

УДК 519.8

UDC 519.8

05.00.00 Технические науки

Technical sciences

**АЛГОРИТМ МУРАВЬИНОЙ КОЛОНИИ ДЛЯ РЕШЕНИЯ ЗАДАЧ ОПТИМАЛЬНОГО РАЗМЕЩЕНИЯ РАСПРЕДЕЛИТЕЛЬНЫХ ЦЕНТРОВ РОЗНИЧНОЙ ТОРГОВОЙ СЕТИ**

**ANT SYSTEM ALGORITHM FOR SOLVING OPTIMAL PLACEMENT OF DISTRIBUTION CENTERS IN RETAIL CHAINS**

Бедакова Наталья Васильевна  
аспирант  
ФГБОУ ВПО «Кубанский государственный технологический университет», г. Краснодар, Россия

Bedakova Natalia Vasilievna  
postgraduate student  
FSEI HPE «Kuban State Technological University», Krasnodar, Russia

В статье изложена природа и идея муравьиной колонии. Представлен алгоритм муравьиной колонии как метод решения трудных комбинаторных оптимизационных задач. Описана схема алгоритма муравьиной колонии для решения трудных комбинаторных оптимизационных задач для задачи о  $p$ -медиане. Предложен алгоритм муравьиной колонии применительно к задаче оптимального размещения распределительных центров для задачи о  $p$ -медиане

The article describes the nature and the idea of an ant colony. We have presented an ant colony algorithm as a method of solving difficult combinatorial optimization problems. The work describes a scheme of the ant colony algorithm for solving difficult combinatorial optimization problems for the problem of  $p$ -median. The ant colony algorithm was proposed in relation to the problem of optimal facility location problem for the  $p$ -median

Ключевые слова: АЛГОРИТМ МУРАВЬИНОЙ КОЛОНИИ, ОПТИМИЗАЦИЯ, РАЗМЕЩЕНИЕ ПРЕДПРИЯТИЙ, РАСПРЕДЕЛИТЕЛЬНЫЙ ЦЕНТР, КОМБИНАТОРНЫЕ ЗАДАЧИ, ЗАДАЧА О  $P$ -МЕДИАНЕ

Keywords: ANT SYSTEM, OPTIMIZATION, FACILITY LOCATION, DISTRIBUTION CENTERS, COMBINATORIAL PROBLEMS, PROBLEM OF THE P-MEDIAN

**Введение**

В современных условиях для успешного развития производства и сферы услуг необходимо решать задачи оптимального размещения распределительных центров (складов, филиалов, офисов и т.п.). В управлении крупными предприятиями большое значение стали придавать физическому распределению предприятий (логистике). Причины возросшего интереса связаны с ростом стоимости транспортных услуг.

Целью транспортной системы любого предприятия является обеспечение доставки товаров клиентам (потребителям) с минимальными затратами в минимальные сроки. Для этого необходимо решить задачу транспортно-складской системы, которая заключается в определении: оптимального количества предприятий; мест их расположения; характеристик каждого предприятия; оптимальной системы перевозок

товаров. В большинстве случаев такие задачи являются сложными и требуют применения моделей исследования операций и оптимизации.

Вычислительная сложность, а также большая размерность задач размещения предприятий, как правило, не позволяют находить оптимальное решение за приемлемое время. В связи с этим особое значение приобрела разработка метода получения приближенных решений. В последние годы активно развиваются алгоритмы, основанные на процедуре локального поиска [2, 3]. Значительный интерес проявляется к алгоритмам, идеи которых заимствованы у живой природы или физических процессов. К таким методам можно отнести генетические алгоритмы, алгоритмы имитации отжига, нейронные сети, а также алгоритмы муравьиной колонии.

Целью данной статьи является исследование алгоритма муравьиной колонии для решения задачи структурно-топологической оптимизации размещения распределительных центров крупной розничной торговой компании. Исследование выполнено в рамках научно-исследовательского проекта РГНФ («Управление эффективностью пространственно распределённых промышленных предприятий с учётом фактора надёжности»), проект № 14-02-00334а.

