

Научная статья
УДК 656.135.022

Решение задачи транспортного планирования путем формирования маршрутов модифицированным муравьиным алгоритмом при внедрении стохастических популяций

Иван Алексеевич Асманов¹, Сергей Борисович Александров²,
Варвара Михайловна Макурина³

^{1,2,3}Московский автомобильно-дорожный государственный технический университет (МАДИ),
Москва, Россия

¹Asmanovvvvv@mail.ru, <https://orcid.org/0009-0002-8051-840X>

²a_s_b@mail.ru

³v.makurina@madi.ru

Аннотация. Одним из методов решения транспортных задач является муравьиный алгоритм. Вопросы транспортного характера обладают многофакторностью, при этом не всегда имеется возможность решить поставленные задачи оптимизации на практике. В данной работе предложен модифицированный муравьиный алгоритм с применением стохастической популяции, позволяющий предотвратить возможность застревания на локальных экстремумах, что делает решение задачи более эффективной. Написан программный код муравьиного алгоритма с модификацией и без модификации. Произведено сравнение по результатам работы двух алгоритмов.

Ключевые слова: муравьиный алгоритм, роевой интеллект, задачи маршрутизации транспорта.

Для цитирования: Асманов И.А., Александров С.Б., Макурина В.М. Решение задачи транспортного планирования путем формирования маршрутов модифицированным муравьиным алгоритмом при внедрении стохастических популяций // Автомобиль. Дорога. Инфраструктура. 2024. №2 (40).

Original article

Solving the transport planning problem by generating routes using a modified ant algorithm when introducing stochastic populations

Asmanov Ivan A.¹, Alexandrov Sergey B.², Makurina Varvara M.³

^{1,2,3}Moscow Automobile and Road Construction State Technical University (MADI), Moscow, Russia

¹Asmanovvvvv@mail.ru, <https://orcid.org/0009-0002-8051-840X>

²a_s_b@mail.ru

³v.makurina@madi.ru

Abstract. One of the methods for solving transport problems is the ant algorithm. Transport issues are multifactorial, and it is not always possible to solve the optimization problems in practice. In this paper, a modified ant algorithm using a stochastic population is proposed to prevent the possibility of getting stuck at local extrema, which will make solving the problem more efficient. The program code for the ant algorithm was written with and without modification. A comparison was made based on the results of the two algorithms.

Keywords: ant algorithm, swarm intelligence, transport routing problems.

For citation: Asmanov I.A., Alexandrov S.B., Makurina V.M. Solving the transport planning problem by generating routes using a modified ant algorithm when introducing stochastic populations. *Avtomobil'. Doroga. Infrastruktura*. 2024. №1 (39).

Введение

Формирование оптимальных маршрутов различного характера на транспортной сети представляет собой одну из основных проблем транспортного планирования, именуемая в комбинаторной оптимизации задачей маршрутизации транспорта (англ. Vehicle Routing Problem, VRP) [1]. Цели задачи данного класса сводятся к снижению затрат на стоимость перевозки, минимизации расстояния или времени на передвижение транспортных средств, за счет построения более выгодной последовательности посещения необходимых пунктов [2, 3]. Задача VRP представляет собой формирование более оптимальной последовательности посещения мест сбора или выгрузки товара. Одним из способов решения данной задачи является оптимизационный метод под названием муравьиный алгоритм.

Муравьиный алгоритм и проблема приостановки работы на локальных экстремумах

Муравьиный алгоритм является метаэвристическим алгоритмом по поиску оптимального объезда заданных вершин (грузоотправителей и грузополучателей). Алгоритм моделирует поведение муравьиной колонии. Когда муравей находит продовольствие, он возвращается в колонию,

откладывая феромон на пройденном маршруте, тем самым повышая вероятность выбора данного пути движения для других муравьев [4].

