

## **АННОТАЦИЯ**

Дипломная работа: 15 с., 0 рис., 1 табл., 5 источников.

## ОГЛАВЛЕНИЕ

	Стр.
<b>АННОТАЦИЯ .....</b>	<b>5</b>
<b>ВВЕДЕНИЕ .....</b>	<b>7</b>
 <b>ГЛАВА 1 Теоретическая часть</b>	 <b>8</b>
1.1 Задача о многомерном рюкзаке .....	8
1.2 Генетические алгоритмы .....	9
1.2.1 Этапы работы алгоритма .....	10
1.2.2 Выбор этапов .....	11
1.2.3 Дополнения алгоритма .....	12
 <b>ГЛАВА 2 Вторая глава</b>	 <b>13</b>
2.1 Результаты реализации алгоритма .....	13
<b>СПИСОК ЛИТЕРАТУРЫ .....</b>	<b>14</b>
<b>ПРИЛОЖЕНИЕ А КОД ПРОГРАММНОГО ПРОДУКТА .....</b>	<b>15</b>

## **ВВЕДЕНИЕ**

## ГЛАВА 1

### Теоретическая часть

#### 1.1 Задача о многомерном рюкзаке

В данной работе рассматривается задача о многомерном рюкзаке (Multidimensional 0-1 knapsack problem, МКР). Эта задача является модификацией классической задачи о рюкзаке, поставленной в 19 веке Джорджем Мэттьюсоном. (см [2]) Данный же вариант задачи впервые был предложен Клиффордом Петерсеном в 1967 году.(см [3])

Постановка задач такова

Пусть существует  $N$  предметов, каждый из которых имеет стоимость  $c_i$  и размеры  $s_{ij}$ , где  $i \in 1, 2, \dots, N, j \in 1, 2, \dots, M$ . Пусть также существует рюкзак с ограничениями по вместимости по измерениям  $r_j$ . Требуется максимизировать сумму

$$\sum_{i=1}^N c_i x_i$$

где  $x_i \in \{0, 1\}$  при условии

$$\sum_{i=1}^N s_{ij} x_i < r_j \forall j \in \{1, 2, \dots, M\} \quad (1.1)$$

И стандартная задача, и её модификация являются NP-полными задачами. Вычислительная сложность задачи такого рода при переборном решении для  $N$  предметов -

$$o(2^N)$$

, что, вкупе с NP-сложностью, делает алгоритмическое решение та-

кой задачи неэффективным для больших  $N$ . Однако такие задачи могут быть решены эвристическими алгоритмами, то есть алгоритмами, для которых их корректность строго не доказана.

## 1.2 Генетические алгоритмы

Генетические алгоритмы являются семейством в множестве эвристических алгоритмов. Впервые такой алгоритм был предложен А. Фразером. (см [4]) Алгоритм является итеративным. Генетический алгоритм моделирует естественные процессы эволюции популяции, а именно - мутацию и скрещивание. Решение задачи с помощью такого алгоритма требует нескольких предварительных этапов:

- Выбор кодирования генотипа.

На этом этапе нужно выбрать способ кодирования генотипа, который будет эффективен для данной задачи. Такой генотип должен однозначно моделировать сущность, рассматриваемую в задаче.

- Выбор начального приближения.

Для запуска итерационного процесса требуется создать начальное множество - пул генотипов.

- Выбор мутации.

На каждой итерации алгоритма часть пула генотипов подвергнется мутациям, то есть определенным образом изменяются их составляющие.

- Выбор механизма скрещивания (кроссинговера).

После мутации происходит создание новых генотипов из частей старых с сохранением признаков родителя. Алгоритм скрещи-

вания позволяет получить из двух родительских генотипов два различных дочерних генотипа.

- Выбор функции оценки(фитнесс-функции).

Такая функция позволяет оценивать генотипы с точки зрения их близости к оптимальному решению и отбирать из них лучшие на каждой итерации.

### 1.2.1 Этапы работы алгоритма

- Создается пул генотипов с использованием заданного алгоритма начального приближения
- Запускается итерационный процесс

Случайным образом выбирается часть пула, которая подвергнется мутации

Выбранная часть пула генотипов мутируется с использованием заданного алгоритма мутации

Мутировавшие генотипы замещают собой исходные в пуле, немутировавшие остаются без изменений

Из пула генотипов выбираются пары для скрещивания

Производится скрещивание с использованием заданного алгоритма

С использованием заданной функции оценки из результатов скрещивания выбираются лучшие

Если выполнено условие останова - например, достигнут предел числа итераций или известный максимум, то итерационный процесс завершается, в противном случае начинается следующая итерация.

