



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ

« Информатика и системы управления»

КАФЕДРА

«Программное обеспечение ЭВМ и информационные технологии»

## ОТЧЕТ ПО ПРЕДДИПЛОМНОЙ ПРАКТИКЕ

Студент Иванов Всеволод Алексеевич  
*фамилия, имя, отчество*

Группа ИУ7-82Б

Тип практики производственная

Название предприятия МГТУ им. Н. Э. Баумана

Студент \_\_\_\_\_ / \_\_\_\_\_  
(подпись, дата) (Фамилия И.О.)

Руководитель практики  
от предприятия \_\_\_\_\_ / \_\_\_\_\_  
(подпись, дата) (Фамилия И.О.)

Руководитель практики  
от МГТУ им. Н. Э. Баумана \_\_\_\_\_ / \_\_\_\_\_  
(подпись, дата) (Фамилия И.О.)

Рекомендуемая оценка \_\_\_\_\_

2022 г.

Кафедра «Программное обеспечение ЭВМ и информационные технологии» (ИУ7)

## **ЗАДАНИЕ**

### **на прохождение производственной практики**

на предприятии \_\_\_\_\_ МГТУ им. Н. Э. Баумана

---

Студент Иванов Всеволод Алексеевич, ИУ7-82Б

---

(фамилия, имя, отчество; инициалы; индекс группы)

Во время прохождения производственной практики студент должен:

1. В первой части подробно описать шаги разрабатываемого метода планирования грузоперевозок с учётом выбранного критерия оптимизации. Разработать структуру программного приложения, определить требования к формату входных и выходных данных. Описать компоненты программы и их взаимодействие, а также ключевые структуры данных, которые будут использоваться при реализации разработанного алгоритма.
2. Во второй части необходимо обосновать выбор программных средств реализации метода оптимизации планирования грузоперевозок. Разработать графический интерфейс пользователя для ввода данных и отображения результатов работы программы. Привести примеры работы программы. Описать используемые методы тестирования программного обеспечения и привести его результаты.
3. В третьей части необходимо исследовать зависимость результатов работы метода от различных параметров системы.

Дата выдачи задания « \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

Руководитель практики от кафедры \_\_\_\_\_ / Строганов Ю. В.  
(подпись, дата) (Фамилия И.О.)

Студент \_\_\_\_\_ / \_\_\_\_\_  
(подпись, дата) (Фамилия И.О.)

# СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ</b>	<b>4</b>
<b>1 Конструкторская часть</b>	<b>5</b>
1.1 IDEF0 диаграмма . . . . .	5
1.2 Схемы алгоритмов . . . . .	6
1.2.1 Основной алгоритм . . . . .	6
1.2.2 Вычисление начального баланса . . . . .	7
1.2.3 Составление опорного плана . . . . .	8
1.2.4 Оптимизация плана . . . . .	8
1.2.5 Формирование расписания . . . . .	10
1.3 Схема сущностей транспортной системы . . . . .	11
1.4 Формат входных данных . . . . .	13
1.5 Формат выходных данных . . . . .	14
<b>2 Технологическая часть</b>	<b>15</b>
2.1 Выбор средств программной реализации . . . . .	15
2.2 Используемые библиотеки . . . . .	15
2.3 Интерфейс программы . . . . .	16
2.4 Результаты работы программы . . . . .	18
2.5 Тестирование программы . . . . .	21
<b>3 Исследовательская часть</b>	<b>23</b>
3.1 Постановка задачи . . . . .	23
3.2 Генерация транспортной системы . . . . .	23
3.3 Проведение экспериментов . . . . .	24
3.3.1 Характеристики вычислительной техники . . . . .	24
3.3.2 Исследование работы алгоритма . . . . .	24
3.3.3 Сравнение плана до и после оптимизации . . . . .	25
3.3.4 Определение ограничений программы . . . . .	26
3.3.5 Выявление закономерностей работы системы . . . . .	27
<b>ЗАКЛЮЧЕНИЕ</b>	<b>30</b>
<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ</b>	<b>31</b>

## ВВЕДЕНИЕ

В настоящее время активно идёт процесс замены интеллектуального труда человека на специализированные программы. Зачастую автоматизации подвергаются лишь отдельные эпизоды работы человека, отличающиеся однотипностью действий, так как они относительно просты в программной реализации и позволяют повысить производительность работника, предоставив ему больше времени для решения более сложных задач.

Примером подобной подзадачи является планирование доставки в профессии логиста. Решение этой проблемы должно учитывать достаточно большое количество факторов. Подобные задачи удаётся решать при помощи программ, так как с использованием математических методов оптимизации они способны принимать наиболее выгодные решения, с точки зрения выделенных критериев, что не всегда способен сделать человек.

Целью преддипломной практики является разработка метода для оптимизации планирования грузоперевозок, ранее изученного в рамках научной исследовательской работы.

Выделены следующие задачи:

- формализовать задание, определить необходимый функционал программного обеспечения;
- определить набор необходимых данных и способ их хранения;
- разработать и протестировать программу в соответствии с выделенным функционалом;
- провести экспериментальную проверку качества функционирования разработанного метода.

# 1 Конструкторская часть

В данном разделе будут представлена IDEF0 диаграмма метода, схемы алгоритма и структура приложения. Описан формат входных и выходных данных.

## 1.1 IDEF0 диаграмма

На рисунках 1.1 – 1.2 представлены IDEF0 диаграммы.



Рисунок 1.1 – IDEF0, контекстная диаграмма A0

Разрабатываемый метод состоит из нескольких этапов, описанных на рисунке 1.2 с помощью IDEF0 диаграммы.

В первую строится опорный план грузоперевозок по модифицированному методу минимального элемента. Далее для него применяется также изменённый под условия решаемой задачи метод потенциалов. Результатом конечного числа итераций данного метода становится оптимальный набор маршрутов пе-

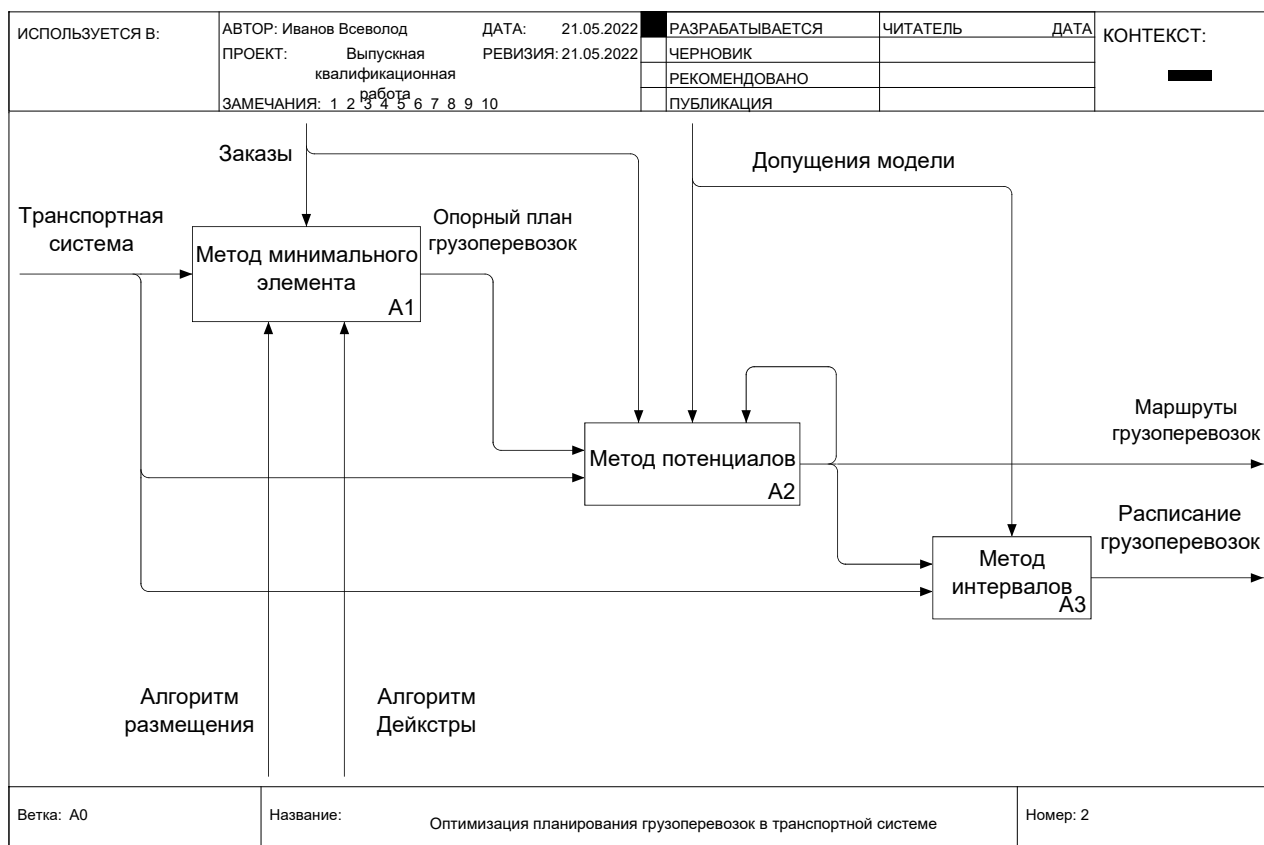


Рисунок 1.2 – IDEF0 диаграмма, декомпозиция A0

ревозки. По нему составляется расписание с использованием модифицированного метода интервалов.

## 1.2 Схемы алгоритмов

На рисунках 1.3 - 1.8 представлены схемы ключевых алгоритмов программы.

### 1.2.1 Основной алгоритм

Весь алгоритм оптимизации изображён на рисунке 1.3. Его можно выделить следующие основные этапы:

- 1) Вычисление начального баланса продуктов.
- 2) Составление опорного плана.
- 3) Оптимизация плана.
- 4) Формирование расписания.