Классический муравьиный алгоритм можно представить в виде (рис. 1):

- выбор исходных параметров: определение вершин, которые необходимо объехать, выбор количества «муравьев» в популяции, определение показателей «стадности» и «жадности», то есть вес выбора следующей вершины по количеству «феромона» на участке к данной вершине или по длине этого участка, значение интенсивности испарения «феромона» и количество повторений алгоритма;
- выбор пути движения: на данном этапе идет моделирование возможных вариантов объезда, имитирующее перебор вариантов движения;
- определение и запись лучшего решения: по пройденным в данной итерации «муравьями» маршрутам определить и записать самый короткий. Если новый вариант хуже предыдущего, то новой записи не происходит;
- откладывание «феромона»: на пройденных «муравьями» участках откладываются «феромоны»;
- обновление «феромона»: на всех участках производится испарение «феромона» в зависимости от заданного значения интенсивности испарения;
- сравнение фактического количества итераций алгоритма с заданным, после чего производится следующий цикл алгоритма или его завершение [5, 6].

При более подробном рассмотрении этапа с определением пути движения «муравья» во время итерации можно выявить причину, из-за которой «муравьи» могут застревать на локальных экстремумах, то есть перемещаться по одному и тому же маршруту, не выбирая новых вариантов движения [7, 8].

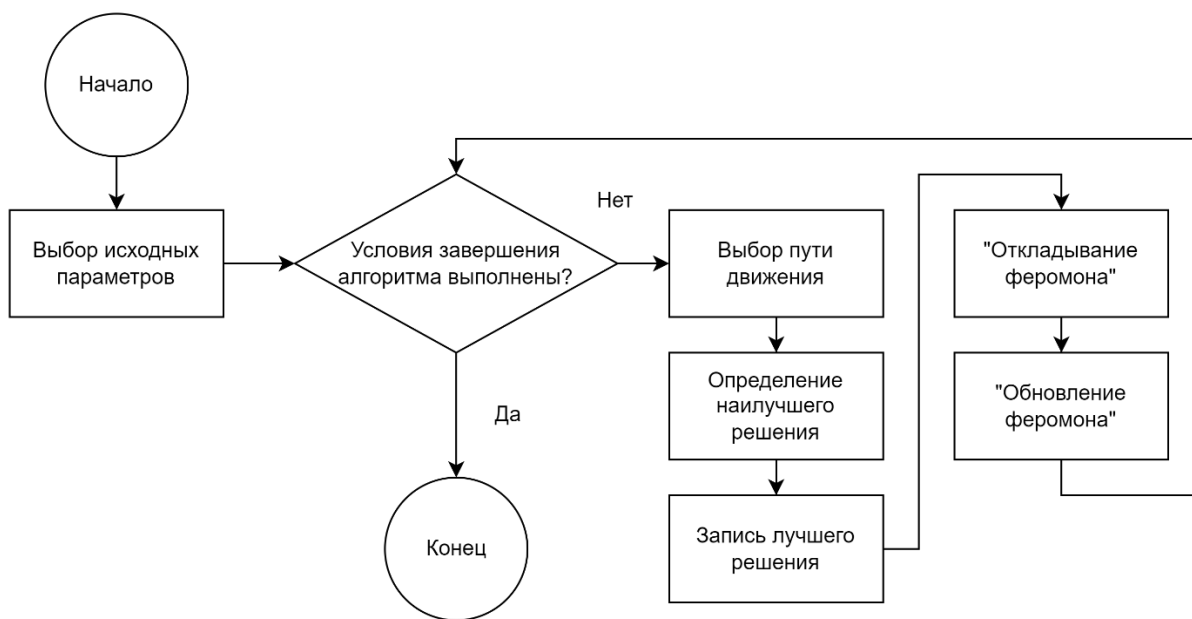


Рис. 1. Блок-схема работы классического муравьиного алгоритма

Для расчета вероятности перемещения «муравья» в определенную вершину необходимы показатели «стадности» и «жадности»: расстояние от вершины с «муравьем» до других возможных для перемещения вершин и количество «феромона», расположенном на участках между этими вершинами (1).

$$P_{ij} = \frac{t_{ij}^a (\frac{1}{d_{ij}})^b}{\sum_{j \in J} t_{ij}^a (\frac{1}{d_{ij}})^b}, \quad (1)$$

где P – вероятность перемещения «муравья» из вершины i в вершину j ;

t – количество «феромона» на участке;

d – эвристическое расстояние между вершинами;

a – показатель «стадности»;

b – показатель «жадности».