Алгоритмы муравьиной колонии впервые были предложены Dorigo, Maniczzo и Colomi [1] как метод решения трудных комбинаторных оптимизационных задач, таких как задача коммивояжера и квадратичная задача о назначениях. С тех пор алгоритмы муравьиной колонии активно развивались и стали применяться к другим задачам дискретной оптимизации. Появились алгоритмы для задач маршрутизации, задачи о раскраске графа, задач раскроя и упаковки, простейшей задачи размещения, задачи о  $p$ -медиане и ряда других задач.

### Общая схема муравьиной колонии

Муравей – это программный агент, который является членом большой колонии и используется для решения какой-либо проблемы. Муравей снабжается набором простых правил, которые позволяют ему выбрать путь на графе.

Введем некоторые обозначения. Пусть дискретная задача оптимизации – это задача определения на множестве  $Z = \{z_1, z_2, \dots, z_n\}$ ,  $D \subseteq 2^Z$  – множество допустимых решений задачи. Функция  $f: D \rightarrow R$  – целевая функция задачи. Требуется найти такое решение задачи  $s^*$ , что  $s^* \in D$  и  $f(s^*) \leq f(s), \forall s \in D$ . Некоторое подмножество  $s$  множества  $Z$  является частичным решением задачи, причем если  $s \in D$ , то  $s$  представляет собой допустимое решение.

На каждой итерации алгоритма муравьиной колонии конечное число искусственных муравьев или агентов ищут допустимое решение дискретной оптимизационной задачи. Решением является путь минимальной стоимости по некоторым состояниям задачи, представляющим собой частичные решения. Муравей строит решение, начиная с некоторого начального состояния в зависимости от специфики решаемой задачи. В результате, после того, как искусственный муравей построил решение, он получает некоторую информацию о задаче, которая обрабатывается и используется агентами в дальнейшем. Эта информация и является аналогом феромона живых муравьев.

Искусственный муравей строит решение, «двигаясь» по состояниям задачи согласно некоторому вероятностному правилу. После завершения построения решения (или в течение построения) агент оценивает решение и изменяет значение уровня феромона на компонентах, используемых в данном решении. Таким образом, искусственный муравей – это некоторый жадный алгоритм, который итеративно шаг за шагом строит решение задачи. На каждом шаге  $r$  муравей  $l$  определяет множество возможных

направлений  $A_i^r(\varphi)$  из текущего состояния  $\varphi$  и выбирает одно из них с некоторой вероятностью. Для муравья  $l$  вероятность  $p_{\varphi\psi}^l$  перехода из состояния  $\varphi$  в состояние  $\psi$  зависит от комбинации значений привлекательности перехода и уровня феромона [3].

Привлекательность перехода  $\eta_{\varphi\psi}$  из состояния  $\varphi$  в состояние  $\psi$  представляет собой число, вычисляемое при помощи некоторой эвристики. Значение привлекательности никак не зависит от того, какие решения были найдены на предыдущих итерациях. Данная величина показывает априорную (независимую) желательность перехода из состояния  $\varphi$  в состояние  $\psi$ .

Уровень феромона  $\tau_{\varphi\psi}$  представляет собой положительное число, показывающее насколько часто муравьи двигались из состояния  $\varphi$  в состояние  $\psi$  на предыдущих итерациях, а также решениям какого качества соответствует данный переход. Таким образом, указанная величина отражает апостериорную (выведенную из опыта) желательность перехода.

Обычно уровень феромона переопределяется в тот момент, когда все искусственные муравьи закончили построение решения на данной итерации. Увеличение или уменьшение уровня феромона для соответствующих переходов между состояниями зависит от того, к решениям какого качества они относятся.

Алгоритм оптимизации муравьиной колонии может быть успешно применен для решения сложных комплексных задач оптимизации. Цель решения сложных комплексных задач оптимизации – поиск и определение наиболее подходящего решения для оптимизации (нахождение минимума или максимума) целевой функции из дискретного множества возможных решений. Пример решения подобной задачи – это задача календарного планирования, квадратичная задача о назначениях, задача маршрутизации транспорта, различных сетей (GPS, ГЛОНАСС, телефонные и

компьютерные и т.п.), распределение ресурсов и работ. Перечисленные задачи возникают в инженерии, производстве, бизнесе и во многих других областях. Алгоритм муравьиных колоний может давать результаты, даже лучше чем при использовании генетических алгоритмов и нейронных сетей.

