

Применение эволюционного метода генетического анализа для оптимизации грузовых перевозок в городской среде

Е.А. Гладченко¹, О.Н. Сапрыкин¹, А.Н. Тихонов¹

¹Самарский национальный исследовательский университет им. академика С.П. Королева, Московское шоссе 34А, Самара, Россия, 443086

Аннотация. Модель, разработанная в данном исследовании, направлена на решение следующей задачи: расчет оптимальных замкнутых маршрутов доставки грузов от распределительного центра к местам назначения, учитывая неравномерность загруженности улично-дорожной сети, а также ограничения по грузоподъемности автопарка. Целью является сокращение логистических затрат путем минимизации времени, затраченного в пути. Таким образом, модель объединяет решение двух отдельных задач: коммивояжера и задачу о ранце.

1. Введение

В наши дни, когда ежедневно во всем мире по земле, воде и воздуху перевозятся сотни тысяч тонн груза, задача оптимизация логистических процессов является крайне актуальной. Это сложная, комплексная проблема, которая требует системного подхода. С одной стороны, это связано с тем, что в процесс перевозки вовлечено большое количество участников, а решения и выполняемые действия должны быть хорошо скоординированы, поэтому необходимо постоянное обеспечение высокого уровня коммуникаций. С другой стороны, цепочка поставки состоит из множества взаимосвязанных шагов, таких как определение объема и характера спроса, выбор маршрута доставки и типа транспортного средства, консолидация и отслеживание заказов, организация перевалок и складирования, обеспечение выполнения мультимодальных перевозок, подготовка сопутствующей документации. Логистическая задача может быть решена с помощью математических методов, однако такое решение требует слишком большого количества времени, ресурсов и дает решение по ограниченному числу узлов, вследствие чего вместо формализованного метода применяют эвристический, к которому, в частности, принадлежит генетический анализ [1].

2. Методология

Для достижения максимального эффекта при решении логистической задачи, ее следует рассматривать как сложную, комплексную проблему, учитывающую всевозможные аспекты процесса перевозки. Однако, в некоторых случаях, эту проблему необходимо дифференцировать на более узкие элементы. Можно выделить несколько основных логических сочетаний логистических подзадач: сгруппированная ориентированная задача, ориентированная задача с временными окнами и сгруппированная ориентированная задача с временными окнами. Каждая из задач в отдельности математически формализуется и решается

[2]. Альтернативным подходом – выделение двух взаимозависимых подзадач: задачи коммивояжера и задачи о ранце, которые в свою очередь могут быть решены итерационно [3].

2.1. Математическая модель

Для получения более точного решения, необходимо брать во внимание изменение интенсивности транспортного потока в разное время суток, в разных районах города, которое влияет на скорость движения машин. Однако статистические данные с городским автомобильным трафиком труднодоступны, в связи с чем был применен метод транспортного зонирования, основанный на гравитационном моделировании, где матрицы корреспонденции содержат информацию о характере перемещения населения, что дает возможность вычисления средней скорости движения [4]. Модель проиллюстрирована на рисунке 1.