Когда показатель «стадности» равен нулю, выбор ближайшей вершины при построении маршрута наиболее вероятен [9]. Если показатель «жадности» в начале работы выбрать равным нулю, «муравьи» начнут

выбирать следующий пункт только по количеству «феромона» между вершинами. При рассмотрении ситуации, когда «муравей» выбирает следующий пункт только по количеству отложенного «феромона», некоторые варианты передвижения могут быть нереализованными.

На рисунке 2 представлен фрагмент модели транспортной сети. В варианте А «муравей» движется из вершины 1. У него есть три варианта, куда он может пойти, при этом на каждом участке есть какое-то количество отложенного «феромона». В данном случае у «муравья», как и у всех последующих, есть вероятность передвижения в любую из трех вершин для формирования нового маршрута. Но во время выполнения алгоритма может произойти ситуация В, при которой «муравьи» чаще всего выбирали вершину под номером 2, а «феромон» к вершинам 3 и 4 испарился. В данном случае «муравьи» перестают видеть возможность перемещения из вершины 1 в вершины 3 и 4, при том, что более оптимальный маршрут может проходить через них.

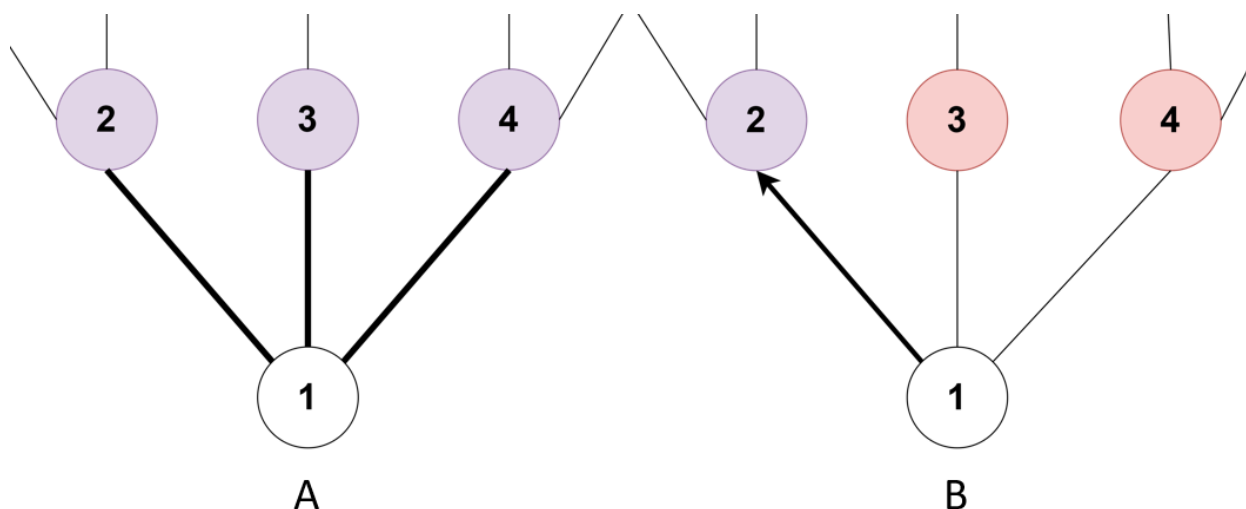


Рис. 2. Фрагмент модели транспортной сети с застреванием на локальных экстремумах

Модифицированный муравьиный алгоритм

Для предотвращения возможности появления застреваний на локальных экстремумах разработан модифицированный муравьиный

алгоритм с применением стохастической популяции, суть которого заключается в добавлении между основными итерациями дополнительных. Во время выполнения этих дополнительных итераций производится изменение классической работы алгоритма.

