

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ПЕТРА ВЕЛИКОГО»
ИНСТИТУТ КОМПЬЮТЕРНЫХ НАУК И ТЕХНОЛОГИЙ
ВЫСШАЯ ШКОЛА ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ
И СУПЕРКОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ

Отчет
о научно-исследовательской работе
на тему: «Применение нейронных сетей для автоматической торговли на бирже»

Кан Максима Евгеньевича, гр. 3530203/80101

Направление подготовки: 02.03.03 Математическое обеспечение и администрирование информационных систем.

Место прохождения практики: СПбПУ, ИКНТ, ВШИСиСТ.

Сроки практики: с 05.06.2021 по 03.07.2021.

Руководитель практики от ФГАОУ ВО «СПбПУ»: Пак Вадим Геннадьевич, доцент ВШИИ, к.ф.-м.н..

Консультант практики от ФГАОУ ВО «СПбПУ»: Резединова Евгения Юрьевна, ст. преп. ВШИСиСТ.

Оценка: _____

Руководитель практики
от ФГАОУ ВО «СПбПУ»

В.Г. Пак

Консультант практики
от ФГАОУ ВО «СПбПУ»

Е.Ю. Резединова

Обучающийся

М.Е. Кан

Дата: 03.07.2021

СОДЕРЖАНИЕ

| | |
|--|----|
| Введение | 3 |
| Глава 1. Модуль для вычисления точности модели | 4 |
| 1.1. Первичная оценка модели | 4 |
| 1.2. Оценка сгенерированных маршрутов..... | 5 |
| Глава 2. Построение модели..... | 6 |
| 2.1. Первое приближение..... | 6 |
| 2.2. Второе приближение | 7 |
| 2.3. Третье приближение | 8 |
| 2.4. Четвертое приближение..... | 8 |
| Глава 3. Генерация маршрутов общественного транспорта | 10 |
| Заключение | 13 |
| Список использованных источников..... | 14 |
| Приложение 1. Модуль для чтения данных измерительных станций..... | 15 |
| Приложение 2. Создание файла с измерениями в формате MATSim | 16 |
| Приложение 3. Использование разных видов транспорта в маршруте на работу | 17 |
| Приложение 4. Сравнение данных в симуляции и в на счетных станциях... | 18 |

ВВЕДЕНИЕ

В предыдущей работе было представлено теоретическое описание и обоснование подхода к моделированию города. Эта работа посвящена практической реализации проекта по моделированию оптимальных маршрутов общественного транспорта.

В первой главе описывается техническая реализация модуля для оценки контрольной модели и сгенерированных маршрутов. Оценка производится после выполнения симуляций, но её определение необходимо для понимания цели и критериев успешности работы.

Для улучшения качества результатов необходимо правильно обрабатывать данные. Во второй главе задокументирован процесс улучшения контрольной модели – от самой простой реализации к более подробным, учитывающим большое количество параметров. Для каждого приближения рассчитана ошибка и обозначены недостатки.

В третьей главе описаны результаты генерации маршрутов с использованием контрольной модели. Рассматривается влияние параметров конфигурации и эффективность выбранного метода генерации.

ГЛАВА 1. МОДУЛЬ ДЛЯ ВЫЧИСЛЕНИЯ ТОЧНОСТИ МОДЕЛИ

Будем использовать тот же способ измерения точности, который предлагали в предыдущей работе. Она состоит из двух частей.

- А. оценка контрольной модели, стремящейся приблизиться к реальному городу;
- В. оценка сгенерированных маршрутов ОТ после замещения ими оригинальных на контрольной модели.

1.1. Первичная оценка модели

К моменту оценки уже должны быть подготовлены и обработаны данные, проведена симуляция. Используемые методы описываются в гл. 2.

Цель симуляции на этом этапе – сгенерировать модель, максимально приближенную к реальности с точки зрения дорожного движения. В идеальном случае все агенты, участвующие в симуляции, должны с точностью повторять все действия своих прототипов из настоящего города.

