

УДК 656.02

*Лебедева Ольга Анатольевна,*  
к.т.н., доцент кафедры «Управление на автомобильном транспорте»,  
ФГБОУ ВО «Ангарский государственный технический университет»,  
тел.: +7(952)6326611, e-mail: kravhome@mail.ru

*Кулакова Ирина Михайловна,*  
к.т.н., доцент кафедры «Вычислительные машины и комплексы»,  
ФГБОУ ВО «Ангарский государственный технический университет»,  
тел.: +7(908)651357, e-mail: iyelkina@mail.ru

*Ерофеев Ермак Владимирович,*  
студент гр. ИТБ-23-1,  
ФГБОУ ВО «Ангарский государственный технический университет»,  
тел.: +7(950)0535465, e-mail: ermak.080@mail.ru

## РАЗРАБОТКА И ИСПОЛЬЗОВАНИЕ БИБЛИОТЕК PYTHON С ОТКРЫТЫМ ИСХОДНЫМ КОДОМ ДЛЯ ЗАДАЧ ТРАНСПОРТНОГО МОДЕЛИРОВАНИЯ

*Lebedeva O.A., Kulakova I.M., Erofeev E.V.*

### DEVELOPING AND USING OPEN-SOURCE PYTHON LIBRARIES FOR TRANSPORTATION MODELING PROBLEMS

**Аннотация.** В статье рассматривается универсальный инструмент транспортного моделирования язык Python, повсеместно распространенный среди приложений, ориентированных на данные. В статье приведен обзор библиотеки Quetzal, которая устраняет некоторые ограничения программного обеспечения для моделирования с закрытым исходным кодом. Она предлагает инструменты, помогающие выполнять задачи моделирования в областях предварительной обработки, проверки целостности и управления данными.

**Ключевые слова:** транспортное моделирование, язык программирования Python, открытый код, обработка данных.

**Abstract.** This article discusses a versatile transportation modeling tool, the Python language, which is ubiquitous among data-centric applications. The article provides an overview of the Quetzal library, which addresses some of the limitations of closed-source modeling software. It offers tools to help perform modeling tasks in the areas of preprocessing, integrity checking, and data management.

**Keywords:** transport modeling, programming language Python, open source, data processing.

Одним из высокоуровневых языков программирования, которые в последний период превратились в универсальный инструмент моделирования, является Python, который был разработан в конце 1980-х годов. В середине 2000-х годов он стал повсеместно распространенным среди приложений, ориентированных на данные, после выпуска пакетов Pandas и Scikit Learn. Разработка Networkx и Shapely позволила проводить эффективный сетевой анализ и геоматику. В течение многих лет Python используется параллельно с традиционным программным обеспечением для моделирования транспорта.

С ростом возможностей программирования на Python, появилось больше эффективных библиотек с открытым кодом, которые давали возможность и средства для моделирования транспорта [1].

Программное обеспечение с закрытым исходным кодом имеет три основных недостатка: сложная интегрируемость, недоступность и низкая прозрачность. Многие компании не могут позволить себе тратить несколько сотен тысяч рублей в год на поддержку программного обеспечения для моделирования транспорта. Интеграция прогнозирования дорожного движения часто затрудняется интерфейсом прикладного программирования обеспечения для моделирования. Библиотеки позволяют использовать всю мощь и гибкость Python для общих задач, таких как взаимодействие с различными источниками дорожного движения, в то время как специализированные задачи моделирования транспорта могут быть выполнены в библиотеке Quetzal. Более того, несмотря на значительные усилия по написанию и поддержанию программного обеспе-

чения с закрытым исходным кодом, фактические алгоритмы и эвристики иногда остаются размытыми.

Цель библиотеки Quetzal – предоставить надежное автономное решение, которое устраняет эти ограничения программного обеспечения для моделирования с закрытым исходным кодом. Опираясь на определенную объектную модель, Quetzal предлагает инструменты, помогающие решать задачи в областях предварительной обработки данных и проверки целостности, управления данными и моделирования транспорта.