Во-первых, количество «муравьев» в популяции сокращается на случайное число от единицы до заданного количества «муравьев». При этом вес «феромона муравья» становится больше и считается по формуле (2).

$$Y_{ij}^n = r \times \frac{1}{D_n}, \quad (2)$$

где Y – количество отложенного «феромона» на участке от вершины i до вершины j «муравьем» n ;

r – коэффициент веса «муравья»;

D – расстояние, пройденное «муравьем» n .

Коэффициент веса «муравья» определяется по формуле (3).

$$r = \frac{N}{N_r}, \quad (3)$$

где N – заданное количество «муравьев» в популяции;

N_r – сгенерированное количество «муравьев» в дополнительной итерации.

Во-вторых, показатель «стадности» и показатель «жадности» меняются местами в расчетах, так, если «муравьи» при выполнении алгоритма застряли на участке с большим количеством «феромона», то они могли выбрать иную стратегию поведения. Определение вероятности перехода в новую вершину при изменении стратегии представлено в формуле (4). После выполнения дополнительной итерации изменения, представленные в этом пункте, отменяются и алгоритм выполняется как классический до следующей дополнительной итерации.

$$P_{ij} = \frac{t_{ij}^b \left(\frac{1}{d_{ij}}\right)^a}{\sum_{j \in J} t_{ij}^b \left(\frac{1}{d_{ij}}\right)^a}, \quad (4)$$

В качестве проверки работы модифицированного муравьиного алгоритма с применением стохастической популяции был сформирован программный код на языке Python версии 3.9.5 с библиотеками random.py и NumPy 1.26.0 (рис. 3) [10, 11]. В качестве исходных данных были использованы:

- матрица кратчайших расстояний на 13 вершин;
- коэффициент испарения «феромона» равный 0,2;
- показатель «жадности» равный 0,2;
- показатель «стадности» равный 0,8;
- количество «муравьев» в популяции равное 13;
- количество итераций равное 1000.

Сравнение работы классического муравьиного алгоритма и модифицированного муравьиного алгоритм с различными дополнительными итерациями представлено в таблице 1.

Таблица 1

Сравнение классического и модифицированного муравьиных алгоритмов

Показатель	Классический алгоритм	Модифицированный алгоритм		
		Каждая 2-я итерация дополнительная	Каждая 4-я итерация дополнительная	Каждая 16-я итерация дополнительная
Кратчайшее расстояние	142	127	137	139
Максимальная итерация, после которой изменения не происходили	113	999	989	977
Количество экспериментов	1000	1000	1000	1000
Количество экспериментов, показывающих кратчайшее расстояние	178	490	672	780

Количество выполнений программ с установленными условиями и модификациями составило тысячу экспериментов в каждом из

рассмотренных случаев. В рамках применения классического алгоритма наблюдалось отсутствие изменений в программе после 113-й итерации. Дополнительные итерации в рамках тысячи экспериментов не приносили значительных изменений, поскольку произошло прекращение работы муравьиного алгоритма из-за достижения локальных экстремумов.

Во время функционирования модифицированной версии алгоритма наблюдался продолжающийся поиск оптимального маршрута практически на каждой итерации. Значительно отличаясь от классического алгоритма, модифицированный вариант продемонстрировал лучшую эффективность в поиске оптимального маршрута с минимальным общим пробегом.

```
for it in range(num_iterations):
    # подготовка к доп итерации
    if it % dopiter == 0:
        alpha = alpha + beta
        beta = alpha - beta
        alpha = alpha - beta
        num_ants = random.randint(1, n)
        k = n/num_ants

    # инициализация муравьев
    ants = [[random.randint(0, num_cities - 1)] for i in range(num_ants)]
    for i in range(num_ants):
        ants[i][0] = 0
    for i in range(num_ants):
        for j in range(num_cities - 1):
            current_city = ants[i][-1]
            # расчет вероятностей перехода в следующий город
            probs = [0] * num_cities
            G=0
            for next_city in range(num_cities):
                if next_city not in ants[i]:
                    G+=((pheromone[current_city][next_city] ** alpha) *
                        ((1.0 / distances[current_city][next_city]) ** beta))
            for next_city in range(num_cities):
                if next_city not in ants[i]:
                    probs[next_city] = ((pheromone[current_city][next_city] ** alpha) *
```