Для построения маршрутной системы для ОТ достаточно добиться точности только в следующих аспектах модели: пропускная способность дорог; примерный объем автомобильного движения на каждой улице; распределение мест, в которые и из которых может понадобиться проезд для агента. Параметры дорог можно загрузить с сервиса OpenStreetMap, где они описаны для большинства городов, а генерацию маршрутов на основе спроса выполняет модуль Minibus[1][2].

Самые простые способы измерить точность на практике – использовать статистические данные. Правительство может публиковать данные о распределении всех поездок между разными способами передвижения, либо количество пассажиров на линиях общественного транспорта. Подобные данные можно сравнивать с выходными данными симуляции, чтобы замечать значительные расхождения. К примеру, в Плимуте доля поездок на личном автомобиле примерно равна 55[3], из чего можно сделать выводы об общем количестве автомобилей на дорогах.

Более точная оценка должна быть более точечной. В других работах по этой теме (Ziemke и др. в [4], Kickhöfer и др. в [5]) используются результаты измерений движения на дорогах. Будем использовать этот метод.

Для Великобритании доступен набор данных по большому количеству точек замеров по всей стране – как на магистрях и шоссе, так и на локальных дорогах.

Так как данные об измерениях достаточно подробны, будем на каждой дороге отдельно считать количество автомобилей, проезжающих в разных направлениях, а также отдельно для каждого часа с 8 до 19. В пределах Плимута получаем 672 значений.

MATSim поддерживает автоматическое сравнение модели со станциями подсчёта во время симуляции. Будем таким образом проводить промежуточные проверки модели. В файл записывается количество автомобилей, проехавшие каждую точку за час и абсолютное отклонение. Для более гибких вычислений и для визуализации в SimWrapper будем использовать отдельный файл.

Для дополнительного анализа необходимо получать данные из файла `events.xml.gz`, в котором записаны все события, произошедшие во время симуляции. Нас интересуют только события, связанные с автомобилями на дорогах, поэтому выбираем все данные о вхождении автомобилей в поток (`vehicle entered traffic`, `vehicle left traffic`) и переходах автомобилей между узлами (`vehicle entered link`, `vehicle left link`). Преимущество встроенной функции по подсчету в том, что берется среднее число из нескольких итераций. При ручной обработке на сохранение событий всех итераций ушло бы большое количество ресурсов.

1.2. Оценка сгенерированных маршрутов

Маршруты будем оценивать, используя выходные данные симуляции. В графиках указано соотношение видов транспорта, которые используют агенты; долю поездок на сгенерированных маршрутных такси можно напрямую сравнить со статистикой Плимута. Дополнительно будем сравнивать общее количество курсирующих автобусов с их количеством в реальном городе. Таким образом, метрикой будет *количество пассажиров, приходящееся на единицу транспорта*.

Альтернативной оценкой может быть среднее расстояние, которое проезжают пассажиры, или среднее расстояние от остановки до места назначения, или количество вредных выбросов. Результаты симуляции в этих случаях проще всего сравнить с аналогичными результатами в контрольной модели.

ГЛАВА 2. ПОСТРОЕНИЕ МОДЕЛИ

В этой работе выполняется построение модели города Плимут. Его преимущества – довольно большое количество информации в OSM, малый размер и население. К городу примыкают пригороды, но они чётко очерчены, что позволяет выбрать для симуляции только город с небольшим запасом. Крупную агломерацию, например, Лондон, можно симулировать только целиком, что требует гораздо больше ресурсов.

Построение модели состоит из следующих шагов.

- А. создание синтетического населения;
- В. генерация планов жителей в CEMDAP на их основе;
- С. форматирование выходных данных (планы агентов) в формат MATSim;
- Д. импорт дорожной сети с общественным транспортом;
- Е. симуляция в MATSim.

2.1. Первое приближение

В первой версии используется только дорожная сеть города (без пригородов) и примерное распределение мест работы и жилых домов. Эти данные загружаются из OpenStreetMap. Жилые дома – все здания с соответствующими тэгами. Рабочие места, магазины и досуг – *точки*, отмеченные соответствующим ключом. Также накладывается существующая сеть маршрутов автобуса, загруженная из открытого источника.

