

## ЭФФЕКТИВНЫЕ АЛГОРИТМЫ ДЛЯ РЕШЕНИЯ ЗАДАЧ УПАКОВКИ ПРОДУКЦИИ

**Калыбаев Айбек Мнажатдинович**

*Магистрант Ташкентского университета информационных технологий имени  
Мухаммада аль-Хорезми*

**Аннотация:** В данной статье рассматривается проблема эффективной упаковки продукции в контейнеры или на палеты. Представлен обзор существующих математических моделей и алгоритмов, используемых для решения этой проблемы. Особое внимание уделено алгоритмам на основе метаэвристик, таких как генетические алгоритмы и муравьиные алгоритмы, которые показали высокую эффективность в решении задач упаковки.

**Введение:** Проблема эффективной упаковки продукции является актуальной для многих производственных и логистических компаний. Эффективная упаковка продукции позволяет экономить место и ресурсы, снижать затраты на транспортировку и хранение продукции. В данной статье рассматриваются математические модели и алгоритмы, которые используются для решения задач упаковки.

**Основная часть:** Существует несколько основных подходов к решению задачи упаковки продукции. Один из них - это применение точных методов, таких как методы динамического программирования и методы целочисленного линейного программирования. Однако, эти методы могут быть неэффективными при работе с большими объемами данных и сложными ограничениями.

Другой подход - это использование эвристических алгоритмов, которые могут быстро находить приближенные решения задачи упаковки. Эти алгоритмы, как правило, основаны на метаэвристических методах, таких как генетические алгоритмы, муравьиные алгоритмы, симуляция отжига и другие.

Генетические алгоритмы - это алгоритмы оптимизации, которые используют принципы естественного отбора и генетической мутации для нахождения оптимальных решений. Они основываются на моделировании эволюции популяции, где каждый индивидум в популяции представляет собой потенциальное решение задачи.

Процесс работы генетического алгоритма можно разделить на несколько этапов:

1. Создание начальной популяции - это случайный набор индивидуумов, который образует первоначальную популяцию.
2. Выбор лучших индивидуумов - это процесс выбора наиболее приспособленных индивидуумов в популяции на основе критерия приспособленности.



Критерий приспособленности может быть задан в виде функции цели, которую необходимо минимизировать или максимизировать.

3. Оператор скрещивания - это процесс создания новых индивидуумов путем комбинации генов (частей решения) двух родительских индивидуумов. Это делается с помощью различных методов скрещивания, таких как одноточечное, двухточечное, равномерное и другие.

4. Оператор мутации - это случайное изменение генов индивидуумов с целью увеличения разнообразия популяции и избежания застревания в локальных оптимумах. Мутация может произойти с небольшой вероятностью для каждого гена.

5. Оценка приспособленности - это процесс оценки приспособленности каждого индивидуума в популяции с помощью критерия приспособленности. На основе этой оценки индивидуумы классифицируются на лучшие и худшие, и выбираются лучшие для дальнейшего использования.

6. Повторение цикла - шаги 2-5 повторяются до тех пор, пока не будет достигнуто определенное условие остановки, например, достижение максимального количества итераций или достижение определенного уровня качества решения.

Пример генетического алгоритма для решения задачи о рюкзаке:

Предположим, что у нас есть рюкзак, который может вместить определенный вес и у нас есть набор предметов, каждый из которых имеет свой вес и стоимость. Задача состоит в том, чтобы выбрать набор предметов с максимальной общей стоимостью, которые могут поместиться в рюкзак.

- Создание начальной популяции: создаем случайную популяцию индивидуумов, где каждый индивидуум представляет собой набор предметов.

- Оценка приспособленности: оцениваем приспособленность каждого индивидуума в популяции с помощью функции цели, которая вычисляет суммарную стоимость выбранных предметов, при условии, что их вес не превышает вместимость рюкзака.

- Выбор лучших индивидуумов: выбираем лучших индивидуумов на основе их приспособленности. Для этого можно использовать различные методы, такие как турнирный отбор или рулеточный отбор.

- Оператор скрещивания: выбираем двух родительских индивидуумов и создаем нового потомка, комбинируя их гены (предметы). Можно использовать одноточечное или двухточечное скрещивание.

- Оператор мутации: случайно изменяем гены потомка, чтобы добавить разнообразия в популяцию. Можно изменять вес или стоимость случайного предмета или заменять его на другой случайный предмет.

- Повторение цикла: шаги 2-5 повторяются до достижения определенного условия остановки.

Формула для вычисления приспособленности в этом примере может быть записана следующим образом:



fitness = sum(values \* mask)

где values - массив стоимостей предметов, mask - бинарный массив размером с количество предметов, который указывает, какие предметы выбраны для помещения в рюкзак. Если предмет выбран, то соответствующее значение в mask равно 1, в противном случае - 0.

