| Тип селекции | Турнирная | Пропорциональная | Турнирная |
| Тип кроссовера | Двухточечный | Одноточечный | Одноточечный |
| *Kgen* | 60 | 60 | 40 |

Таблица 1. Сводная таблица по настройке генетического алгоритма для решения двумерной задачи ортогональной упаковки.

Из всего вышеперечисленного можно сделать следующие выводы:

- для всех видов задач наиболее подходящий вид декодера – **усовершенствованный нижний-левый** (*IBL*).

- **турнирная селекция** лучше всего работает с теми задачами, в которых ширина прямоугольника больше или приблизительно равно его высоте. В случае прямоугольников вертикальной ориентации лучше выбирать **пропорциональную селекцию**. Это можно объяснить тем, что для вертикально-ориентированных прямоугольников важнее, чтобы только самые лучшие особи текущего поколения привнесли свои свойства в новое поколение. В остальных двух случаях ( и ) важнее не остановиться в точке локального максимума и выбирать лучшего только из *T* (размер турнира) случайных особей.

- **Двухточечный кроссовер** однозначно лучше всего использовать для задач второго типа, когда все прямоугольники можно вращать.

- Конкретной зависимости функции пригодности от коэффициента формирования нового поколения *Kgen* не выявлено, однако для задач третьего типа характерны намного большие значения этого коэффициента, нежели для других типов задач.

**В шестой главе** приведена инструкция пользователя.

**В заключении** формулируются основные результаты, полученные в магистерской работе.

**СПИСОК ПУБЛИКАЦИЙ РАБОТ ПО ТЕМЕ ДИССЕРТАЦИИ**

1. Тимофеева О.П, Соколова Э.С, Иконникова Н.С. Применение генетического алгоритма для оптимизации двумерной ортогональной упаковки // Научно-технический вестник Поволжья. - 2012г. – №6. – С. 401-404
2. Тимофеева О.П, Иконникова Н.С. Двумерная задача об ортогональной упаковке. Реализация на основе генетического алгоритма с использованием различных видов декодеров // Материалы XVIII Международной научно-технической конференции «Информационные системы и технологии (ИСТ-2012)».– Н. Новгород, 2012.
3. Тимофеева О.П, Иконникова Н.С. Применение генетических алгоритмов в решении задачи одномерной упаковки // Сборник трудов Х международной молодежной научно-технической конференции «Будущее технической науки». – Н.Новгород, 2011.