Рис. 3. Фрагмент кода модифицированного муравьиного алгоритма

Заключение

Анализ результатов эксперимента выявляет существенные различия в функционировании классического алгоритма в сравнении с модифицированным. Путем применения дополнительных итераций был обнаружен более оптимальный сценарий объезда вершин, а поиск новых вариантов маршрута осуществлялся и в финальных стадиях выполнения

программы, что свидетельствует об отсутствии застреваний «муравьев» в локальных экстремумах. При проведении эксперимента классический алгоритм продемонстрировал худшую эффективность из участвующих алгоритмов в определении маршрута с минимальным общим пробегом. В ходе выполнения алгоритма наблюдалось прекращение процесса поиска новых маршрутов, что произошло после выполнения приблизительно восьмисот однотипных вычислений.

Список источников

1. Чернышев, Ю. О. Обзор динамических задач маршрутизации транспорта / Ю. О. Чернышев, В. Н. Кубил // Программные продукты и системы. – 2020. – № 3. – С. 491-501. – EDN QAPNLC.
2. Lutsenko, E. A. Model for determining the optimal size of a multi-nomenclature cargo batch, taking into account the weight and volume characteristics of the transport vehicle / E. A. Lutsenko // International Journal of Advanced Studies. – 2020. – Vol. 10, No. 1. – P. 26-34. – DOI 10.12731/2227-930X-2020-1-26-34. – EDN IUVAXK.
3. The Use of Artificial Neural Networks (ANN) as an Auxiliary Factor in Planning Transportation Routes: Theoretical Aspects of Artificial Intelligence Systems Development for Transportation Engineering / A. A. Chernyshev, E. A. Koryagina, D. G. Moroz, S. S. Titova // 2022 Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022 - Conference Proceedings, Moscow, 15–17 марта 2022 года. – Moscow, 2022. – DOI 10.1109/IEEECONF53456.2022.9744087. – EDN ZICXHD.
4. Просов, С. Н. Декомпозиция задачи маршрутизации по эвристикам метода Кларка-Райта / С. Н. Просов, Е. А. Кузьменко // Мир транспорта. – 2018. – Т. 16, № 3(76). – С. 190-199. – EDN LXZFNH.
5. Сидоренко, Д. О. О решении задачи маршрутизации транспорта с помощью подвижного генетического алгоритма / Д. О. Сидоренко, А. Ю. Городилов // Вестник Пермского университета. Математика. Механика. Информатика. – 2021. – № 4(55). – С. 43-48. – DOI 10.17072/1993-0550-2021-4-43-48. – EDN XWCTDB.
6. Микулик, И. И. Распараллеливание гибридного алгоритма муравьиной колонии с изменяющимися с помощью генетического алгоритма параметрами / И. И. Микулик, Е. А. Благовещенская // Проблемы информатики. – 2023. – № 2(59). – С. 86-94. – DOI 10.24412/2073-0667-2023-2-86-97. – EDN HBTPLC.