Многие варианты программного обеспечения с закрытым исходным кодом, такие как VISUM и CUBE используются во многих транспортных организациях. Установлено, что спецификация данных этого программного обеспечения не столь универсальна.

Процедуры и алгоритмы для эффективного поиска множественных путей подробно описаны в литературе [2-4]. Эти описания использовались для внедрения алгоритма Франка-Вульфа. Решения с пунктом отправления и пунктом назначения не масштабируются для матриц корреспонденций. В противном случае установлено, что простота взвешенного графа более удобна в меняющемся многомодальном контексте. Действительно, когда варианты пересадки представлены в более сложной структуре, их сложнее объединить с другими процедурами в эффективной стратегии поиска пути. Поэтому поиск был сужен до алгоритмов поиска пути в взвешенном графе (где атрибуты вершин не используются, а ребра не несут никакой информации, кроме своего веса). В плотных сетях количество возвращаемых путей увеличивается экспоненциально с критериями толерантности. Для отображения данных на улично-дорожных сетях задачи были решены путем объединения алгоритмов четырех ключевых библиотек: Pandas, Networkx, Scikit-Learn и Shapely.

В объектно-ориентированном языке «объект» – это сущность, которая может содержать данные, также называемые «атрибутами» и кодом или «методами». Входные данные методов могут быть внутренними по отношению к «атрибутам» объекта или внешними, в этом случае они называются «параметрами». Выходные данные могут либо храниться как атрибуты, либо возвращаться. Quetzal основан на объекте, называ-

емом «StepModel», который сохраняет данные, и производит процедуры моделирования доступными методами (рис. 1).

В рассматриваемом примере, на рис. 2, основное внимание уделяется методу назначения. Этот метод основан на: матрице «места назначения» (атрибут); транспортных путях (атрибуты); штрафе за пересадку (параметр).

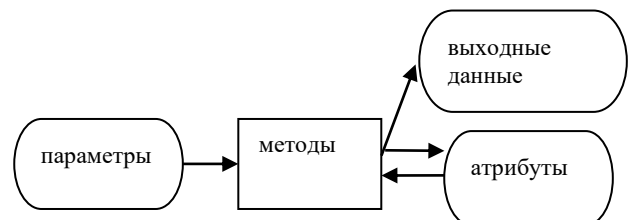


Рисунок 1 - Взаимодействие между методами, параметрами, выходными данными и атрибутами.

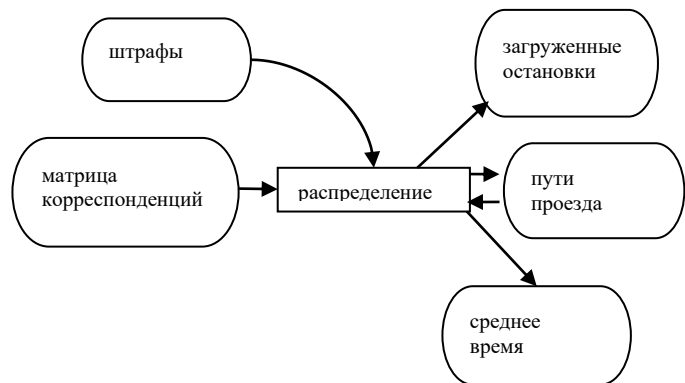


Рисунок 2 - Пример взаимодействия

Метод создает новые данные: посадка и высадка на остановочных пунктах (loaded\_stations); загрузка в пересадочных узлах; среднее время поездки.

Для каждого метода требуются свои атрибуты и программные продукты. Эти взаимодействия можно отобразить в виде ориентированного графа. При процедуре проектирования необходимо продвигаться по всей сети. Из набора доступных данных атрибуты постепенно добавляются к объекту, пока не будет достигнута цель исследования. Две основные группы методов – это методы подготовки или предварительной обработки и пошаговые методы. Последние содержат алгоритмы для реализации фактической модели транспортировки, такой как шаги распределения или назначения.