Рассмотрим упомянутые этапы подробнее в следующих схемах.

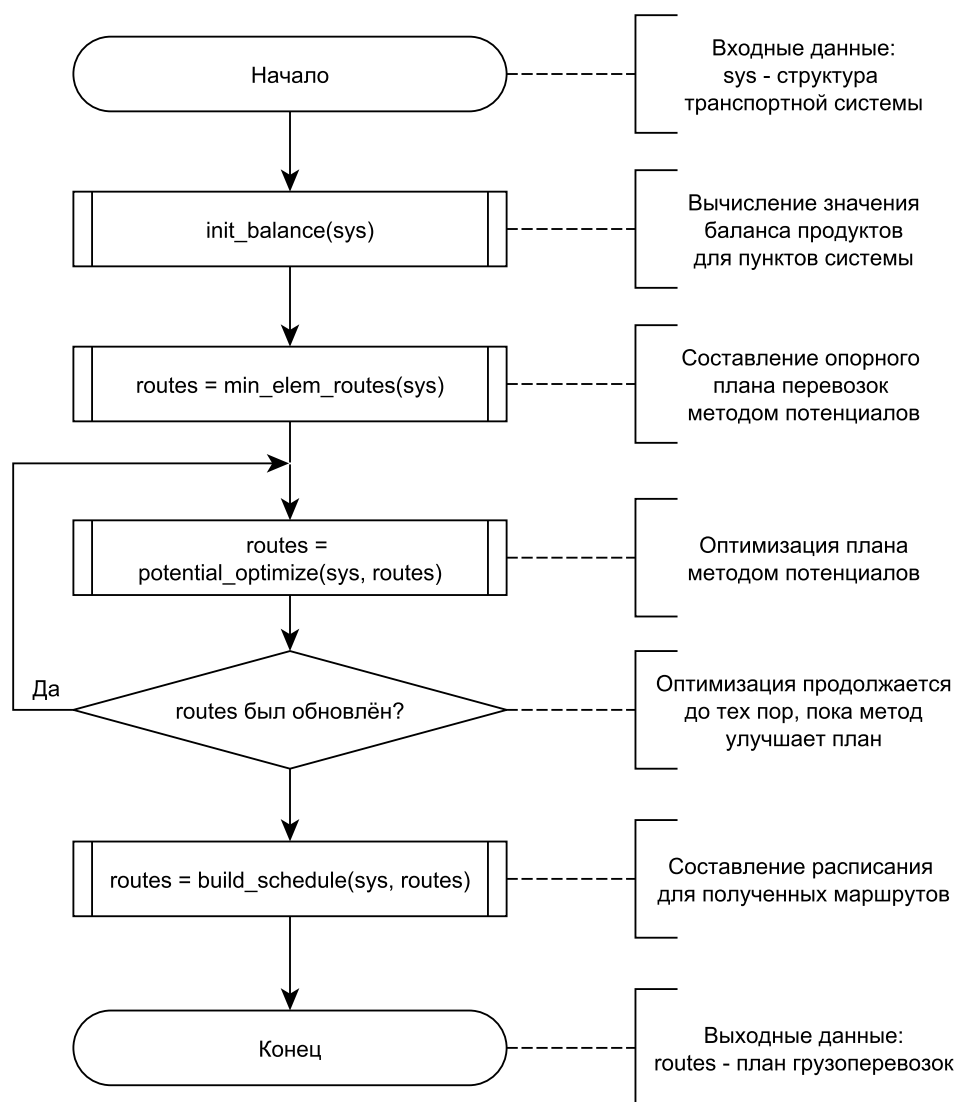


Рисунок 1.3 – Схема общего алгоритма программы

### 1.2.2 Вычисление начального баланса

При вычислении баланса, схема которого изображена на рисунке 1.4, для всех складов значение выставляется как количество хранимой продукции, для всех потребителей число заказанных тар продуктов со знаком минус. Баланс стоянки нулевой. Данный этап нужен для проверки соблюдения ограничения количества продукции в различных пунктах.

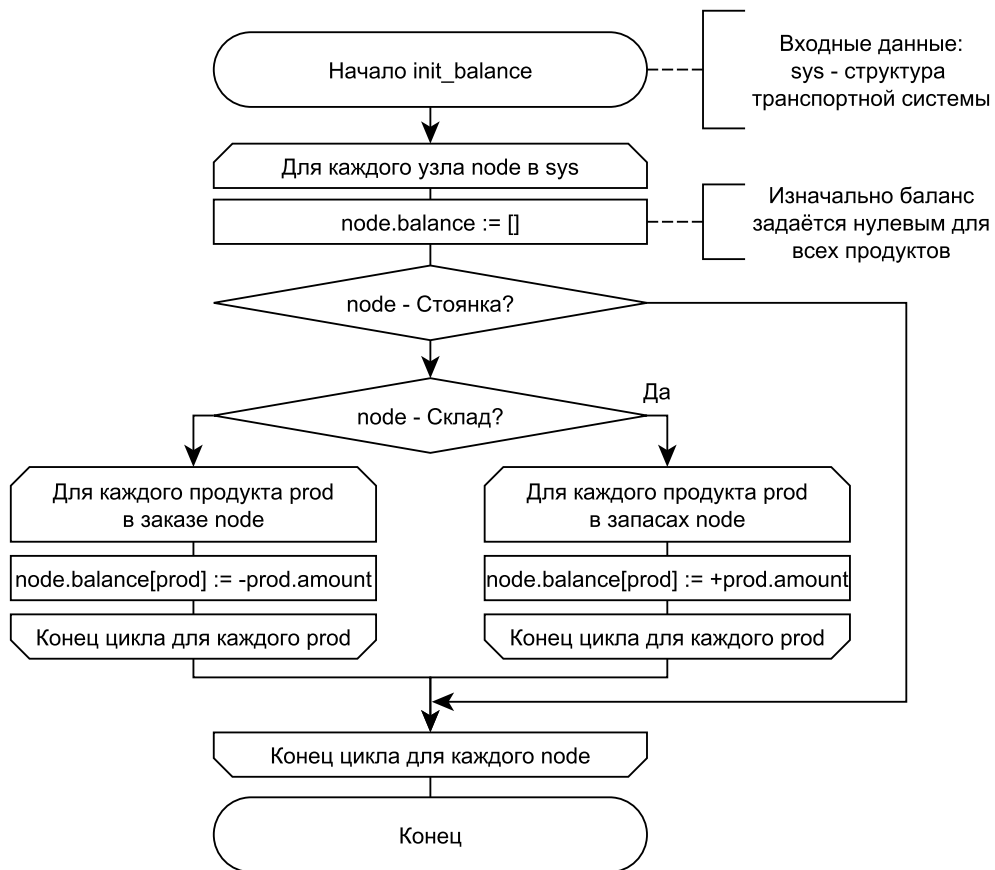


Рисунок 1.4 – Схема алгоритма вычисления начального баланса

### 1.2.3 Составление опорного плана

На этом этапе формируется опорный план грузоперевозки. Для этого все маршруты, ведущие к каждому потребителю через склад по вычисленному кратчайшему пути, просматриваются в соответствии с методом минимального элемента — в порядке возрастания их стоимости.

### 1.2.4 Оптимизация плана

Рисунки 1.6 - 1.7 описывают алгоритм оптимизации плана. После вычисления потенциалов начинается рассмотрение узлов, для которых обнаружены невязки. Анализируется возможность замены перевозки текущих маршрутов-поставщиков альтернативными, завершающимися на смежных пунктах. Другие маршруты продлеваются по очереди убывания их выгодности для данного пункта и по мере ограничений.