7. Предикативный подход к анализу конфликтности в транспортном потоке / В. В. Дронсейко, А. М. Меркович, А. В. Замыцких, О. И. Максимычев // Мир транспорта и технологических машин. – 2023. – № 3-1(82). – С. 86-92. – DOI 10.33979/2073-7432-2023-3-1(82)-86-92. – EDN VUPKVF.
8. Проблемы создания регуляторной базы в процессе цифровизации автоперевозок / Т. Е. Мельникова, С. Е. Мельников, В. М. Макурина, С. Кахраманова // Транспорт: наука, техника, управление. Научный информационный сборник. – 2021. – № 9. – С. 49-52. – DOI 10.36535/0236-1914-2021-09-9. – EDN NLLPSA.
9. Asmanov, I.A. Development of A Hardware and Software Complex for Optimizing Logistics Activities in the Field of Consumer Waste Management / I.A. Asmanov, V.V. Zavyazkina, D.G. Moroz, A.I. Zhukov // Systems of Signals Generating and Processing in the Field of on-Board Communications. – 2023. – Vol. 1, No. 6. – P. 45-48. – DOI 10.1109/IEEECONF56737.2023.10092123.
10. GitHub. Asmanson: AntAlgorithm [Электронный ресурс]. – Режим доступа: <https://github.com/Asmanson/AntAlgorithm/blob/main/AntMod> (дата обращения: 30.11.2023).
11. Python: Welcome to Python.org [Электронный ресурс]. – Режим доступа: <https://www.python.org/> (дата обращения: 30.11.2023).

References

1. Chernyshev Yu.O., Kubil V.N., *Programmnyye produkty i sistemy*, 2020, no. 3, pp. 491-501.
2. Lutsenko E.A. *International Journal of Advanced Studies*, 2020, vol. 10, no. 1, pp. 26-34, DOI 10.12731/2227-930X-2020-1-26-34.
3. Chernyshev A.A., Koryagina E.A., Moroz D.G., Titova S.S. 2022 *Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022*, Conference Proceedings, Moscow, 2022, DOI 10.1109/IEEECONF53456.2022.9744087.
4. Prosov S.N., Kuzmenko E.A. *Mir transporta*, 2018, vol. 16 no. 3(76), pp. 190-199.
5. Sidorenko D.O., Gorodilov A.Y. *Vestnik Permskogo universiteta. Matematika. Mekhanika. Informatika.*, 2021, no. 4(55), pp. 43–48, DOI 10.17072/1993-0550-2021-4-43-48.
6. Mikulik I.I., Blagoveshchenskaya E.A. *Problemy informatiki*, 2023, no. 2 (59), pp. 86-94, DOI 10.24412/2073-0667-2023-2-86-97.
7. Dronseiko V.V., Merkovich A.M., Zamytskikh A.V., Maksimych O.I. *Mir transporta i tekhnologicheskikh mashin*, 2023, no. 3-1 (82), pp. 86-92.
8. Makurina V.M., Melnikova T.E., Melnikov S.E., Kahramanova S. *Transport: nauka, tekhnika, upravleniye. Nauchnyy informatsionnyy sbornik*, 2021, no. 9, pp. 49-52, DOI 10.36535/0236-1914-2021-09-9.

9. Asmanov I.A., Zavyazkina V.V., Moroz D.G., Zhukov A.I. *Systems of Signals Generating and Processing in the Field of on-Board Communications*, 2023, vol. 1, no. 6, pp. 45-48, DOI 10.1109/IEEECONF56737.2023.10092123.
10. URL: <https://github.com/Asmanson/AntAlgorithm/blob/main/AntMod> (11.30.2023).
11. URL: <https://www.python.org/> (11.30.2023).

Рецензент: Ю.А. Короткова, канд. техн. наук, доц., МАДИ

Информация об авторах

Асманов Иван Алексеевич, студент, МАДИ.
Александров Сергей Борисович, канд. техн. наук, доц., МАДИ.
Макурина Варвара Михайловна, аспирант, МАДИ.

Information about the authors

Asmanov Ivan A., student, MADI.
Alexandrov Sergey B., Candidate of Sciences (Technical), associate professor, MADI.
Makurina Varvara M., postgraduate, MADI.

Статья поступила в редакцию 15.12.2023; одобрена после рецензирования 22.05.2024; принята к публикации 13.06.2024.

The article was submitted 15.12.2023; approved after reviewing 22.05.2024; accepted for publication 13.06.2024.