Для генерации планов агентов в CEMDAP загружается информация о жителях – случайно выбранные зоны работы и проживания, наличие машины, доход.

С такой конфигурацией RMSE (корень среднеквадратичного отклонения) составляет 508,4, MAE (абсолютное отклонение) - 285,6. Таким образом, в среднем ошибка составляет 50%. При визуализации заметно сильное отклонение на шоссе, проходящем через город – очевиден недостаток грузового и транзитного транспорта. Также, судя по часам пик на дорогах между Плимутом и его пригородами, необходимо включить эти пригороды в симуляцию. Значительное отклонение наблюдается и на некоторых точках внутри города.

```

<person id="9540">
  <attributes>
    <attribute name="subpopulation" class="java.lang.String">default</attribute>
  </attributes>
  <plan selected="yes">
    <activity type="home" x="253866.95819370603" y="57231.43504554021" end_time="9:53:00"/>
    <leg mode="car"/>
    <activity type="work" x="251863.09010335957" y="57408.55801556733" end_time="12:30:00"/>
    <leg mode="car"/>
    <activity type="leisure" x="242774.17352491923" y="59596.99180844436" end_time="12:52:00"/>
    <leg mode="car"/>
    <activity type="leisure" x="248299.0643108893" y="60435.76952616133" end_time="13:13:00"/>
    <leg mode="walk"/>
    <activity type="work" x="252007.18950700166" y="57128.120326987926" end_time="14:1:00"/>
    <leg mode="walk"/>
    <activity type="leisure" x="251686.1407134567" y="57130.300553931556" end_time="14:13:00"/>
    <leg mode="car"/>
    <activity type="work" x="252124.0070753956" y="57310.27374751921" end_time="19:53:00"/>
    <leg mode="car"/>
    <activity type="home" x="253866.95819370603" y="57231.43504554021"/>
  </plan>
</person>

```

Рис.2.1. Сгенерированный план одного из агентов

2.2. Второе приближение

Перед этой попыткой проведена более точная предобработка данных. Из OSM загружаются не только отдельные жилые дома, но и области, отмеченные как жилые районы. Это позволяет располагать жильё там, где нет подробной информации о расположении домов. Для определения рабочих мест также используются списки зданий и области (например, промышленная, военная или торговая зона).

Каждое здание или область имеют вес, соответствующий их площади, что позволит точнее распределить жителей. Внутри программы вес используется для расстановки случайных точек внутри области или здания; эти точки в дальнейшем могут использоваться для генерации планов.

Для большей точности будут выделяться не только рабочие и жилые места, но также магазины и места досуга – это позволяет точнее приводить планы к формату MATSim.

Для **симуляции транзитных маршрутов** необходимо связать количество транспорта, въезжающее и выезжающее из города по основным магистралям, с некоторыми статистическими параметрами. Сейчас будем использовать результаты замеров на станциях подсчёта, расположенных на границах города.

Транспорт будет въезжать через одну из магистралей и выезжать через другую. Всего транзитного трафика будет столько же, сколько отмечено счётными станциями, с равномерным распределением по времени.

При таких настройках ошибка уменьшается: RMSE до 426,9, MAE – до 240,8. На границах города входящий поток соответствует измерениям на станциях, но

исходящий сильно отличается. Необходимо выбрать другую схему передвижений для транзитного транспорта. Кроме того, слишком низкий поток между городом и пригородами в часы пик – около 9 и 18 часов.

2.3. Третье приближение

Расширим зону симуляции на пригороды – Салташ, Плимптон, Плимсток. Торпоинт не связан с городом дорогами, поэтому его пропускаем. Так как в этих районах доступно меньше данных, все неразмеченные здания будем считать жилыми.

Для генерации планов в CEMDAP используем дополнительную информацию: крупные коммерческие зоны, количество жилых и рабочих мест, примерный медианный доход в зонах.