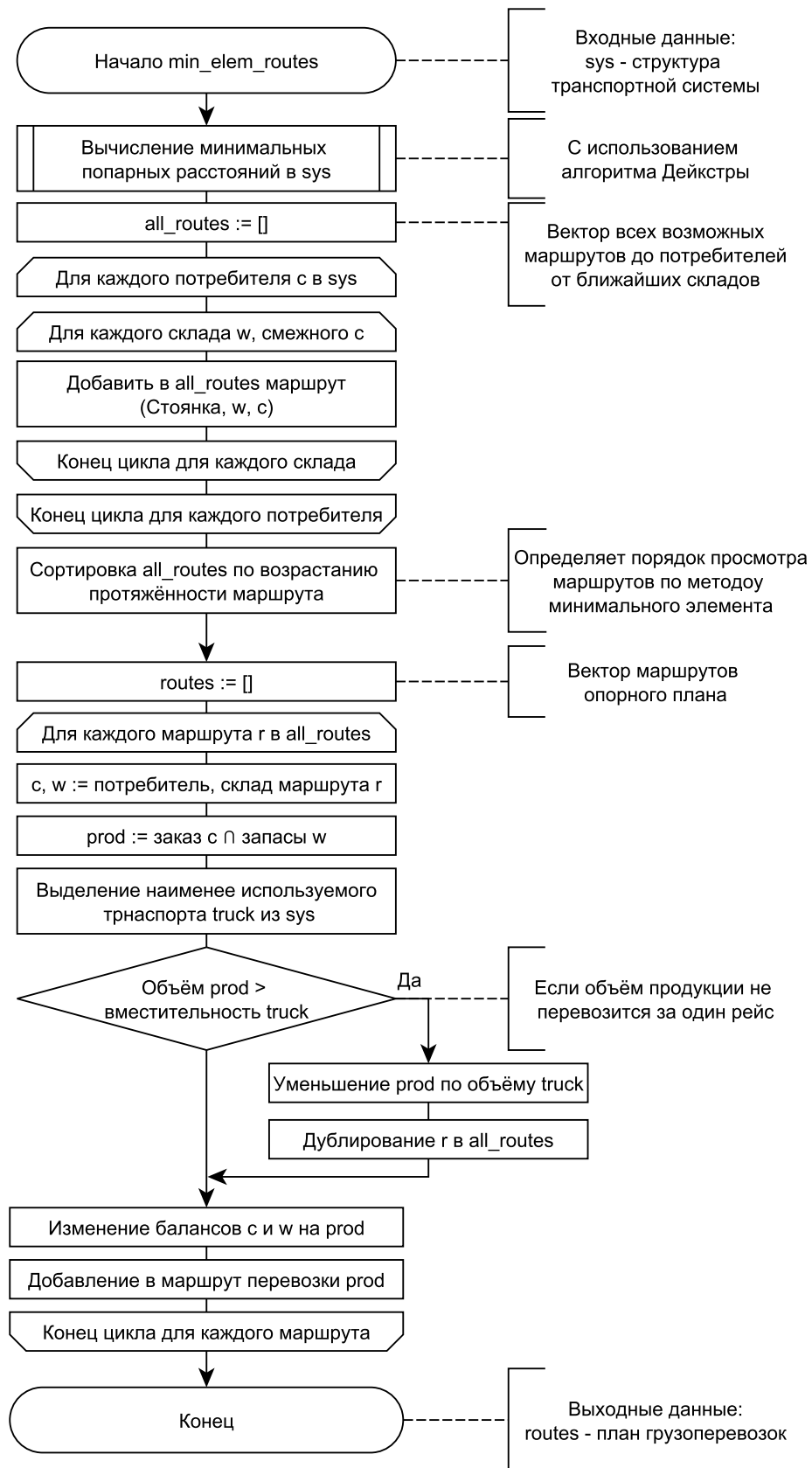


Рисунок 1.5 – Схема алгоритма минимального элемента

Если после замены оригинального маршрута в данном пункте функция стоимости для плана становится меньше, то изменения принимаются за новый опорный план. Иначе, поиск оптимизации продолжается далее.

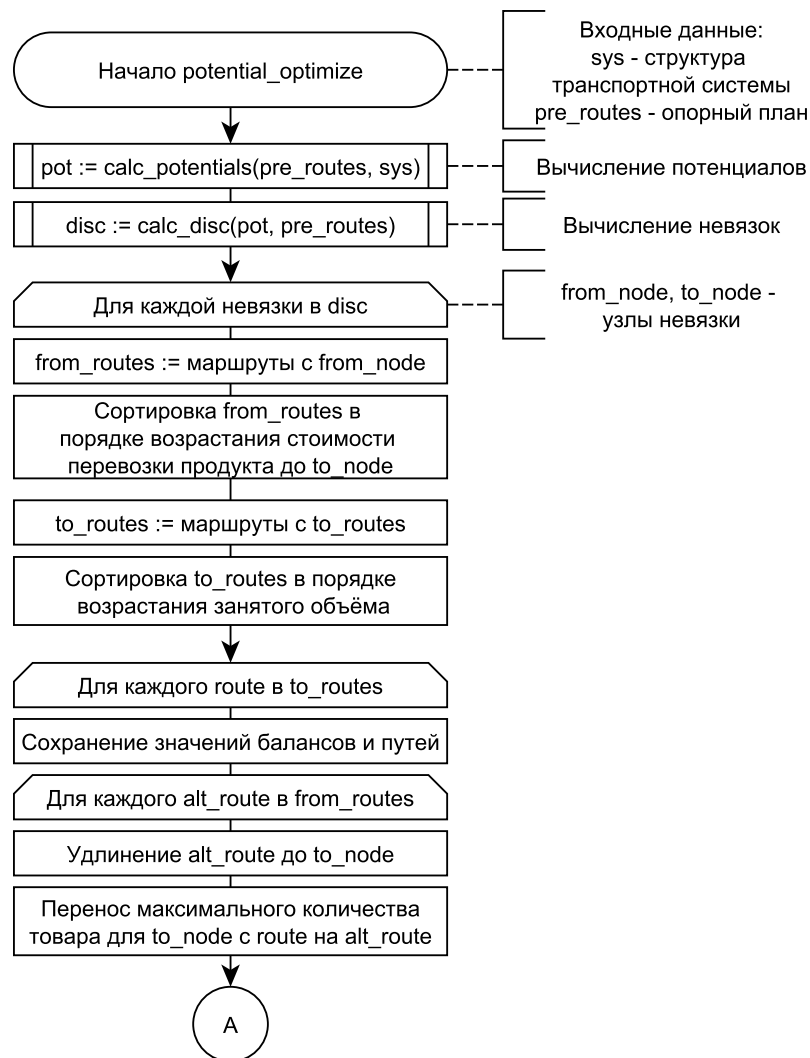


Рисунок 1.6 – Схема алгоритма оптимизации методом потенциалов

### 1.2.5 Формирование расписания

Рисунок 1.8 описывает алгоритм формирования расписания. Изначально все маршруты выставляются на начало работы. В соответствии с описанными в системе продолжительностями проезда по дорогам и временами обслуживания в пунктах, формируется время прибытия и отбытия из каждого пункта маршрута.

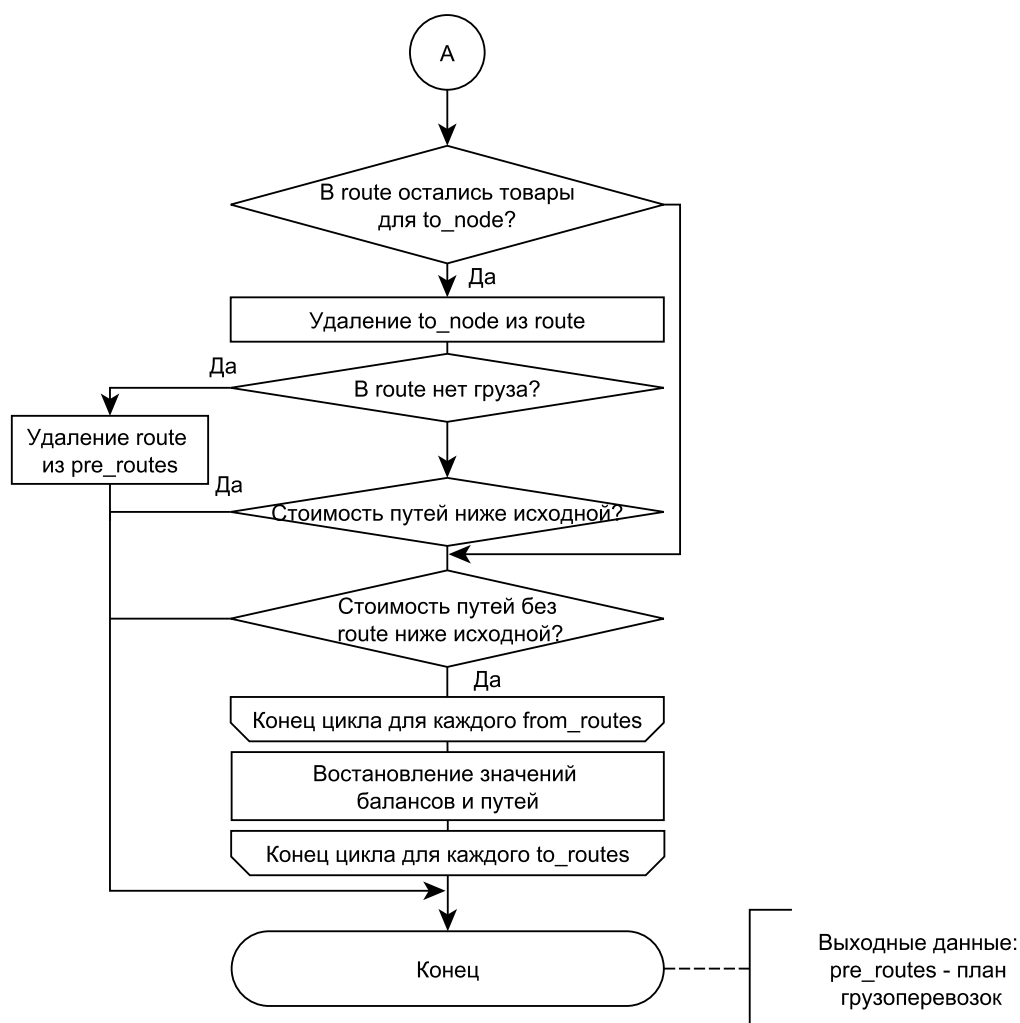


Рисунок 1.7 – Схема алгоритма оптимизации методом потенциалов (продолжение)

Далее осуществляется поочерёдное рассмотрение расписаний маршрутов, если оно не пересекается с уже принятыми, то маршрут добавляется в общее расписание с закреплением транспорта. Пересечением называется остановка на одном пункте в одинаковый промежуток времени. Величиной пересечения является общее время нахождения на пункте.

### 1.3 Схема сущностей транспортной системы

В соответствии с ранее описанной математической моделью, в системе должны быть представлены сущности, изображённые на рисунке 1.9 ER-диаграммой.