Подготовка данных является важным этапом каждого исследования. Поскольку он часто занимает много времени, использова-

ние адаптированного набора инструментов позволит значительно сократить продолжительность этой процедуры. Помимо очистки данных, которая обычно специфична для

каждого проекта и, в основном выполняется вручную, подготовка данных часто включает агрегацию и слияние сетевой информации.

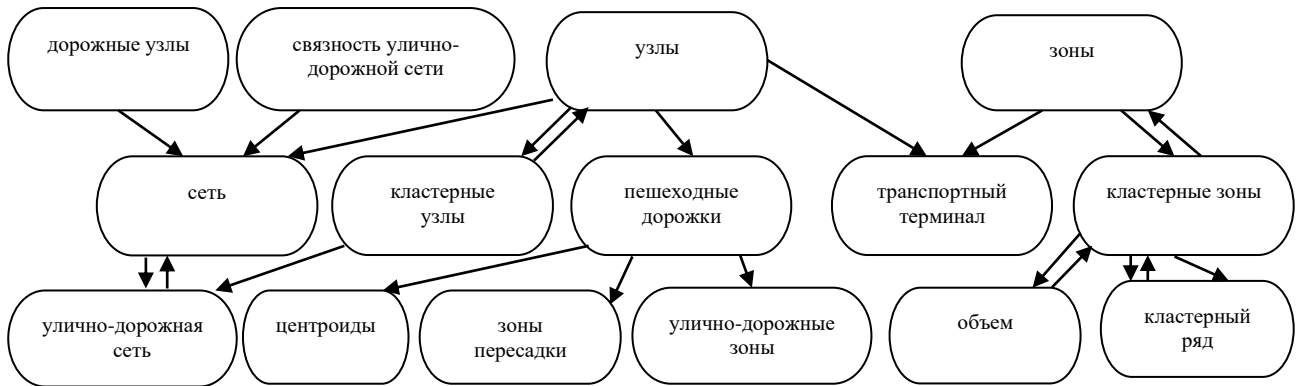


Рисунок 3 - Объектная модель панели инструментов предварительной обработки

Чтобы сократить время вычислений, необходимо уменьшить размер входных данных. Широко используя существующие алгоритмы геоматики, Quetzal позволяет строить агрегированные зоны и кластеры. Поэтому можно задавать время выполнения модели с ранних стадий исследования и автоматически определять размер входных данных соответствующим образом.

Рассматривая модель Quetzal с  $L$  связями,  $N$  узлами и  $Z$  зонами, отметим, что ее основные алгоритмы имеют следующие особенности:

1. Поиск в основном строится с применением алгоритма Дейкстры. Для одного пункта назначения алгоритм имеет сложность  $O((L + N) * \log N)$ . Для парного поиска пунктов отправления/назначения сложность составляет  $O(Z * (L + N) * \log N)$ ;

2. Многие процедуры (модальное разделение), попарно применяются к матрицам и имеют сложность  $O(Z^2)$ . Это позволяет сократить количество зон и узлов для снижения вычислительной сложности.

Первоначальное зонирование представляет собой набор полигонов, а процедура агрегации фокусирует координаты их центроидов. Они кластеризуются методом  $K$ -средних, где  $K$  – количество желаемых агрегированных зон. После вычисления кластеров центроидов полигоны объединяются для формирования агрегированного зонирования.

Кластеризация остановочных пунктов основана на том же алгоритме  $K$ -средних,

что и зональная агрегация:

- исходные остановочные пункты подаются в алгоритм  $K$ -средних;
- каждая остановка назначается одному из кластеров  $K$ .

В транзитных связях исходный остановочный пункт заменяется кластером. Расписание работы общественного транспорта и улично-дорожные сети являются ключевыми входными данными для многих исследований (информация поступает из независимых источников). Следовательно, эти данные редко связаны, и для моделирования физического взаимодействия (влияние движения индивидуальных автомобилей на скорость общественных видов транспорта) их необходимо объединить. Каждый остановочный пункт может быть соотнесен со многими узлами, и редко бывает, что ближайший из них является наиболее оптимальным. Разработанное в Quetzal решение реализовано для этой задачи. Таким образом, данные сети возможно эффективно объединять при построении модели.