Для каждого автомобиля и грузовика, проезжающего город транзитом, будем выбирать некоторую точку в городе (из списка рабочих мест), где он будет останавливаться на какое-то время, после чего будет выезжать из города через ту же магистраль, по которой въехал. Таким образом удастся более реалистично моделировать поведение, например, грузового транспорта, который движется между складами в разных городах.

После выполнения симуляции ошибки составляют: RMSE – 365,9; MAE – 215,4. Здесь так же наблюдаются отклонения, связанные с транзитными потоками. На схеме потоков заметна аномалия: так как весь транзитный транспорт останавливается на одних и тех же точках за пределами города, симулятор не позволяет оставлять автомобили на одном и том же участке дороги (до 2 тыс. автомобилей на участке длиной 20 метров).

Некоторые здания и часть дорог не попали в область симуляции, так как находятся за пределами формальной границы города. Для дальнейших симуляций необходимо вручную выделить область города, включая все значимые места.

2.4. Четвертое приближение

Будем использовать выборку данных внутри полигона, чтобы точнее выбирать область для симуляции. Также будем загружать крупные дороги вокруг города.

На конечных позициях транзитного транспорта за пределами города увеличим размер участков дороги. Эти участки будут выступать в роли парковочных

зон. Поскольку они находятся далеко за пределами симулируемого города, они не должны повлиять на качество симуляции. В соответствии с формулой, приведенной в [6], увеличим на этих участках количество полос до 2 млн. (заведомо больше, чем нужно для стоянки автомобилей).

Аномалии, связанные с построением дорожной сети, возникают в тупиковых путях на границах города (например, односторонняя дорога, ведущая в другой город и из которой нет пути назад в пределах выборки). В Плимуте такая проблема возникает с магистралью в направлении Лондона. Для ее решения создадим искусственный путь, по которому машина может повернуть через разделительную полосу и вернуться в город.

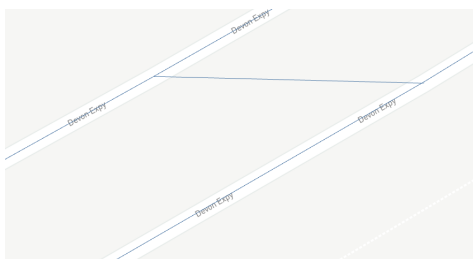


Рис.2.2. Искусственный путь сквозь разделительную полосу

В CEMDAP будем передавать данные с предположением, что у каждого жителя есть автомобиль. Это позволит более реалистично отображать предпочтения жителей, так как они будут выбирать другие виды транспорта не из-за отсутствия выбора, а ради удобства.

Используем откалиброванные параметры для видов транспорта из работы [4].

В этом случае ошибка равна: RMSE – 341,1; MAE – 205,0. Не удалось достичь лучшего качества из-за неточностей в карте. Причинами стали отсутствующая часть восточного района города, неразмеченные дома в пригородах и недостаточно подробная разметка рабочих мест.

Визуализация потоков, сгенерированных на этом этапе, представлена на рис.2.3.

Таблица 2.1

Некоторые параметры файла конфигурации

| Параметр | Значение | Описание |
|-----------------------------|---|---|
| changeMode.modes | car,pt,walk,other | Доступные виды транспорта |
| mobsim | hermes | Быстрый симулятор |
| routingAlgorithmType | SpeedyALT | Быстрый алгоритм маршрутизации |
| strategysettings | SubtourModeChoice TimeAllocationMutator ChangeExpBeta | Модули, регулирующие изменения в планах агентов |
| subtourModeChoice .modes | car,pt,walk,other | Режимы, которые можно использовать в пределах одного маршрута |



Рис.2.3. Визуализация транспортных потоков

ГЛАВА 3. ГЕНЕРАЦИЯ МАРШРУТОВ ОБЩЕСТВЕННОГО ТРАНСПОРТА

Генерация маршрутов выполняется с той же конфигурацией, как симуляция для контрольной модели, но без использования файлов `transitSchedule.xml.gz`, `transitVehicles.xml.gz`. Необходимо выключить параметр `oneThreadPerHandler`, так как в Minibus используется другой модуль для симуляции.