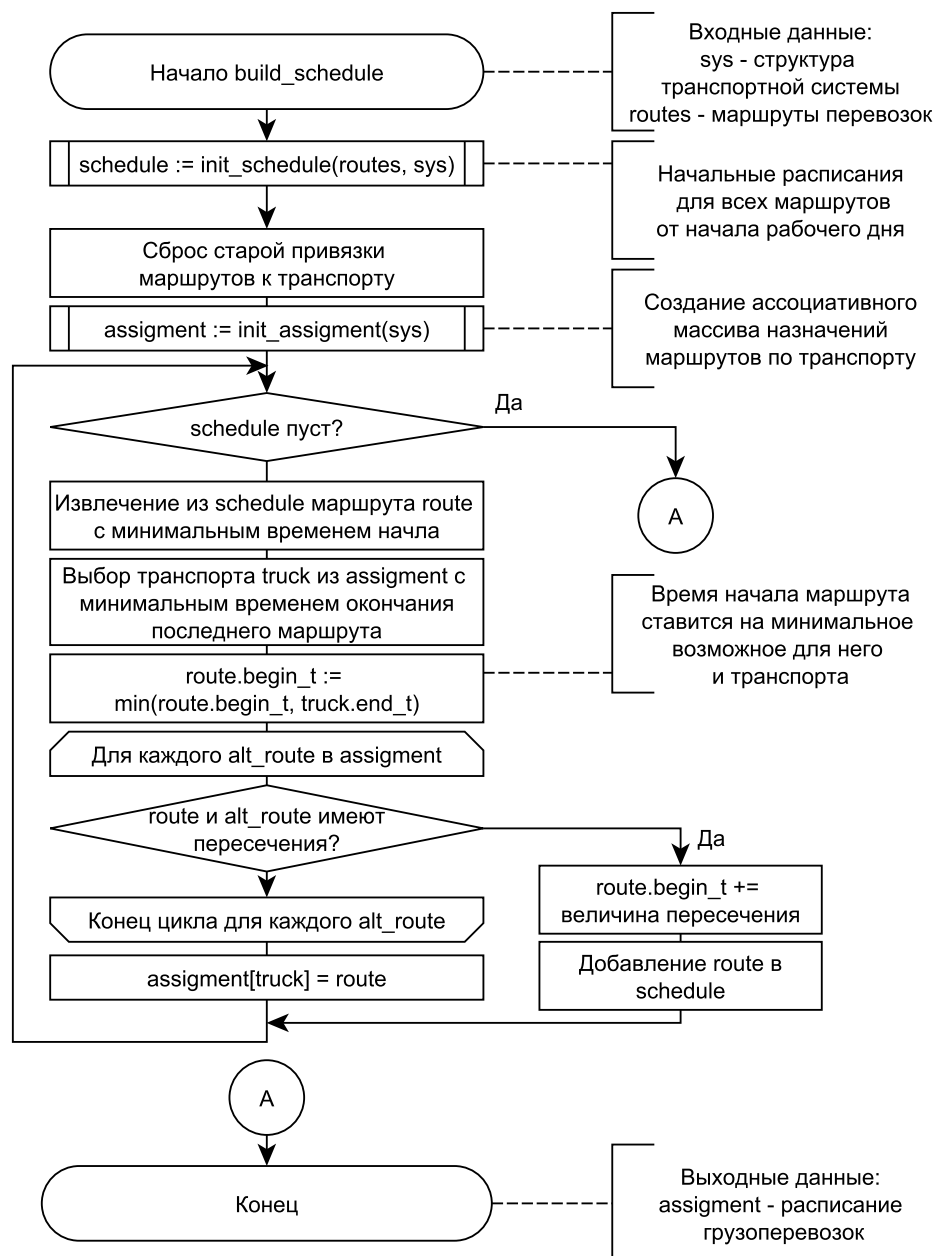


Рисунок 1.8 – Схема алгоритма составления расписания



Рисунок 1.9 – ER-диаграмма

#### 1.4 Формат входных данных

В качестве входных данных выступает описание транспортной системы. Оно должно содержать в себе следующую информацию:

- Список всех пунктов маршрута.
- Вместительность (в м<sup>3</sup>) и количество транспорта.
- Список запасов складов. Каждый склад описывается списком продуктов (название и количество тар).
- Список заказов потребителей – аналогично запасам складов.
- Список дорог: два связанных пункта, расстояние (в км.), время проезда (в мин.).
- Объём одной тары в (в м<sup>3</sup>).
- Время начала и завершения рабочего дня.

## **1.5 Формат выходных данных**

В качестве выходных данных выступает описание плана грузоперевозок. Оно состоит из списка маршрутов, каждый из которых должен содержать в себе следующую информацию:

- Последовательность посещения пунктов.
- Номер привязанного транспортного средства.
- Список товаров, загружаемых или выгружаемых на каждом пункте.
- Список времени прибытия и отбытия в каждый пункт маршрута.

### **Вывод**

В данном разделе были представлены IDEF0 и ER диаграммы, схемы алгоритмов. Описан формат входных и выходных данных программы.

## **2 Технологическая часть**

В данном разделе будет представлено обоснование выбора языка и среды программирования, описан интерфейс и применяемые методы тестирования программы.

### **2.1 Выбор средств программной реализации**

В качестве языка программирования был выбран Python 3, ввиду следующих факторов.

- За время обучения был накоплен существенный опыт в использовании данного средства, что позволит сократить время разработки программы.
- Язык поддерживает объектно-ориентированный подход, что полезно при структурировании большого количества схожих объектов, которые были выделены ранее.
- Наличие библиотек для создания графического интерфейса, визуализации графов и временных диаграмм, которые необходимы для более наглядной демонстрации работы программы.

В качестве среды разработки был выбран PyCharm по следующим причинам.

- Данная IDE предоставляется бесплатно для пользования в учебном заведении[7].
- Имеется значительный опыт в использовании данной среды разработки.
- Представлен удобный набор инструментов для написания, тестирования и отладки кода.

### **2.2 Используемые библиотеки**

На выбранном языке Python написаны библиотеки, позволяющие упростить визуализацию данных. Перечислим те, которые были использованы при разработке программы.

При разработке графического интерфейса пользователя использовался PyQt5. Qt является популярным графическим фреймворком[8]. Наличие среды

для разработки интерфейсов Qt Designer значительно упрощает работу с данной библиотекой при создании GUI.

При создании графов применялась библиотека `networkx`. Она позволяет генерировать различные типы случайных графов, что может быть использовано при тестировании и исследовании реализованной программы для создания случайных транспортных систем. Также имеется возможность привязки необходимой информации непосредственно к узлам и рёбрам графов.

Для визуализации временной диаграммы маршрутов и графа транспортной системы использовалась библиотека `plotly`[10].

### 2.3 Интерфейс программы

Для удобства задания и изменения параметров системы был разработан графический интерфейс, представленный на рисунках 2.1 – 2.3.

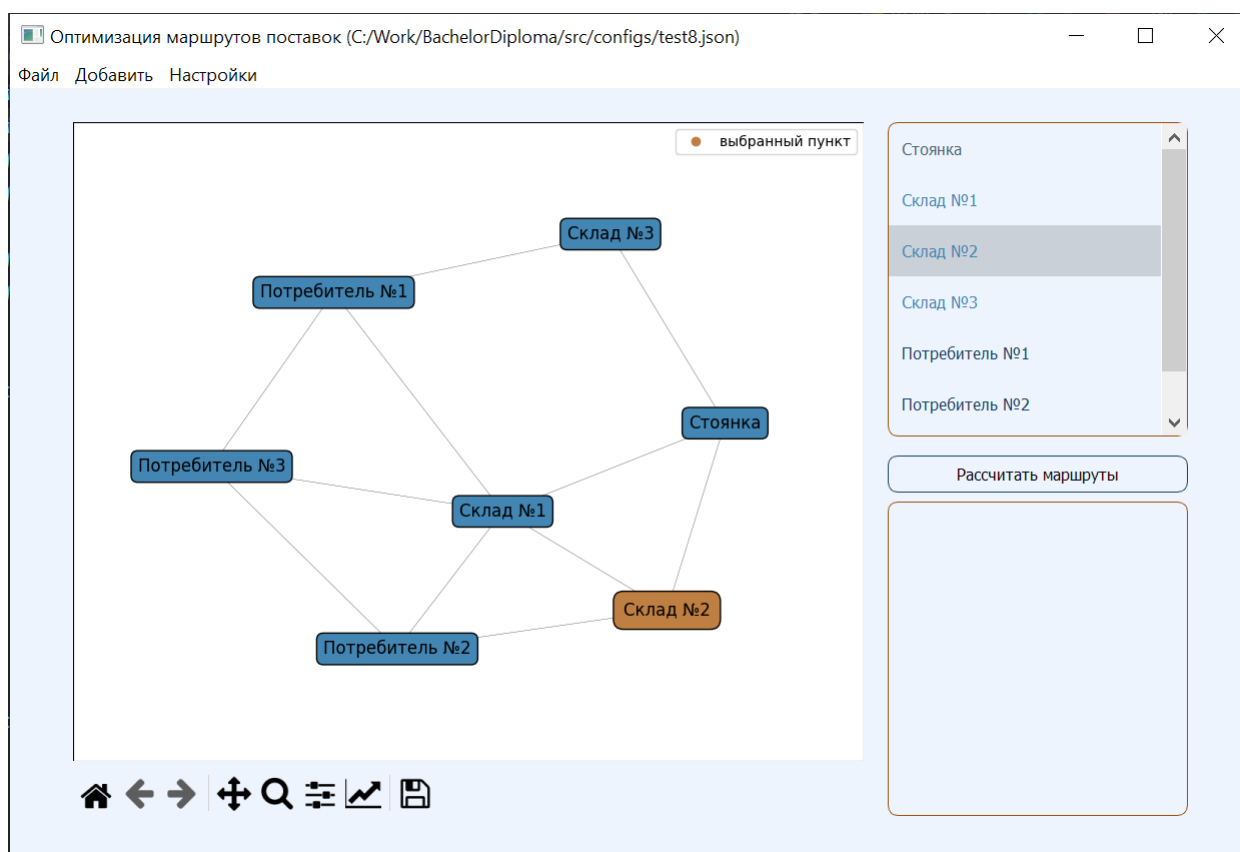


Рисунок 2.1 – Интерфейс главного окна программы

В главном окне программы происходит отображение имеющихся пунктов



транспортной системы, а также её схематичной визуализации в виде графа. В данном окне пользователь может совершить следующие действия.