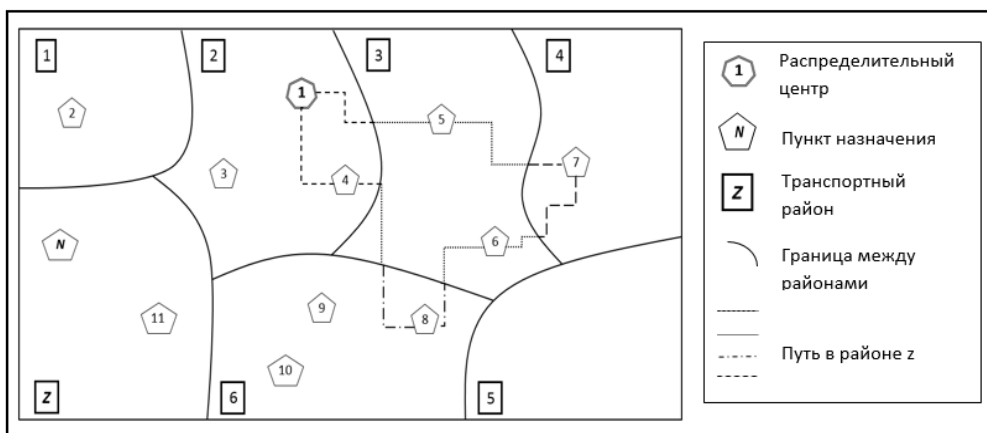


Рисунок 1. Пример маршрута доставки груза, начало и конец которого лежит в распределительном центре (1-5-7-6-8-4-1).

2.2. Генетический алгоритм

Эволюционный метод генетического анализа дает возможность обработать большой массив данных, закодированный двоичным кодом, выполняя последовательно заданные итерации до тех пор, пока оптимальное решение не будет найдено. Родительская хромосома генерируется случайно, а последующие поколения – путем применения операторов скрещивания и/или мутации. Хромосома имеет вид ориентированной матрицы корреспонденции M_k , строки которой – пункты отправления (i), а столбцы – назначения (j) [5]. Пример приведен на рисунке 2. Чтение матрицы начинается с верхней строчки $i = 1$. Если значение в таблице $x_{ij} = 1$, значит, маршрут включает в себя сообщение между пунктами i и j, в остальных случаях $x_{ij} = 0$.

На матрицу накладываются следующие ограничения:

$$\sum_{i=1}^N x_{ij} \leq 1; \quad \forall j = 1 \dots N, \tag{1}$$

$$\sum_{j=1}^N x_{ij} \leq 1; \quad \forall i = 1 \dots N. \tag{2}$$

Из матрицы M_k формируется вектор P_k , для которого должно соблюдаться правило: $p_1^P = p_n^P = 1$, так как из условия задачи маршрут берет начало и заканчивается в распределительном центре.

В данной модели задано две целевых функции, что связано с комбинированием сразу двух задач – поиск оптимального маршрута и максимальной загрузки. Генетический алгоритм реализован с использованием пакета Dear [6], который содержит генетические операторы и средства для построения любой эволюционной модели. Первая целевая функция была реализована как пользовательская функция и зарегистрирована в Dear framework как параметр «оценить». Внутри основного цикла эволюции, который решает задачу коммивояжера, мы добавили вложенный цикл эволюции, который решает задачу ранца для каждой хромосомы, выбранной в качестве популяции для следующего поколения [7].

Первая целевая функция T_k – время, затраченное на доставку груза по составленному маршруту.

<i>i \ j</i>	1	2	3	4	5	6	7	8	...	N-1	N
1	0	0	0	0	1	0	0	0		0	0
2	0	0	0	0	0	0	0	0		0	0
3	0	0	0	0	0	0	0	0		0	0
4	1	0	0	0	0	0	0	0		0	0
5	0	0	0	0	0	0	1	0		0	0
6	0	0	0	0	0	0	0	1		0	0
7	0	0	0	0	0	1	0	0		0	0
8	0	0	0	1	0	0	0	0		0	0
...											
N-1	0	0	0	0	0	0	0	0		0	0
N	0	0	0	0	0	0	0	0		0	0

Рисунок 2. Пример ориентированной матрицы с маршрутом (1-5-7-6-8-4-1).

$$T_k = \sum_{j=p_1}^{p_{n-2}} \sum_{z=1}^Z \left(\frac{2 \times l_{j,j+1}}{v_z^j + v_z^{j+1}} + d_{j+1} \times ts + te \right) + \frac{2 \times l_{p_{n-1}, p_n}}{v_z^{p_{n-1}} + v_z^{p_n}} \rightarrow \min \quad (3)$$

где d_j – объем спроса в пункте j , ($j=2...N$); c – грузоподъемность машины; z – транспортный район города, ($z=1...Z$); l_{ij} – расстояние между i и j , ($i=1...N$), ($j=1...N$); \bar{v}_z^I, \bar{v}_z^J – средняя скорость в z -ом районе; ts – средняя длительность отгрузки в пункте (задается); te – дополнительное время обслуживания (задается); p_n^P – ген хромосомы, означающий номер узла в маршруте, где n – порядковый номер гена в хромосоме P .