Цель этого этапа в том, чтобы заменить настоящие маршруты на искусственно сгенерированные и сравнить две сети между собой.

При первом тестовом программы создалось 30 маршрутов с общим размером автопарка 650 автобусов, обслуживающих примерно такое же количество пассажиров (18%), как 2000 автобусов в контрольной симуляции. Это показывает, что генерация эффективна для снижения затрат на маршруты при поддержании их качества.

В таблице П4.1 представлены основные параметры Minibus. С их помощью можно регулировать поведение операторов общественного транспорта. Например, при увеличении значения для параметра `earningsPerKilometerAndPassenger` операторы будут стараться работать с длинными маршрутами, при повышении стоимости топлива – с короткими маршрутами в густонаселенных районах, и т.д. Тем не менее, эти параметры стоит воспринимать не буквально, а как гиперпараметры для машинного обучения.

Важно выбрать правильные настройки в соответствии с масштабом системы. В данной работе используется масштаб 0,1, то есть, на одного агента в симуляции приходится 10 человек в реальном городе. В [6] рекомендуется изменять только вместимость транспорта. Таким образом, автобусы в контрольной модели и при работе Minibus вмещают 7 человек, занимая при этом место, эквивалентное 2,8 легковым автомобилям.

Таблица 3.1

Параметры для модуля Minibus

| Параметр | Значение | Описание |
|---|----------|------------------------------------|
| <code>numberOfOperators</code> | 30 | Максимальное количество операторов |
| <code>initialBudget</code> | 2001,0 | Начальный бюджет оператора |
| <code>passPerVehicle</code> | 7 | Количество пассажиров |
| <code>passengerCarEquivalents</code> | 2,8 | Занимаемое место в потоке |
| <code>costPerVehicleAndDay</code> | 100,0 | Стоимость обслуживания автобусов |
| <code>costPerKilometer</code> | 0,30 | Стоимость топлива |
| <code>costPerHour</code> | 10,0 | Расходы за час работы |
| <code>earningsPerBoardingPassenger</code> | 9,0 | Стоимость билета |
| <code>earningsPerKilometerAndPassenger</code> | 1,0 | Доход за километр на пассажира |
| <code>pricePerVehicleBought</code> | 500,0 | Цена автобуса при покупке |
| <code>pricePerVehicleSold</code> | 500,0 | Цена автобуса при продаже |

В обновленной контрольной модели использование автобусов на уровне 12,5%. При симуляции в MATSim такого уровня удалось достичь с использованием

порядка 800 автобусов. Таким образом, достигнуто улучшение эффективности на 60%. Однако превысить долю пассажиров, использующих общественный транспорт, пока не удалось. Возможно, маршрутная сеть достигла такой концентрации, при которой ее дальнейшее увеличение не может окупиться. В этом случае имеет смысл протестировать Minibus с другим набором настроек.

ЗАКЛЮЧЕНИЕ

Цель преддипломной практики достигнута. Разработан набор модулей, позволяющих загружать данные из OpenStreetMap и выполнять предобработку. Реализована программа для оценки результатов контрольной симуляции и генерации маршрутов. Разработаны файлы конфигурации для проводимых симуляций в MATSim.

В ходе выполнения работы обнаружилось некоторые сложности, которые могут помешать при разработке сценариев. Недостаток данных в OpenStreetMap может изменить транспортные потоки. Например, в Плимуте в большей части города не отмечены жилые районы. С другой стороны, большие агломерации требуют большое количество ресурсов, так как могут занимать площадь нескольких крупных городов. В последнем случае необходимо проводить симуляции в масштабах целой страны, как это было сделано для Швейцарии в сценарии, описанном в статье [7].