Проверки целостности часто игнорируются при подготовке данных, а проблемы с согласованностью обнаруживаются только при возникновении явной ошибки. Такие проблемы очень распространены в сложных и тесно связанных данных, обрабатываемых в процессе транспортного моделирования. Quetzal встраивает инструменты для автоматизированного тестирования наиболее распространенных ошибок, встречающихся на улично-дорожных сетях и данных расписа-

ния, таких как невыпуклые сети, неполные последовательности остановочных пунктов и конфликты имен (общих для независимых объектов).

Чтобы сократить время вычислений, часто предлагается уменьшить размер входных данных. Набор инструментов Quetzal предоставляет удобные функции для быстрого и эффективного картирования любых данных на улично-дорожной сети, независимо от того, получены ли они с карты или собраны из другой модели. Количество зон и остановочных пунктов может быть впоследствии сокращено, чтобы соответствовать фактической географической точности входных данных и целевому времени выполнения модели. Обработанные стандартные данные затем используются в качестве входных при моделировании дорожного движения.

Транспортная модель представляет собой последовательность процедур, называемых «шагами». Широко известна «четырёхшаговая модель» основана на следующих методах: генерация, распределение, модальное разделение и назначение. Хотя Quetzal – это библиотека, которую можно использовать в различных типах архитектур, четырёхшаговая модель применяется для представления основных методов библиотеки; она включает:

- этап генерации, не требующий специальных методов и выполняющийся в чистом коде Python;
- распределение выполняется в выделенном модуле Quetzal [распределение];
- модальное разделение достигается с помощью модуля `logit` [logit];
- назначение является результатом перечисления путей, выполняемое алгоритмами взвешивания поиска пути, которое осу-

ществляется с помощью модулей [logit] и [pt\_assignment];

- слияния веса путей и нагрузки на соответствующей паре корреспонденций.

Четырёхшаговая модель способ понять последовательность выборов пути проезда, однако реализация модели часто более сложна: циклы могут быть построены с помощью методов, два шага могут быть объединены в уникальную процедуру, а сами шаги являются продуктом нескольких подшагов, таких как перечисление и оценка в случае шага назначения. Чтобы реализовать модель, необходимо найти логическую последовательность процедур в объектной модели. Когда нет необходимости в цикле между методами, ее можно представить в виде дерева (ациклического направленного графа).

Алгоритм «четырёхшаговой модели» сводится к следующему:

1. Матрица спроса формируется путем генерации и распределения.
2. Затем матрица используется в качестве входных данных при поиске пути (назначении).
3. Одновременно выполняются поиски путей общественного транспорта.
4. Пути, перечисленные двумя поисками, взвешиваются с помощью модуля `logit`.
5. Объемы перевозок назначаются общественному транспорту.

В заключении хочется отметить, что библиотеку Quetzal можно использовать как набор инструментов или самодостаточную систему моделирования. Это эффективный инструмент с открытым исходным кодом особенно актуален в контексте открытых данных.

## СПИСОК ЛИТЕРАТУРЫ

1. **Chasserieau, Q. Goix, R.** Design and use of Quetzal, an Open-Source Python Library for Transport Modeling / Q. Chasserieau, R. Goix, European Transport Conference, 2019. Dublin, Ireland.
2. **Крипак, М.Н.** Оценка состояния улично-дорожной сети крупного города /М.Н. Крипак, О.А. Лебедева //Современные технологии. Системный анализ. Моделирование. 2016. № 3 (51). С. 171-174.
3. **Лебедева, О.А.** Сравнительный анализ методов решения транспортных за-

дач при оптимальном планировании перевозочного процесса /О.А. Лебедева., В.Е. Гозбенко, А.А. Пыхалов, Ю.Ф. Мухопад // Современные технологии. Системный анализ. Моделирование. 2020. № 3 (67). С. 134-139.

4. **Лебедева, О.А.** Транспортное планирование в рамках интеграции моделей землепользования и оценки спроса / О.А. Лебедева. Сборник научных трудов Ангарского государственного технического университета. 2022. № 19. С. 103-107.