После чего решения проходят проверку на ограничение по грузоподъемности. Оно формируется, исходя из сравнения грузоподъемности машины и суммарного тоннажа груза в доставке (зависит от состава хромосомы).

$$\sum_{j=p_2}^{p_{n-1}} d_j \leq c \quad (4)$$

Затем отобранные хромосомы рассчитываются по второй целевой функции:

$$\sum_{j=p_2}^{p_{n-1}} d_j \rightarrow \max \quad (5)$$

Маршрут с рисунка 1 можно представить в виде вектора P_1 : (1-5-7-6-8-4-1), где $p_1^P = p_n^P = 1$ в любом из случаев. Это условие выполняется, так как доставка всегда начинается и заканчивается в депо, число которого равно $i = j = 1$.

После этого целевая функция (ЦФ) ЦФ1 (3) рассчитывается для всех K индивидуумов, и алгоритм проверяет ограничение (2). Некоторые хромосомы из первой популяции будут удалены из-за несоответствия с ограничением по грузоподъемности.

После этих шагов стандартный алгоритм может перейти к следующим итерациям, но наш подход содержит дополнительный шаг. Остальные K векторы оцениваются по ЦФ2 (5). На рисунке 3 показан пример одного из возможных решений задачи.

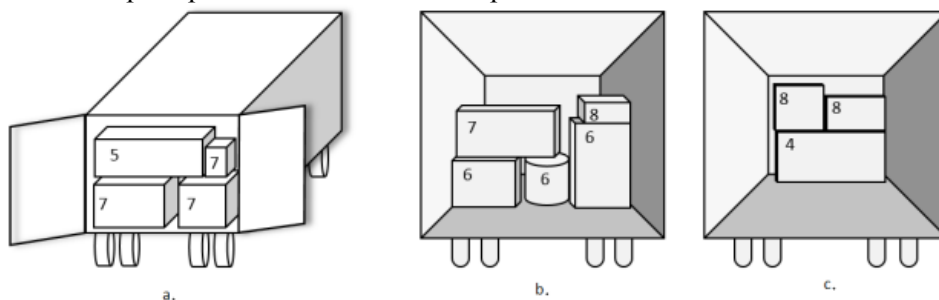


Рисунок 3. Пример загрузки автомобиля, доставляющего груз по маршруту (1-5-7-6-8-4-1): а) третий (крайний) слой загрузки; б) второй (средний) уровень загрузки; с) первый (начальный) слой загрузки.

Блок-схема выполнения программы приведена на рисунке 4.