В случае достаточно отдаленных городов достаточно симулировать движение транзитного и грузового транспорта по основным магистралям, ведущим сквозь город. Поскольку для формирования карты маршрутов необходимо рассчитать только примерные перемещения людей в течение дня, достаточно добиться ошибки в 20% для первичной генерации маршрутов. При необходимости более точной оценки необходимо производить калибровку с CaDyTS, добавлять данные с счетных станций в городе и использовать другие частные методы получения данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. *Neumann A.* A paratransit-inspired evolutionary process for public transit network design. — Berlin: epubli GmbH, 2014. — 268 с.
2. *Neumann A., Röder D., Joubert J. W.* Toward a simulation of minibuses in South Africa // *Journal of Transport and Land Use*. — 2015. — Т. 8, № 1. — С. 137—154.
3. *Plymouth Report: тех. отч.* / Plymouth City Council. — Plymouth, United Kingdom, 2019. — 99 с.
4. *Ziemke D., Kaddoura I., Nagel K.* The MATSim Open Berlin Scenario: A multimodal agent-based transport simulation scenario based on synthetic demand modeling and open data // *Procedia computer science*. — 2019. — Т. 151. — С. 870—877.
5. Creating an open MATSim scenario from open data: The case of Santiago de Chile / *B. Kickhofer [и др.]* // TU Berlin, Transport System Planning and Transport Telematics: Berlin, Germany. — 2016. — № 02.
6. *MATSim User Guide* / *M. Rieser [и др.]*; MATSim Community. — 2014. — 57 с.
7. *Graur D., Bruno R., Bischoff J.* Introducing HERMES [Электронный ресурс]. — 2020. — URL: <https://www.matsim.org/2020/hermes> (дата обращения: 20.05.2022).

Приложение 1

Модуль для чтения данных измерительных станций

```

public static void addToBase(String point_id, String dir,
    int hour, double lat, double lon, int cnt) throws IOException {
    ProjCoordinate station = MatsimGenerator.transformFromWGS84(lat, lon);
    Coord c = new Coord(station.x, station.y);
    double dist = Double.MAX_VALUE;
    Link closest = null;
    List<Link> links = deduceLinkId(point_id, lat, lon);
    for (Link l : links) {
        Coord first = l.getFromNode().getCoord();
        Coord last = l.getToNode().getCoord();
        double dX = last.getX() - first.getX();
        double dY = last.getY() - first.getY();
        if ((dir.equals("N") || dir.equals("n")) && dY > 0
            || (dir.equals("S") || dir.equals("s")) && dY <= 0
            || (dir.equals("E") || dir.equals("e")) && dX > 0
            || (dir.equals("W") || dir.equals("w")) && dX <= 0) {
            double newdist = NetworkUtils.getEuclideanDistance(c,
                NetworkUtils.findNearestPointOnLink(c, l));
            if (newdist < dist) {
                dist = newdist;
                closest = l;
            }
        }
    }
    if (closest == null) {
        System.err.println("Could not find link near " + c.toString());
        return;
    }
    base.merge(closest.getId().toString(), LinkHourlyCount
        .linkInit(point_id, hour, cnt), (v1, v2) -> v1.set(hour, cnt));
}

```

Приложение 2**Создание файла с измерениями в формате MATSim**

```
private static void createMatsimCountsFile(Map<String,
    LinkHourlyCount> cntMap) throws JAXBException {
    Counts<String> counts = new Counts<>();
    counts.setName("GB Traffic Counts");
    counts.setYear(2019);
    counts.setDescription("Traffic counts for Plymouth, UK");

    for (var entry : cntMap.entrySet()) {
        Id<String> id = Id.create(entry.getKey(), String.class);
        counts.createAndAddCount(id, entry.getValue().getStationNo());

        int[] data = entry.getValue().getCounts();
        for (int i = 7; i <= 18; i++) {
            counts.getCount(id).createVolume(i+1, data[i]);
        }
    }

    new CountsWriter(counts).write("scenarios/plymouth/counts.xml");
}
```