### **Алгоритм муравьиной колонии для задачи размещения распределительных центров розничной торговой сети (для задачи о $p$ -медиане)**

Задача о  $p$ -медиане - это задача о размещении заданного числа (скажем,  $p$ ) пунктов обслуживания, при которых сумма кратчайших расстояний от вершин графа до ближайших пунктов принимает минимально возможное значение.

Дано:

- множество пунктов возможного размещения;
- список розничных торговых точек компании (расположение магазинов);
- затраты на транспортировку (стоимость прохождения 1 км пути);
- необходимое количество распределительных центров для открытия;
- расположение ближайшего распределительного центра компании.

Цель: открыть склад и прикрепить к нему магазины так, чтобы затраты были минимальными.

Задача целочисленного линейного программирования:

$$F(z, X) = \sum_{i \in I} T_i + \sum_{i \in I} \sum_{j \in J} t_{ij} x_{ij} \rightarrow \min \quad (1)$$

$$\left\{ \begin{array}{l} \sum_{i \in I} x_{ij} = 1, j \in J \\ \sum_{i \in I} z_i = p \\ z_i \geq x_{ij}, i \in I, j \in J \\ z_i \in \{0, 1\}, x_{ij} \geq 0, i \in I, j \in J \end{array} \right.$$

Обозначения:

$m$  – число пунктов возможного размещения распределительного центра;

$i$  – номер пункта возможного размещения распределительного центра;

$n$  – число торговых точек;

$j$  – номер торговой точки;

$t_{ij}$  – затраты на удовлетворение спроса  $j$  магазина  $i$  распределительного центра (транспортные затраты);

$x_{ij}$  – затраты на удовлетворение спроса  $j$  клиента;

$z_i = \begin{cases} 1, \text{ если предприятие открыто в } i \\ 0, \text{ в ином случае} \end{cases};$

$T_i$  – транспортные затраты на доставку товаров в распределительный центр с ближайшего распределительного центра;

$p$  – количество распределительных центров, которые нужно открыть.

В крупной розничной торговой компании, такой как АО «Тандер», товар доставляется из одного распределительного центра в другой распределительный центр, и доставка в конечный пункт будет осуществляться с ближайшего распределительного центра. Необходимо учитывать транспортные затраты на доставку товара на распределительный центр.

Для описания алгоритма воспользуемся комбинаторной постановкой задачи, которая имеет следующий вид [4].

Дана не отрицательная  $m \times n$  матрица  $T = (t_{ij})$  с множеством индексов строк  $I$  и множеством индексов столбцов  $J$ , а также натуральное число  $p \leq m$ . Для всякого непустого подмножества  $I' \subseteq I$  положим

$$F(I') = \sum_{j \in J} \min_{i \in I'} t_{ij}. \quad (2)$$

Задача состоит в отыскании такого множества  $I'$  мощности  $p$ , что значение  $F(I')$  минимально, то есть

$$\min_{I' \subseteq I} \{ F(I') : |I'| = p \}. \quad (3)$$

Решение  $s$  задачи будем называть булев вектор  $z$  размерности  $m$  такой, что  $z_i = 1$ , если  $i \in I_s$ , и 0 – в противном случае, где  $I_s$  – множество открытых в решении  $s$  распределительных центров.

Введем вектор  $\alpha^k = (\alpha_i^k)$ ,  $i \in I$ , с положительными координатами. Величину  $\alpha_i^k$  будем называть уровнем феромона для  $i$ -го распределительного центра на итерации  $k$  алгоритма муравьиной колонии,  $i \in I$ .

Обозначим через  $\bar{s}^k$  лучшее найденное решение до начала итерации  $k$ , тогда  $\bar{z}^k$  и  $\bar{f}^k$  – булев вектор и значение целевой функции, соответствующие рекордному решению  $\bar{s}^k$ . Пусть  $s^k$  – лучшее решение по значению целевой функции, найденное на итерации  $k$ , тогда  $z^k$  и  $f^k$  – булев вектор и значение целевой функции (рекорд на итерации  $k$ ), соответственно. Параметр  $\alpha_{\min}$  – вещественное положительное число, задающее минимальное возможное значение уровня феромона  $\alpha_i^k$  для всех  $i \in I$ .

0. Определяем начальный вектор феромона  $\alpha^1$ ; рекорд  $\bar{f}^1 := \infty$ ,

Итерация  $k$ ,  $k \geq 1$ .