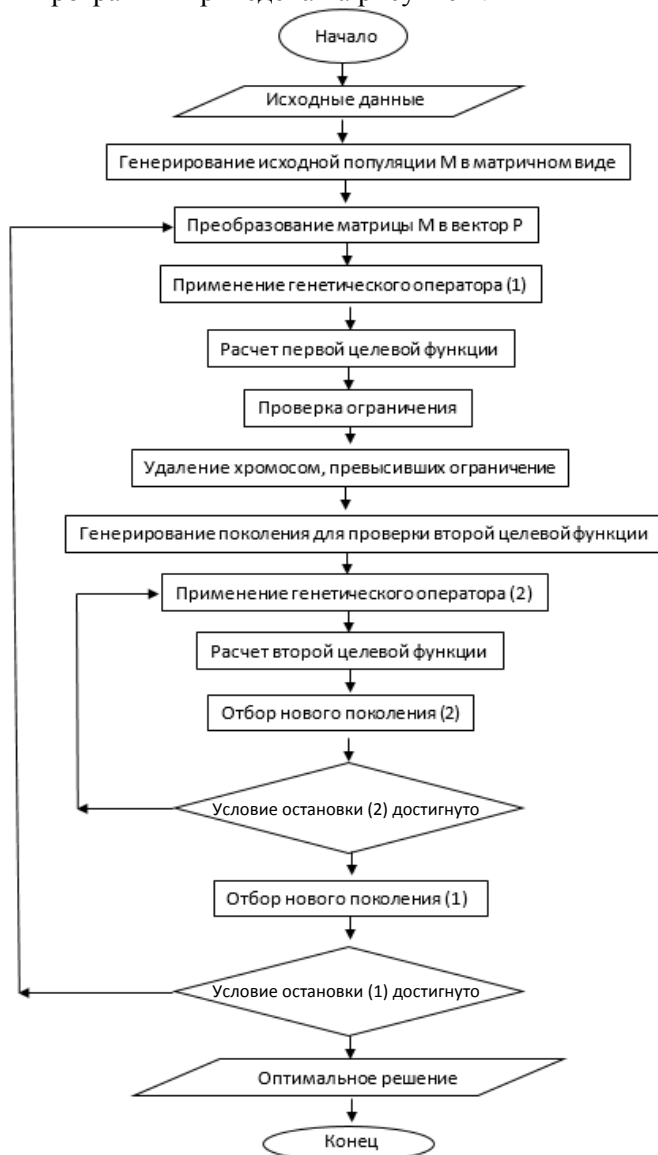


Рисунок 4. Блок-схема решения поставленной задачи.

3. Реализация

Решение поставленной задачи реализовано на языке программирования Python, который позволяет работать с моделью итеративно, пошагово внося изменения. Для работы с пространственными данными в нашем проекте использовались следующие пакеты Python:

- Shapely (низкоуровневая обработка геометрий) [8];
- GeoPandas (обработка кадров данных с помощью столбца геометрии) [9];
- NetworkX (работа с пространственными графами) [10];
- OSMnx (сеть загрузки для некоторого региона из ресурса OpenStreetMap) [11];
- Matplotlib (электроинструмент для печати с поддержкой географических карт) [12].

В качестве среды был выбран Zerpelin [13], поскольку он предоставляет возможность создания масштабируемых ноутбуков с богатым пользовательским интерфейсом. Разработанное программное обеспечение реализовано в виде Docker контейнера, который содержит все настроенные модули и пакеты и может быть развернут в большинстве распространенных операционных систем. Скрипты и ноутбуки хранятся в репозитории Bitbucket git [14]. На рисунке 5 представлена архитектура предлагаемого решения.

Генетический алгоритм реализован с использованием пакета Dear, который содержит генетические операторы и средства для построения любой эволюционной модели. Первая целевая функция была реализована как пользовательская функция и зарегистрирована в Dear framework как параметр «оценить». Внутри основного цикла эволюции, который решает задачу коммивояжера, мы добавили вложенный цикл эволюции, который решает задачу ранца для каждой хромосомы, выбранной в качестве популяции для следующего поколения.

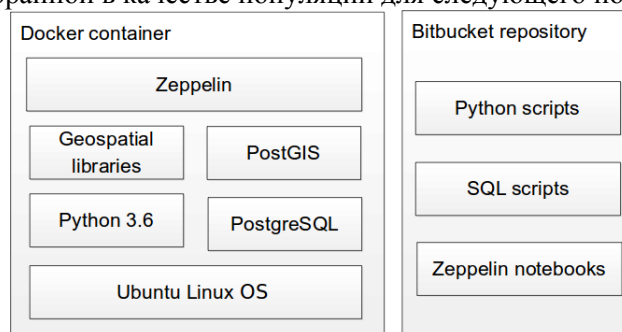


Рисунок 5. Архитектура решения.