Приложение 3

Использование разных видов транспорта в маршруте на работу

```

public static void stats() {
    Config conf = ConfigUtils.createConfig();
    MutableScenario pop = ScenarioUtils.createMutableScenario(conf);
    pop.setPopulation(PopulationUtils.readPopulation(
        "output/plymouth_0/plymouth_0.output_plans.xml.gz"));
    //      Population pop = ScenarioUtils.loadScenario(
        ConfigUtils.loadConfig( "scenarios/plymouth/config.xml"))
        .getPopulation();
    Map<String, Integer> cnt = new HashMap<>();
    for (var plan : pop.getPopulation().getPersons().values().stream()
        .map(Person::getSelectedPlan).collect(Collectors.toList())) {

        ArrayList<PlanElement> pe = (ArrayList<PlanElement>)
            plan.getPlanElements();
        for (int i = 1; i < pe.size(); i+= 2) {
            if (i < (pe.size() - 5)
                && ((Activity) pe.get(i-1)).getType().equals("home")
                && ((Activity) pe.get(i+1)).getType().equals("pt interaction")
                && ((Activity) pe.get(i+3)).getType().equals("pt interaction")
                && ((Activity) pe.get(i+5)).getType().equals("work")) {
                cnt.merge("pt", 1, Integer::sum);}
            else if (((Activity) pe.get(i-1)).getType().equals("home")
                && ((Activity) pe.get(i+1)).getType().equals("work")
                && plan.getPerson().getAttributes()
                    .getAttribute("subpopulation").equals("default")) {
                cnt.merge(((Leg) pe.get(i)).getMode(), 1, Integer::sum);
            }}}
        int overall = cnt.values().stream().reduce(0, Integer::sum);
        for (var e : cnt.entrySet()) {
            System.out.println(e.getKey() + ": " + ((double) e.getValue() / overall))
        }}

```

Приложение 4

Сравнение данных в симуляции и в на счетных станциях

Таблица П4.1

Измерения и ошибка на станции 947308 с 8 до 19 часов

| Станция | Час | Кол-во в симуляции | Кол-во на станции | Ошибка | Нормализованная ошибка |
|---------|-----|--------------------|-------------------|--------|------------------------|
| 947308 | 8 | 360 | 308 | 0.169 | 0.144 |
| 947308 | 9 | 362 | 384 | -0.057 | 0.057 |
| 947308 | 10 | 268 | 213 | 0.258 | 0.205 |
| 947308 | 11 | 200 | 196 | 0.02 | 0.02 |
| 947308 | 12 | 334 | 210 | 0.59 | 0.371 |
| 947308 | 13 | 276 | 201 | 0.373 | 0.272 |
| 947308 | 14 | 370 | 167 | 1.216 | 0.549 |
| 947308 | 15 | 396 | 274 | 0.445 | 0.308 |
| 947308 | 16 | 342 | 264 | 0.295 | 0.228 |
| 947308 | 17 | 446 | 310 | 0.439 | 0.305 |
| 947308 | 18 | 496 | 376 | 0.319 | 0.242 |
| 947308 | 19 | 444 | 197 | 1.254 | 0.556 |
| 947308 | 8 | 254 | 252 | 0.008 | 0.008 |
| 947308 | 9 | 282 | 315 | -0.105 | 0.105 |
| 947308 | 10 | 170 | 207 | -0.179 | 0.179 |
| 947308 | 11 | 174 | 200 | -0.13 | 0.13 |
| 947308 | 12 | 182 | 234 | -0.222 | 0.222 |
| 947308 | 13 | 174 | 253 | -0.312 | 0.312 |
| 947308 | 14 | 222 | 229 | -0.031 | 0.031 |
| 947308 | 15 | 228 | 225 | 0.013 | 0.013 |
| 947308 | 16 | 268 | 306 | -0.124 | 0.124 |
| 947308 | 17 | 238 | 315 | -0.244 | 0.244 |
| 947308 | 18 | 264 | 445 | -0.407 | 0.407 |
| 947308 | 19 | 286 | 223 | 0.283 | 0.22 |