Муравьиный алгоритм начинается с создания начальной популяции решений (муравьев), которые последовательно посещают узлы графа (города). Каждый муравей выбирает следующий узел на основе вероятностной функции, которая зависит от феромона на дуге и эвристической информации, такой как расстояние до узла. После того, как муравей завершает маршрут, феромон на всех посещенных дугах обновляется, основываясь на эффективности маршрута.

#### 1. Создание муравьев.

В разных задачах начальное размещение муравьев может быть разным. Это зависит от условий задачи. Ещё на этом шаге происходит первоначальное размещение небольшого количества феромона для того, чтобы на первом шаге вероятность перехода в соседнюю вершину была не нулевой.

#### 2. Поиск решений.

Формула для вычисления вероятности перехода муравья из вершины  $i$  в  $j$ :

$$P_{ij,k}(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in J_{i,k}} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta}$$

где  $\tau_{ij}(t)$  – количество феромона между вершинами  $i$  и  $j$ ,  $\eta_{ij}$  – расстояние между этими вершинами.  $\alpha$ ,  $\beta$  – константные параметры. Их необходимо подбирать опытным путём, их значение должно быть такое, чтобы алгоритм не был слишком жадным и не застревал в локальных минимумах.

Чем ближе к нулю параметр  $\beta$ , тем меньше муравьи в выборе пути будут руководствоваться расстоянием между вершинами и будут ориентироваться только на феромон. С увеличением  $\beta$  значение близости растёт. Параметр  $\alpha$  действует так же, но для уровня феромона.

Верхняя часть формулы описывает желание муравья перейти из вершины  $i$  в вершину  $j$ . Оно пропорционально близости вершины и уровню феромона на пути к ней.

Таким образом, вероятность перехода из вершины  $i$  в вершину  $j$  равняется желанию перейти в неё, делённому на сумму желаний перейти из вершины  $i$  ко всем доступным вершинам, которые ещё не были посещены. Сумма всех вероятностей равна 1.

Разложив все вероятности на числовой прямой от 0 до 1, можно сгенерировать случайное вещественное число в этом интервале. Результат покажет, в какую вершину перейдёт муравей.



### 3. Обновление феромона.

Формула для пересчёта уровня феромона на каждой итерации алгоритма:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{k \in \{used(i,j)\}} \frac{Q}{L_k(t)}$$

где  $\rho$  – скорость испарения,  $t$  – номер итерации,  $L_k(t)$  – цена текущего решения для  $k$ -ого муравья, а  $Q$  – параметр, имеющий значение порядка цены оптимального решения, то есть  $Q/L_k(t)$  – феромон, откладываемый  $k$ -ым муравьём, использующим ребро  $(i, j)$ .

Таким образом, количество феромона на ребре между  $i$  и  $j$  на новой итерации равно количеству феромона на старой итерации, умноженное на коэффициент испарения (феромон постоянно испаряется), и к полученному результату добавляется сумма всех новых порций феромона, который отложили все муравьи на этом участке. Добавка феромона, которую делает муравей, проходя по ребру, равна константе  $Q$ , делённой на длину маршрута  $L$ , пройденную муравьём  $k$ , при условии, что это ребро попало в маршрут муравья.

Кроме того, существуют и другие алгоритмы, такие как методы мультиагентного моделирования и генеративные модели, которые также могут быть применены к задачам упаковки продукции.

Заключение:

В данной статье были рассмотрены математические модели и алгоритмы, которые используются для решения задач упаковки продукции. Было показано, что использование метаэвристических алгоритмов, таких как генетические алгоритмы, муравьиные алгоритмы и симуляция отжига, может быть эффективным способом нахождения оптимальных решений. Кроме того, существуют и другие методы оптимизации, которые могут быть применены к этой проблеме. Дальнейшие исследования в этой области могут привести к разработке новых математических моделей и алгоритмов, которые могут помочь производственным и логистическим компаниям сократить расходы на транспортировку и хранение продукции.

### СПИСОК ЛИТЕРАТУРЫ:

1. Bortfeldt, A., & Gehring, H. (2014). Container loading and related problems: A classification overview. *European Journal of Operational Research*, 233(2), 289-311.
2. Wang, Y., He, Y., Wu, H., & Guo, Z. (2019). A hybrid ant colony algorithm for three-dimensional container loading problem with practical constraints. *International Journal of Production Research*, 57(7), 2175-2195.
3. Lodi, A., & Martello, S. (2003). Three-dimensional packing problems. *INFORMS Journal on Computing*, 15(2), 204-217.



4. Eleyan, D., & Rahim, M. A. (2016). A simulated annealing algorithm for solving the container loading problem with weight and stability considerations. *Journal of Cleaner Production*, 112, 1688-1696.
5. Agatz, N. A., Campbell, A. M., & Fleischmann, B. (2008). Exact algorithms for the three-dimensional bin packing problem with stochastic item sizes. *European Journal of Operational Research*, 187(3), 851-872.