4. Заключение

Решение реализовано в виде записной книжки Python, что позволяет итеративно работать с моделью и шаг за шагом улучшать ее. Генетический анализ используется в качестве инструмента поддержки принятия решений по грузовым перевозкам. Модель городских грузоперевозок позволяет нам определить оптимальный маршрут доставки (точки отправления-назначения), минимизируя время в пути и максимизируя загрузку грузового автомобиля. Основным преимуществом предлагаемого подхода является применение транспортного зонирования, которое позволяет приблизить решения к реальной ситуации.

Дальнейшие исследования могут учитывать временные рамки поставок и различия в возможностях парка машин. Кроме того, генетический алгоритм требует настройки, чтобы получить оптимальную производительность выполнения.

5. Литература

- [1] Speranza, M.G. Trends in transportation and logistics // *European Journal of Operational Research*. – 2016. – Vol. 264. – P. 830-836.
- [2] Vansteenwegen, P. The orienteering problem: A survey / P. Vansteenwegen, W. Souffriau, D.V. Oudheusden // *European Journal of Operational Research*. – 2010. – Vol. 209. – P. 1-10.
- [3] Yafrani, El.M. A local search based approach for solving the Travelling Thief Problem: The pros and cons / El.M. Yafrani, B. Ahiod // *Applied Soft Computing*. – 2016. – Vol. 52. – P. 795-804.
- [4] Barthelemy, M. Spatial Networks // *Physics Reports*. – 2011. – Vol. 499. – P. 1-101.
- [5] Michalewicz, Z. Genetic Algorithms + Data Structures = Evolution Programs // New York: Springer-Verlag, Berlin Heidelberg, 1996.
- [6] Вычислительная среда Dear [Электронный ресурс]. – Режим доступа: <https://github.com> (05.07.2018).
- [7] Saprykin, O.N. Validation of Transport Infrastructure Changes via Microscopic Simulation: A Case Study for the City of Samara, Russia / O.N. Saprykin, O.V. Saprykina // *Proceedings of the 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems, 2017*. – P. 788-793.
- [8] Репозиторий Shapely для работы с геометрическими объектами [Электронный ресурс]. – Режим доступа: <https://pypi.org> (05.07.2018).
- [9] Библиотека GeoPandas для работы с геопространственными данными [Электронный ресурс]. – Режим доступа: <http://geopandas.org> (05.07.2018).
- [10] Пакет NetworkX для Python [Электронный ресурс]. – Режим доступа: <https://networkx.github.io> (05.07.2018).

- [11] Пакет Open Street Map для Python [Электронный ресурс]. – Режим доступа: <https://osmnx.readthedocs.io> (05.07.2018).
- [12] Двухмерная библиотека черчения Python Matplotlib [Электронный ресурс]. – Режим доступа: <https://matplotlib.org> (05.07.2018).
- [13] Интерпретатор Python для Apache Zeppelin [Электронный ресурс]. – Режим доступа: <https://zeppelin.apache.org> (05.07.2018).
- [14] Хостинг проекта [Электронный ресурс]. – Режим доступа: <https://bitbucket.org> (05.07.2018).

Evolutionary approach for urban freight transportation optimization

Е.А. Gladchenko¹, О.Н. Saprykin¹, А.Н. Tikhonov¹

¹Samara National Research University, Moskovskoe Shosse 34A, Samara, Russia, 443086

Abstract. With the passing of the years, the process of logistics decision-making become more complicated that is why data driven decisions required nowadays. As well as shipment problems are NP-hard, the heuristic methods should be applied to resolve them. In this article we propose a genetic algorithm to solve the complex problem consists of the Travelling Salesman Problem combined with the Knapsack Problem. We have developed an urban freight transportation model which is focused on the minimization of the underway time as well as on the maximization of the truck's loading. A significant contribution in our method is the census of traffic frequency by using traffic zoning.