- Перейти к окну редактирования параметров пункта.
- Добавить новый пункт в систему.
- Импортировать транспортную систему из указанного файла.
- Экспортировать текущее состояние системы в файл.
- Настроить прочие параметры системы.
- Вызвать построение плана маршрутов.
- Посмотреть график маршрутов на временной диаграмме.

На рисунке 2.2 изображено окно изменения параметров стоянки. В данном окне пользователь может совершить следующие действия.

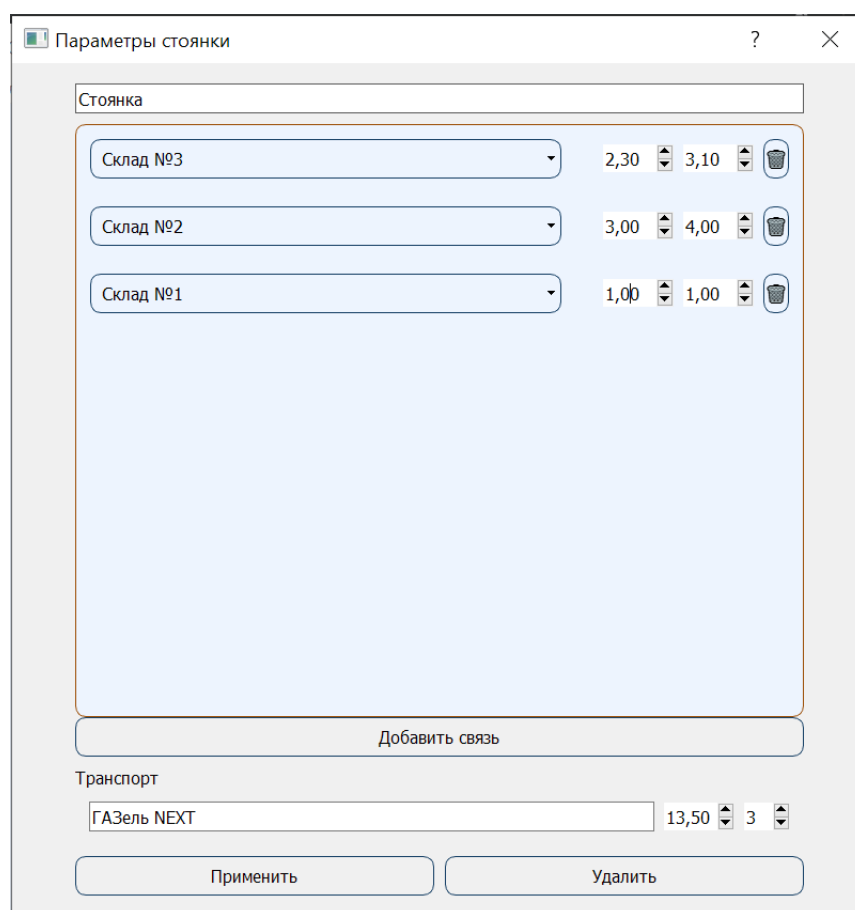


Рисунок 2.2 – Интерфейс окна параметров стоянки

- Изменить название пункта.

- Редактировать существующие дороги до других пунктов, удалять и добавлять новые.
- Изменить название, вместительность и количество машин в автопарке.
- Применить изменения.
- Удалить сущность из системы.

Похожий интерфейс имеет окно изменения параметров склада и потребителя, изображённое на рисунке 2.3. Оно позволяет дополнительно управлять набором продуктов, составляющих заказ или запас в пункте.

The screenshot shows a window titled 'Параметры склада' (Warehouse Parameters). At the top, there is a text field labeled 'Склад №2'. Below it, there is a list of items with their respective quantities and a 'Добавить связь' (Add link) button. The items are:

Item	Quantity
Потребитель №2	1,00
Склад №1	1,00
Стоянка	3,00

Below the list, there is a 'Добавить связь' button. Underneath, there is a section for adding products, with a 'Добавить продукт' button. The products listed are:

Product	Quantity
кола	10
кофе	20

At the bottom, there are two buttons: 'Применить' (Apply) and 'Удалить' (Delete).

Рисунок 2.3 – Интерфейс окна параметров склада

## 2.4 Результаты работы программы

Графический интерфейс также визуализирует результаты составления и оптимизации плана грузоперевозок, что отражают рисунки 2.4 – 2.9. На них изображены результаты работы метода на сети малого и большого размеров.

Приведены визуализации и отчёты маршрутов, временная шкала графика перемещения транспорта по узлам сети.

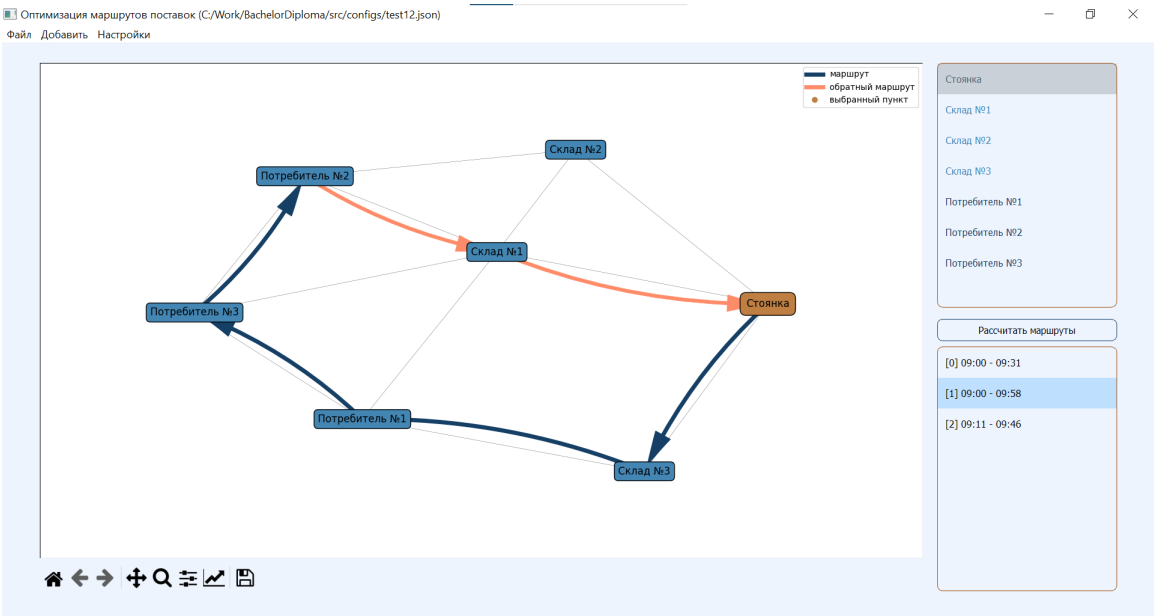


Рисунок 2.4 – Пример 1. Визуализация маршрута

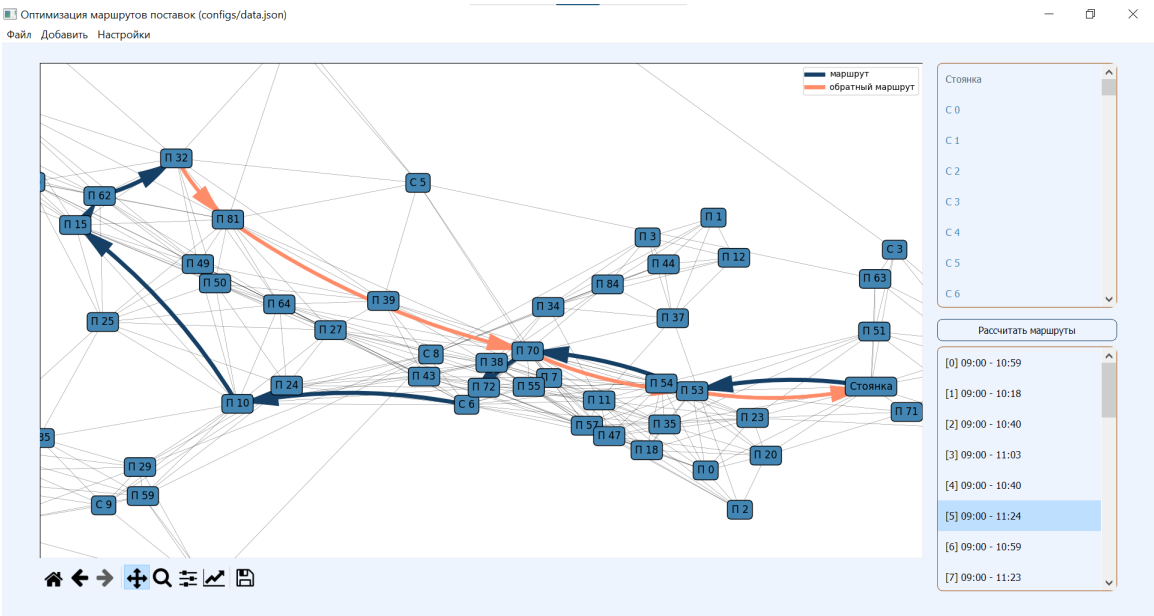


Рисунок 2.5 – Пример 2. Визуализация маршрута

Маршрут перевозки			?	×
[1] ГАЗель NEXT				
Стоянка			09:00 - 09:02	^
Склад №3			09:05 - 09:15	
<-	кола	10		
<-	шоколад	20		
Потребитель №1			09:16 - 09:28	
->	кола	10		
->	шоколад	10		
Потребитель №3			09:29 - 09:41	
->	шоколад	5		
Потребитель №2			09:42 - 09:54	
->	шоколад	5		
Склад №1			09:55	
Стоянка			09:56 - 09:58	▼