1. Строим  $L$  допустимых решений одним из алгоритмов искусственного муравья (алгоритм искусственного муравья представляет собой вероятностную модификацию жадного алгоритма<sup>1</sup>, описан ниже).

Алгоритм искусственного муравья определяет привлекательные возможные расположения распределительных центров.

2. Среди этих решений выбираем  $l$  лучших по целевой функции с помощью локального поиска.

Выбирается наилучшее расположение распределительного центра из возможных привлекательных расположений.

3. Находим значения  $\alpha_i^{k+1}$ ,  $i \in I$ .

Определяем уровень феромона

4. Если  $f^k < \bar{f}^k$ , то  $\bar{f}^{k+1} = f^k$ ;  $\bar{s}^{k+1} = s^k$ ;

для ненулевых компонент  $z^k$  полагаем  $\alpha_i^{k+1} = \alpha_{min}$ .

Иначе  $\bar{f}^{k+1} = \bar{f}^k$ ;  $\bar{s}^{k+1} = \bar{s}^k$ .

Проверяем затраты у найденного расположения распределительного центра.

5. Если выполнен критерий остановки, то работа завершается.

Переходим на следующую итерацию,  $k := k + 1$ .

При запуске  $k$  итераций будет определено  $k$  мест размещения и одно или несколько из них будет с наименьшими производственно-транспортными затратами. Данное решение будет искомым местом размещения распределительного центра.

Компоненты вектора  $\alpha_i^{k+1}$  вычисляются следующим образом:

$$\alpha_i^{k+1} = \frac{\alpha_{min} + \rho \gamma_i^k (\alpha_i^k - \alpha_{min})}{\beta_k}, \quad i \in I, \quad (4)$$

где  $\beta_k \in (0, 1)$  – коэффициент затухания (испарение феромона) на итерации  $k$ ;  $\gamma_i^k \in [0, 1]$  – частота появления распределительного центра  $i$  в  $l$  лучших

<sup>1</sup> Жадный алгоритм (англ. Greedy algorithm) — алгоритм, заключающийся в принятии локально оптимальных решений на каждом этапе, допуская, что конечное решение также окажется оптимальным.



решениях, выбираемых на шаге 2 итерации  $k$ ; параметр  $q \in (0,1)$ . Таким образом, при данных значениях параметров  $\beta_k$  и  $q$ , чем чаще распределительный центр  $i$  попадает в  $l$  лучших решений по значению целевой функции, тем меньше становится соответствующее значение  $\alpha_i$ ,  $i \in I$ .

### Алгоритм искусственного муравья ant2-pm

Алгоритм искусственного муравья ant2-pm представляет собой вероятностную модификацию жадного алгоритма спуска [4]. Здесь  $I_s^r$  – множество открытых распределительных центров;  $\Delta f_i^r$  – изменение целевой функции в результате закрытия распределительного центра  $i, i \in I_s^r$ .

Алгоритм ant2-pm начинает работу с множества  $I_s^1 = I$  и завершает ее при  $|I_s^r| = p$ . На каждой итерации алгоритма ant2-pm генерируется множество распределительных центров

$$W^r(\lambda) = \left\{ i \in I_s^r \mid \Delta f_i^r < (1 - \lambda) \min_{i \in I_s^r} \Delta f_i^r + \lambda \max_{i \in I_s^r} \Delta f_i^r \right\} \quad (5)$$

где  $\lambda$  – параметр алгоритма,  $\lambda \in [0,1]$ . Используя  $W^r(\lambda)$ , вычисляются значения привлекательности  $\eta_i^r$  каждого  $i \in I_s^r$ . Привлекательность вычисляется по формуле:

$$\eta_i^r = \begin{cases} \Delta f_{max}^r - \Delta f_i^r, & i \in W^r(\lambda), \\ \varepsilon, & i \in I_s^r \setminus W^r(\lambda), \end{cases} \quad (6)$$

где  $f_{max}^r = \max_{i \in W^r(\lambda)} \Delta f_i^r$ , параметр  $\varepsilon > 0$ . Данный параметр необходим для того, чтобы всякий распределительный центр  $i \in I_s^r$  имел шанс быть закрытым. Любой распределительный центр  $i \in I_s^r$  имеет ненулевую вероятность закрытия. На каждом шаге  $r$  алгоритма ant2-pm с распределением вероятностей

$$p_i^r = \frac{\alpha_i^k * \eta_i^r}{\sum_{i \in I_s^r} \alpha_i^k \eta_i^r}, \quad i \in I_s^r, \quad (7)$$

случайным образом выбирается одно предприятие  $i_0$  из множества  $I_s^r$ , которое необходимо закрыть.