- Результат итерационного процесса отдается пользователю

### 1.2.2 Выбор этапов

Наиболее естественным кодированием отдельного решения задачи о рюкзаке в генотип является бинарная последовательность длины  $N$ , состоящая из нулей и единиц. Каждый  $i$ -й элемент такой последовательности является индикатором вхождения  $i$ -го предмета в текущее решение. Такая модель требует наличия проверки коефективности генотипа - соблюдения условия [1.1](#)

Для генерации начального приближения был использован жадный алгоритм. Сначала создается генотип из единиц, соответствующий конфигурации рюкзака, в который положены все предметы. Затем в случайном порядке единицы заменяются на нули, пока полученная конфигурация не будет удовлетворять условию корректности. После этого полученный генотип мутируется с помощью текущей мутации до заполнения пула решений.

В ходе работы было реализовано несколько алгоритмов мутации и скрещивания с целью сравнения их эффективности. Были реализованы следующие алгоритмы мутации:

- Мутация в одной позиции, при которой заменяется значение в одной случайно выбранной точке генотипа.
- Инверсионная мутация, при которой половина генотипа заменяется на противоположные значения.

Были реализованы следующие алгоритмы скрещивания:

- Скрещивание по 1 точке, при котором выбирается произвольная точка в последовательности генотипа, значения до точки берутся

от первого генотипа, после - от второго.

- Скрещивание по двум точкам, при котором выбираются две различные произвольные точки, значения внутри интервала и в самих точках берутся из первого генотипа, вне интервала - из второго.
- Побитовое скрещивание, при котором значения на нечетных позициях берутся из первого генотипа, на четных - из второго.

В качестве функции оценки используется стоимость всех предметов, содержащихся в рюкзаке, соответствующем конфигурации.

### 1.2.3 Дополнения алгоритма

В связи со спецификой задачи в алгоритм были внесены дополнения.

Были введены проверки генотипов на корректность после мутации и скрещивания. Если генотип не удовлетворяет условию корректности, то значения начиная с первой позиции начинают зануляться до достижения генотипом корректности.

Были введены дополнительные пулы лучших конфигураций за время работы алгоритма. Такие пулы решают одновременно несколько задач

- Недопущение сильного ухудшения результатов решения вследствие случайных мутаций.
- Возможность сравнения решений после окончания работы алгоритма
- Возможность сохранения результатов при перезапуске алгоритма с другим начальным приближением. Такой перезапуск оправдан при получении генотипа - локального максимума.



## ГЛАВА 2

### Вторая глава

#### 2.1 Результаты реализации алгоритма

Для оценки эффективности работы алгоритма было проведено его тестирование на различных наборах тестов. Первый набор тестов взят из книги Петерсена(см. [3]) и содержит 7 задач. Условия этих задач(см таблицу 2.1) позволяют проверить эффективность простейшей версии генетического алгоритма без модификаций и оценить её эффективность.

Таблица 2.1 --- Параметры первого набора тестов

№ задачи	Размерность	Количество предметов
1	6	10
2	10	10
3	15	10
4	20	10
5	28	10
6	39	5
7	50	5

Второй набор тестов взят из статьи Чу (см [5]) и содержит в себе 30 задач с одинаковыми параметрами: размерность рюкзака равна 5, рассматривается 100 различных предметов.

## СПИСОК ЛИТЕРАТУРЫ

1. Кнорринг, Г.М. Справочная книга для проектирования электрического освещения. / Г. М. Кнорринг, Ю. Б. Оболенцев, Р. И. Берим, В. М. Крючков; Под ред. Г.Б. Кнорринга. Л.: Энергия, 1976. 384с.
2. Mathews, G. B. On the partition of numbers / G.B. Mathews / / Proceedings of the London Mathematical Society. 28: С. 486–490.
3. C.C.Petersen "Computational experience / with variants of the Balas algorithm applied to the selection of R&D projects"Management Science 13(9) (1967) 736-750.
4. Fraser Alex. Computer Models in Genetics. — New York: McGraw-Hill, 1970. — ISBN 0-07-021904-4.
5. P.C.Chu and J.E.Beasley "A genetic algorithm for the multidimensional knapsack problem-/ Journal of Heuristics, vol. 4, 1998, С. 63-86.

ПРИЛОЖЕНИЕ А

**КОД ПРОГРАММНОГО ПРОДУКТА**

```
namespace Hellenist {  
    public enum Case {  
        Nominative = 1,  
        Vocative = 2,  
        Accusative = 3,  
        Genitive = 4,  
        Dative = 5  
    }  
}
```