Рисунок 2.6 – Пример 1. План маршрута

Маршрут перевозки			?	×
[5] Нива				
П 68			09:27	^
С 7			09:27 - 09:37	
<-	пряники	43		
П 22			09:40 - 09:52	
->	пряники	8		
П 9			09:58 - 10:10	
->	пряники	8		
П 74			10:10 - 10:22	
->	пряники	9		
П 80			10:25 - 10:37	
->	пряники	10		
П 68			10:45 - 10:57	
->	пряники	8		
П 14			11:02	
П 46			11:03	▼

Рисунок 2.7 – Пример 2. План маршрута

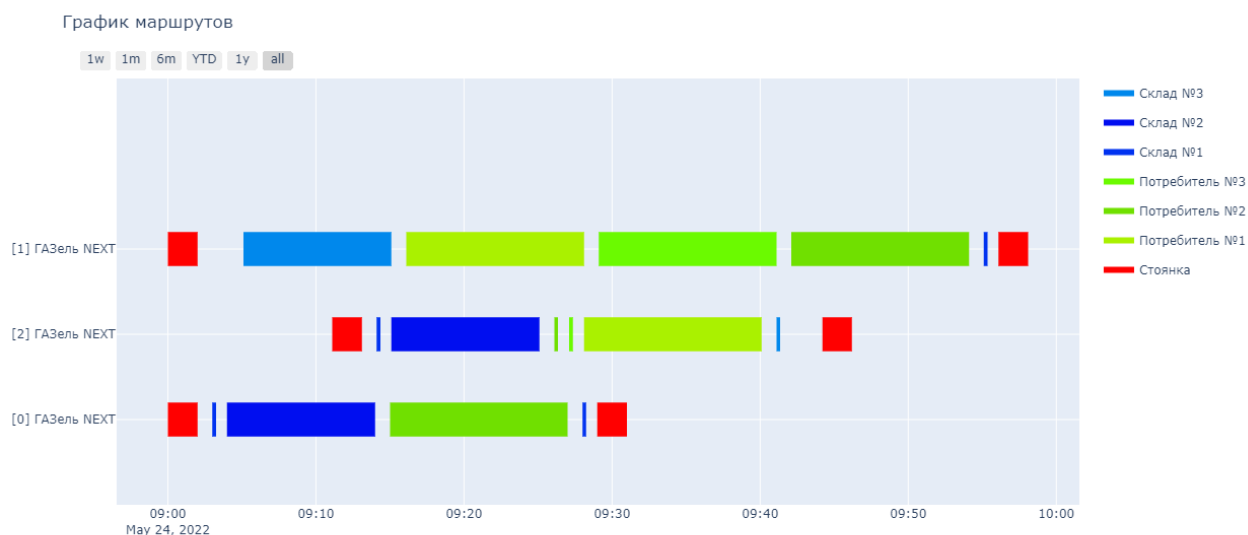


Рисунок 2.8 – Пример 1. Временной график грузоперевозок

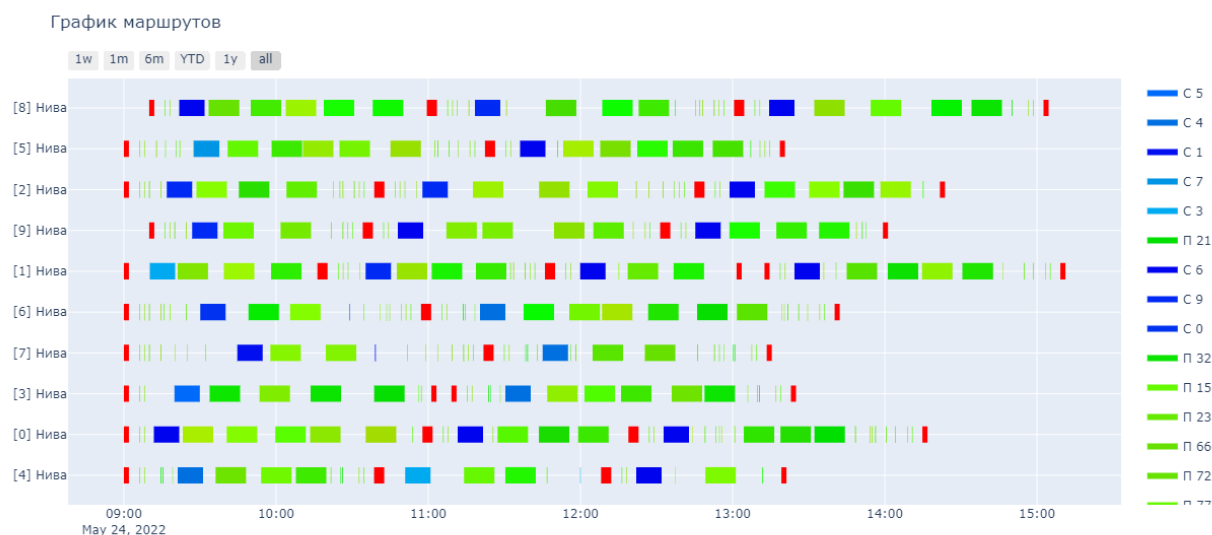


Рисунок 2.9 – Пример 2. Временной график грузоперевозок

## 2.5 Тестирование программы

Для поддержания работоспособности программы в процессе её разработки и было использовано автоматическое тестирование кода. Для этого была использован встроенный модуль unittest[11].

Тестирование осуществлялось по методу чёрного ящика. В качестве входных данных выступали файлы с транспортными системами, подразумевающие различные сценарии оптимизации. В ходе одного теста проверялась корректность завершения работы метода, а также следующие выходные данные:

- количество и стоимость маршрутов в плана (сравнивается с вручную вычисленным значением для данного теста);
- удовлетворение спроса потребителей;
- соблюдение ограничений запасов продукции и вместительности транспорта;
- целостность и замкнутость маршрутов.

В качестве негативных тестов на вход программы подавались примеры систем, в которых присутствуют следующие нарушения:

- суммарный объём заказываемого товара больше имеющихся запасов;
- отдельный пункт не имеет дорог;
- граф пунктов содержит более одной компоненты связности;
- стоянка не имеет транспорта;

Ожидаемое поведение в данном случае – сообщение о неверных входных данных.

В окончательной версии программы все тесты завершаются корректно.

## **Вывод**

Результатом технологической части стал выбор средств реализации программы, а также набора необходимых библиотек. В рамках созданного приложения был реализован ранее описанный метод оптимизации. Продемонстрирован и описан интерфейс, а также примеры работы программы. Разработанное программное обеспечение покрыто тестами.

### **3 Исследовательская часть**

В данном разделе будет проведено исследование разработанной программы. Будет верифицирована её работоспособность в различных конфигурациях системы. Также будут выявлены ограничения применимости приложения. Результаты всех экспериментов будут сопровождены выводами.

#### **3.1 Постановка задачи**

С целью проверки работоспособности разработанного метода требуется сравнить стоимости начального и оптимизированного плана при различных размерностях транспортной системы.

Также необходимо произвести замеры среднего времени работы программы при разном количестве пунктов маршрутов для определения ограничений его применимости.

С целью определения закономерностей работы метода будут проведены исследования зависимости общей стоимости маршрутов от следующих параметров:

- 1) вместительность грузовика;
- 2) средняя удалённость пунктов маршрута от стоянки;

#### **3.2 Генерация транспортной системы**

Для проведения экспериментов необходимо иметь описания большого количества (не менее ста) транспортных систем. Собрать большое количество реальных данных крайне затруднительно. Поэтому более предпочтительной является генерация случайной транспортной системы.

При генерации сети в первую очередь с помощью библиотеки `networkx` создаётся случайный граф. Полученные позиции вершин и их наличие рёбер между ними используется для создания дорог в транспортной системе. После этого узлам графа случайным образом сопоставляется роль пункта, набор продуктов в заказе или запасах. Пример использования сгенерированной системы приведён на рисунке 2.5.

### 3.3 Проведение экспериментов

#### 3.3.1 Характеристики вычислительной техники

Эксперименты проводились на персональном компьютере со следующими характеристиками.

- Операционная система — Windows 10, 64-разрядная.
- Процессор — AMD Ryzen 5 5500U with Radeon Graphics 2.10 GHz.
- Оперативная память — 8,00 ГБ.

#### 3.3.2 Исследование работы алгоритма

Метод потенциалов является многоитерационным, поэтому имеется возможность пронаблюдать изменение системы на каждом шаге и проанализировать за счёт чего достигается более оптимальный план.

В данном эксперименте будет исследована зависимость стоимости маршрутов, их количества, средней длине и средней загруженности от текущей итерации оптимизации. Результат проведённого эксперимента изображён в виде графиков на рисунке 3.1. В исследовании использовались случайные транспортные системы с размерностями в 100 и 200 пунктов маршрута.