Как видно из схемы алгоритма ant2-pm, множество  $W^r(\lambda)$  также состоит из кандидатов для закрытия на шаге  $r$ , но уже в вероятностном смысле, т.е. даже если распределительный центр  $t \in I_s^r$  не попал во множество  $W^r(\lambda)$ , то он все равно имеет ненулевую вероятность быть закрытым.

Алгоритм ant2-pm:

0. Определяем начальное множество  $I_s^1 = I$ .

Шаг  $r, r \geq 1$ .

1. Если  $|I_s^r| = p$ , то работа алгоритма завершается.

2. Формируется множество  $W^r(\lambda)$  согласно (5).

3. Используя (7), случайно выбираем элемент  $t_0 \in I_s^r$ .

4. Определяем  $I_s^{r+1}, I_s^{r+1} := I_s^r \setminus \{t_0\}$ .

Переходим на следующий шаг,  $r := r + 1$ .

### Заключение

Муравьиные алгоритмы основаны на имитации самоорганизации социальных насекомых посредством использования динамических механизмов, с помощью которых система достигает глобальной цели в результате локального низкоуровневого взаимодействия элементов. В статье рассмотрен общий алгоритм муравьиной колонии для задач оптимального размещения распределительных центров розничной торговой сети. Алгоритм может быть успешно применен для решения различных задач размещения.

Основная идея, лежащая в основе алгоритма муравьиной колонии, заключается в использовании механизма положительной обратной связи, который помогает найти наилучшее приближенное решение в сложных задачах оптимизации. То есть, если в данном узле муравей должен выбрать между различными вариантами и если фактически выбранные результаты будут хорошими, то в будущем такой выбор будет более желателен, чем предыдущий. Эффективность муравьиных алгоритмов увеличивается с

ростом размерности задачи оптимизации. Муравьиные алгоритмы обеспечивают решения и других комбинаторных задач не хуже общих метаэвристических технологий оптимизации и некоторых проблемно ориентированных методов. Все это позволяет рекомендовать применение муравьиных алгоритмов для решения сложных комбинаторных задач оптимизации.

### **Литература**

1. Dorigo M. Optimization, Learning and Natural Algorithms // PhD thesis, Dipartimento di Electronica, Politecnico di Milano, IT, 1992 (in Italia).
2. Леванова Т.В., Лореш М.А. Алгоритм муравьиной колонии и имитации отжига для простейшей задачи размещения // Материалы конференции «Дискретный анализ и исследование операций», Новосибирск, 2002. – С. 235.
3. Лореш М.А. Алгоритмы муравьиной колонии для простейшей задачи размещения: Препринт. – Омск, ОмГУ, 2006. – 19с.
4. Лореш М.А. Разработка и исследование алгоритмов муравьиной колонии для решения задач оптимального размещения предприятий: дис. на соискание уч. степени канд. техн. наук: 05.13.01 – Системный анализ, управление и обработка информации; ОмГУ. Омск, 2006. 113 с.

### **References**

1. Dorigo M. Optimization, Learning and Natural Algorithms // PhD thesis, Dipartimento di Electronica, Politecnico di Milano, IT, 1992 (in Italia).
2. Levanova T.V., Loresh M.A. Algoritm murav'inoj kolonii i imitacii otzhiga dlja prostejshej zadachi razmeshhenija // Materialy konferencii «Diskretnyj analiz i issledovanie operacij», Novosibirsk, 2002. – S. 235.
3. Loresh M.A. Algoritmy murav'inoj kolonii dlja prostejshej zadachi razmeshhenija: Preprint. – Omsk, OmGU, 2006. – 19s.
4. Loresh M.A. Razrabotka i issledovanie algoritmov murav'inoj kolonii dlja reshenija zadach optimal'nogo razmeshhenija predpriyatij: dis. na soiskanie uch. stepeni kand. tehn. nauk: 05.13.01 – Sistemnyj analiz, upravlenie i obrabotka informacii; OmGU. Omsk, 2006. 113 s.