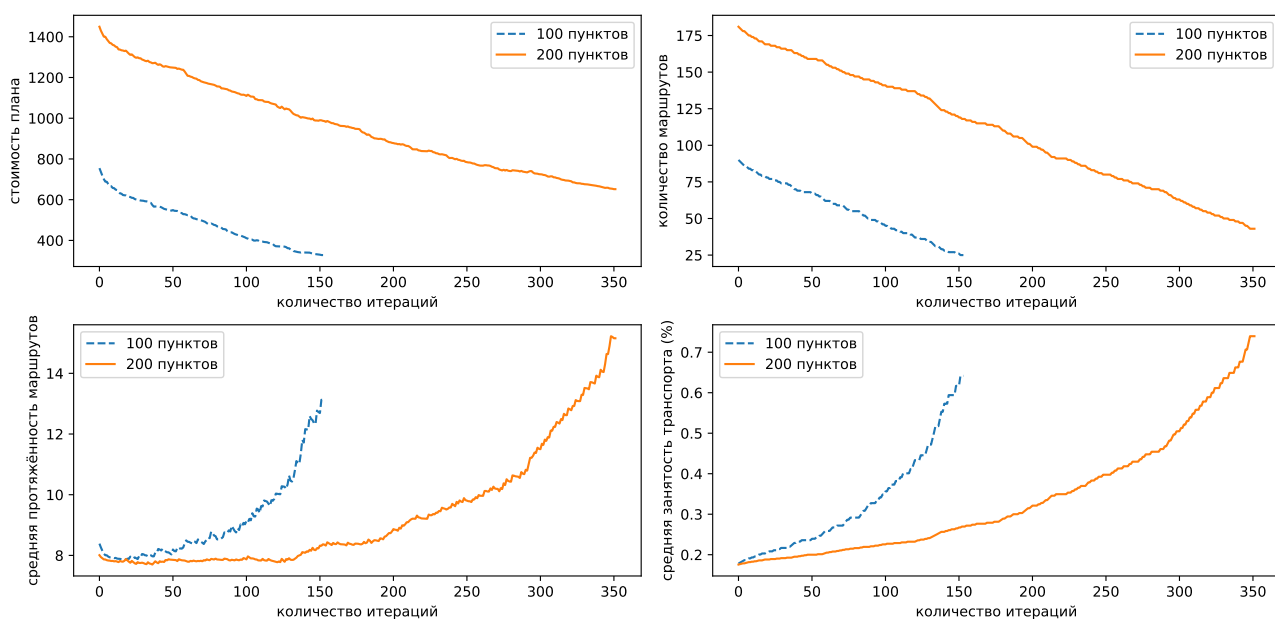


Рисунок 3.1 – Исследование работы алгоритма



Из данного графика можно сделать вывод о том, что главным образом снижение стоимости достигается за счёт снижения общего числа маршрутов посредством продления и распределения груза на другие маршруты. Также можно отметить, что данные закономерности не зависят от размера транспортной сети.

### 3.3.3 Сравнение плана до и после оптимизации

Главным показателем корректности реализованного метода является наличие оптимизации. Оно выражается в том, что конечный план должен обладать меньшей суммарной стоимостью для всех маршрутов по сравнению с начальным, опорным планом.

Чтобы установить выполнение данного условия в общем случае, в данном эксперименте были использованы случайно созданные транспортные сети размером от 20 до 180 пунктов. Измерения для каждой размерности проводятся многократно для усреднения полученных значений. Результат проведённого эксперимента изображён на графике, на рисунках 3.2 – 3.3.

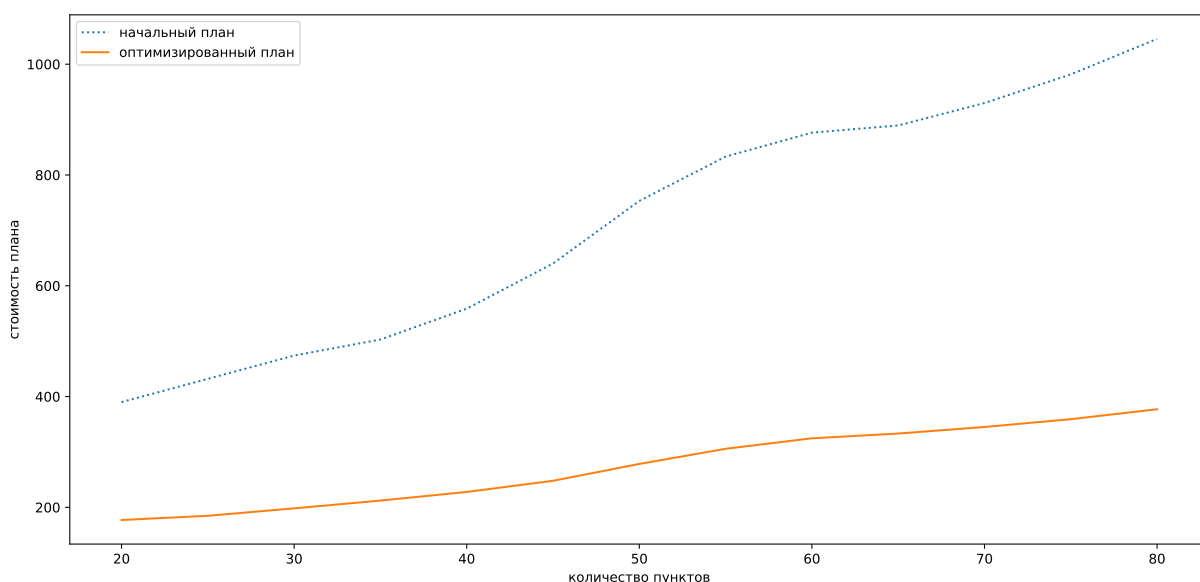


Рисунок 3.2 – Сравнение стоимостей плана до и после оптимизации

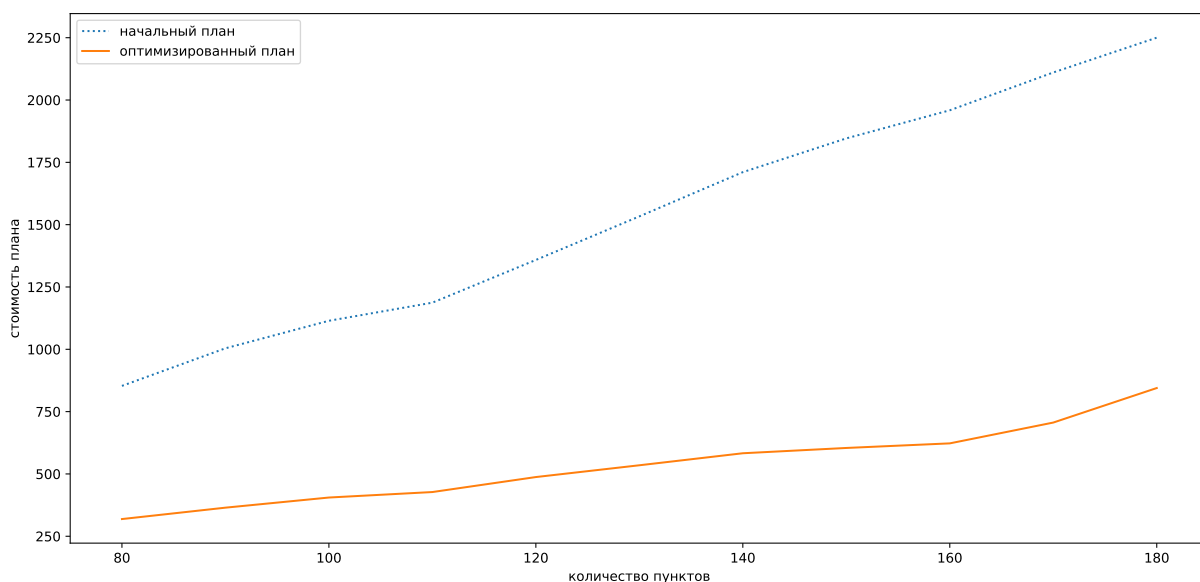


Рисунок 3.3 – Сравнение стоимостей плана до и после оптимизации (продолжение)

Результаты эксперимента показывают, что при любом рассмотренном размере оптимизация опорного плана сокращает стоимость грузоперевозок как минимум вдвое, из чего можно заключить, что метод работоспособен.

### 3.3.4 Определение ограничений программы

Основным ограничением в работе программы может являться рост времени её исполнения при увеличении размерности системы. Для определения существенности данного ограничения следует провести эксперимент по установлению зависимости времени обработки входных данных от их размера.

Для этого будут использованы случайно сгенерированные транспортные системы размером от 20 до 250 узлов. Результат проведённого эксперимента сведён в график, изображённый на рисунке 3.4.

Из графика можно сделать вывод о том, что характер роста функции является нелинейным. При размере сети менее 50 пунктов время оптимизации составляет менее одной секунды. Большее количество узлов приводит к значительному росту времени обработки. Можно сделать вывод о том, что программа завершает оптимизацию за приемлемое время (не более 100 секунд) в

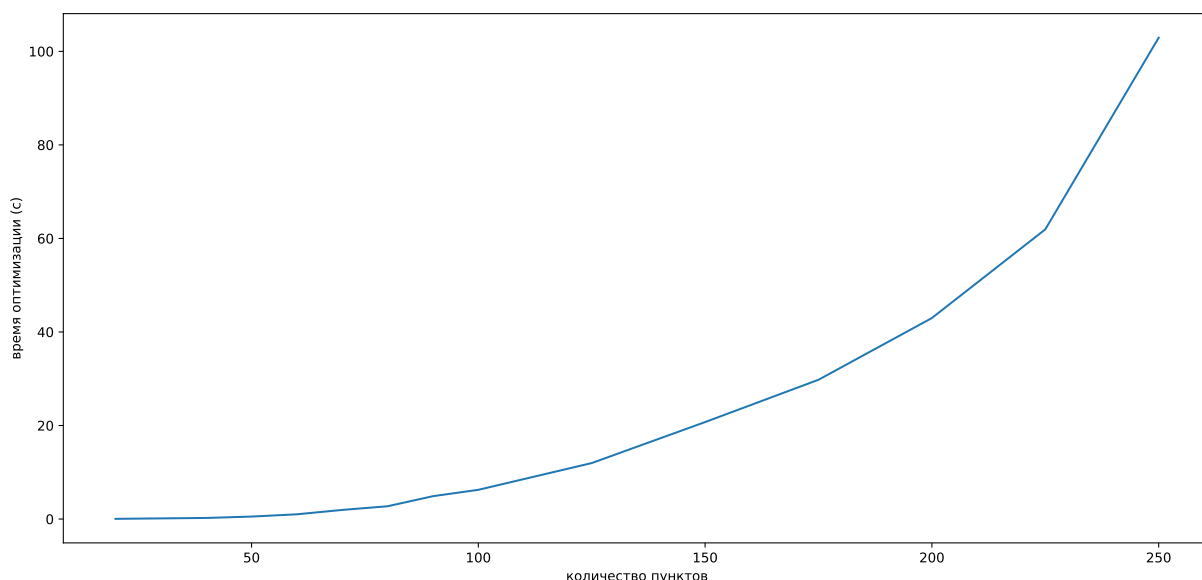


Рисунок 3.4 – Зависимость времени работы от размерности

диапазоне исследованных размерностей.

Ввиду того, что сети подобного масштаба едва ли могут встретиться на практике, а также учитывая допустимость сравнительно длительного времени работы программы, полученные характеристики можно считать приемлемыми для выполнения поставленной задачи.

### 3.3.5 Выявление закономерностей работы системы

Разработанная программа позволяет моделировать поведение транспортной системы при изменении некоторых параметров. Рассмотрим некоторые из них.

В данном эксперименте изучается влияние вместительности грузовика на стоимость грузоперевозок. Размер системы принят равным 50 пунктам. Средний заказ имеет объём в  $1\text{ м}^3$ . Результат эксперимента приведён на рисунке 3.5.

Из данного графика видно, что использование грузовиков меньше объёма одного заказа невыгодно.

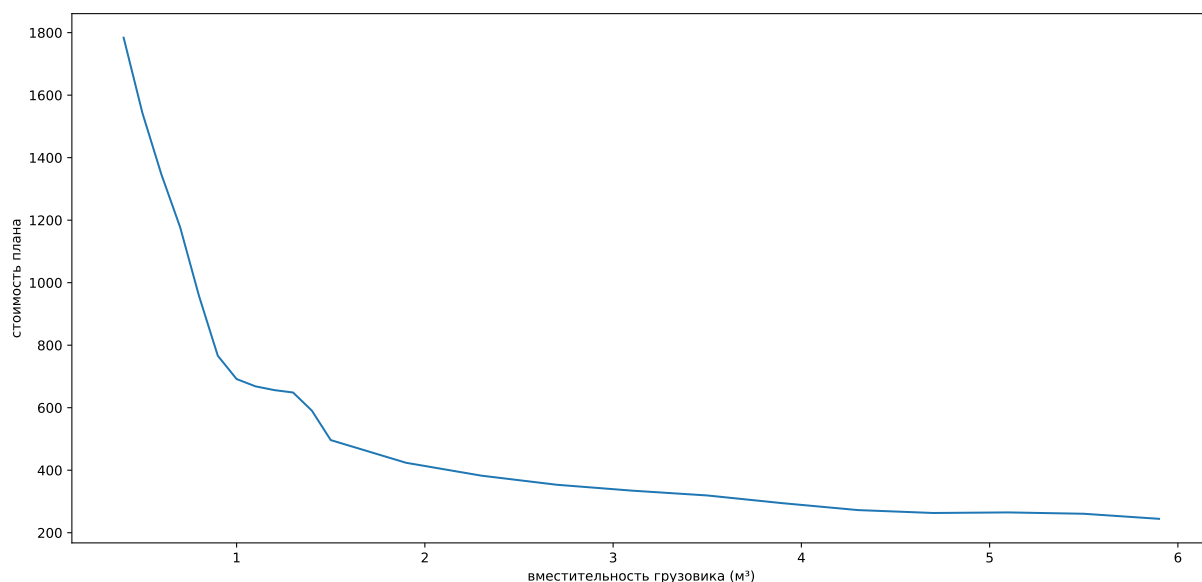


Рисунок 3.5 – Зависимость стоимости маршрутов от вместительности грузовиков

Так как все маршруты начинаются и завершаются на стоянке, её положение относительно других пунктов может существенно влиять на маршруты. На рисунке 3.6 изображен эксперимент по выявлению зависимости стоимости маршрутов от среднего расстояния до стоянки.

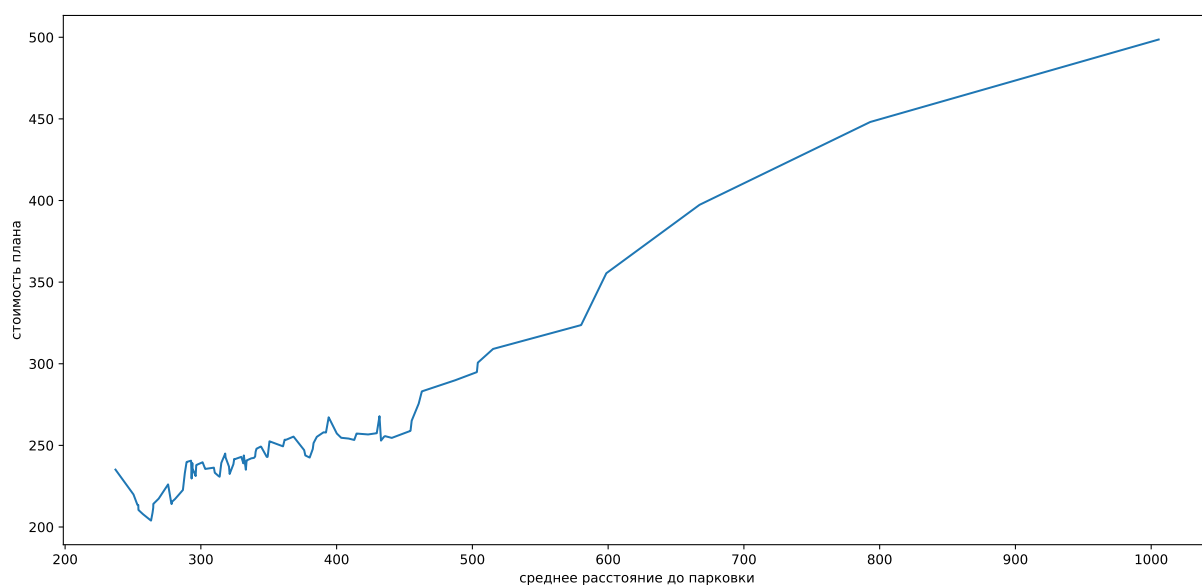


Рисунок 3.6 – Зависимость стоимости маршрутов от удалённости стоянки

Получившийся график не является монотонным, так как средняя удалённость считается по случайно сгенерированной системе, в которой могут иметься иные параметры, влияющие на результат. Однако, с ростом среднего расстояния видная тенденция на рост общей стоимости.

Из этих экспериментов можно сделать вывод о том, что в целях оптимизации издержек, транспортной компании следует минимизировать расстояние от стоянки до потребителей и использовать транспорт, способный вместить в себя по крайней мере несколько целых заказов.

### **Вывод**

В данном разделе были описаны цели и планы проводимых экспериментов. В их рамках были проанализированы изменения различных параметров системы при работе алгоритма оптимизации, экспериментально обоснована работоспособность метода. Были выявлены ограничения для использования разработанной программы, а также некоторые закономерности работы транспортной системы, на основе которых были предложены рекомендации по организации грузоперевозок.

## ЗАКЛЮЧЕНИЕ

В результате прохождения преддипломной практики была достигнута её цель: реализован метод для оптимизации планирования грузоперевозок, основанный на методе потенциалов.

Были выполнены следующие задачи:

- выделен, описан и реализован метод оптимизации грузоперевозок в транспортной системе;
- определён необходимый функционал программного обеспечения;
- описан набор и формат входных и выходных данных;
- разработана структура программного обеспечения, описаны её компоненты, их взаимодействие, а также ключевые алгоритмы и структуры;
- разработан интерфейс, позволяющий задавать конфигурацию транспортной системы и визуализировать результаты работы метода;
- проведено тестирование программы;
- проведена экспериментальная проверка работоспособности программы и ограничений её применения.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Герасименко Е. М. Метод потенциалов для определения заданного потока минимальной стоимости в нечетком динамическом графе // Известия ЮФУ. Технические науки. 2014. №4 (153).
2. Кривопапов В. Ю. РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ С ПРОМЕЖУТОЧНЫМИ ПУНКТАМИ И ОГРАНИЧЕНИЕМ ПО ТРАНЗИТУ //Главный редактор СВ Симак. – С. 28.
3. Цехан О. Б. Моделирование и алгоритмизация одной задачи планирования многопродуктовых перевозок с запрещенным транзитом //Веснік ГрДУ імя Я. Купалы.–Серия. – 2011. – Т. 2. – С. 73-89.
4. Пиневиц Е. В., Ганженко Д.В. МЕТОД ИНТЕРВАЛОВ КАК ИНСТРУМЕНТ ПРИ ОПТИМИЗАЦИИ СОСТАВЛЕНИЯ РАСПИСАНИЯ ГРУЗОПЕРЕВОЗОК // ЗАМЕТКИ УЧЕНОГО - 2021. - С. 317 - 321.
5. Алгоритмы. Построение и анализ : пер. с англ. / Кормен Т., Лейзерсон Ч., Ривест Р. [и др.]. - 3-е изд. - М. : Вильямс, 2018. - 1323 с. : ил.
6. Алгоритмы. Руководство по разработке — 2-е изд.: Пер. с англ. — СПб.: БХВ-Петербург, 2018. — 720 с.: ил.
7. Бесплатные лицензии для обучения программированию [Электронный ресурс]. Режим доступа: <https://www.jetbrains.com/ru-ru/community/education>, свободный (дата обращения 09.04.2022)
8. Сайт PyQt [Электронный ресурс]. Режим доступа: <https://riverbankcomputing.com/software/pyqt/intro>, свободный (дата обращения 09.05.2022)
9. Сайт документации Networkx [Электронный ресурс]. Режим доступа: <https://networkx.org>, свободный (дата обращения 10.05.2022)

10. Сайт графической библиотеки Plotly [Электронный ресурс]. Режим доступа: <https://plotly.com>, свободный (дата обращения 10.05.2022)
11. Сайт документации фреймворка для тестирования unittest [Электронный ресурс]. Режим доступа: <https://docs.python.org/3/library/unittest.html>, свободный (дата обращения 20.04.2022)