

Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)**

Кафедра автоматизированных систем управления (АСУ)

К ЗАЩИТЕ ДОПУСТИТЬ

Заведующий кафедрой АСУ

д. т. н., профессор

_____ А.М. Кориков

«__» _____ 2018 г.

Разумов Дмитрий Сергеевич

**ПРОГРАММНЫЙ КОМПЛЕКС МОДЕЛИРОВАНИЯ ГОРОДСКОГО
ДОРОЖНОГО ДВИЖЕНИЯ**

Направление магистратуры 09.04.01 – Информатика и вычислительная техника
Магистерская программа – Программное обеспечение вычислительных машин,
систем и компьютерных сетей

Диссертация на соискание степени магистра

Магистрант гр. 436-М1

_____ Д.С. Разумов

«__» _____ 2018 г.

Научный руководитель:

к.т.н., доцент каф. АСУ

_____ А.А. Шелестов

«__» _____ 2018 г.

Томск 2018

Реферат

Магистерская диссертация содержит 101 страницу, 41 рисунок, 4 таблицы, 23 формулы, 19 источников.

МОДЕЛЬ, СТОХАСТИЧЕСКИЙ ПОТОК, ТРАНСПОРТ, МИКРОСКОПИЧЕСКИЙ ПОДХОД, МОДЕЛЬ НАГЕЛЯ-ШРЕКЕНБЕРГА, ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.

Цель работы - проектирование и создание программного комплекса, моделирующего городское дорожное движения, основанного на микроскопической модели Нагеля–Шрекенберга[1].

В результате работы были разработаны алгоритмы движения автотранспортных средств на имитационной модели, проведены исследования автодорожной сети города, а также реализован программный комплекс для моделирования городского дорожного движения г. Томска.

Магистерская диссертация выполнена в текстовом редакторе MicrosoftWord 2010.

Модель – система, воссоздающая поведение некоторого процесса, объекта, устройства, в реальных условиях.

Имитационная модель - логико-математическое определение объекта, для возможного проведение экспериментов на электронно-вычислительной машине (ЭВМ) с целью проектирования, оценки функционирования, анализа поведения объекта в эксперименте.

Поток - совокупность объектов, воспринимаемых как одно целое, существующих в виде процесса измеряемого в абсолютных единицах в некотором временном интервале.

Стохастические поток – поток, задаваемый случайными величинами, принимающий в определённые промежутки времени параметры, которые возвращают определенное значение с некоторой определенной вероятностью.

The abstract

The master's thesis contains 101 pages, 41 figures, 4 tables, 23 formulas, 19 sources.

MODEL, STOCHASTIC FLOW, TRANSPORT, MICROSCOPIC APPROACH, NAGEL-SHREKENBERG MODEL, SOFTWARE.

The aim of the work is the design and creation of a software complex simulating urban traffic, based on the microscopic model of Nagel-Schreckenberg [1].

As a result of the work, algorithms for the movement of vehicles on the simulation model have been developed, research has been carried out on the city's transport network, and a software package for modeling urban traffic in Tomsk has been implemented.

Master's thesis is executed in the text editor MicrosoftWord 2010.

A model is a system that recreates the behavior of a certain process, object, device, in real conditions.

The simulation model is the logical and mathematical definition of an object, for the possible conduct of experiments on an electronic computer (computer) with the purpose of designing, evaluating the functioning, analyzing the behavior of an object in an experiment.

A stream is a collection of objects perceived as a single entity existing as a process measured in absolute units in a certain time interval.

Stochastic flow is a stream given by random variables that takes in certain time intervals parameters that return a certain value with some certain probability.

Содержание

ВВЕДЕНИЕ.....	6
1 ПРОБЛЕМЫ ТРАНСПОРТНОЙ ИНФРАСТРУКТУРЫ РОССИИ	9
2 МОДЕЛИРОВАНИЕ ДОРОЖНОГО ДВИЖЕНИЯ.....	10
1.1 Цели и задачи моделирования дорожного движения.....	11
1.2 Классификация методов исследования дорожного движения.....	12
2.2.1 Математическое моделирование транспортных потоков.....	13
2.2.2 Имитационное моделирование транспортных потоков.....	15
3 МОДЕЛЬ НАГЕЛЯ-ШРЕКЕНБЕРГА	20
4 ПОНЯТИЯ И ВИДЫ ПОТОКОВ.....	22
5 АЛГОРИТМЫ РАЗБИЕНИЯ КАРТЫ НА ЛИНЕЙНЫЕ УЧАСТКИ	25
5.1 Принятые обозначения.....	25
5.2 Точки и операции над точками	25
5.3 Определение центров ячеек полос участков дорог	25
5.4 Расположение автомобилей в произвольный момент времени	30
6 АЛГОРИТМЫ ДВИЖЕНИЯ ТРАНСПОРТА	31
6.1 Алгоритм однополосного движения.....	31
6.2 Алгоритм смены полосы	32
6.3 Алгоритм движения по многополосному участку	32
6.4 Алгоритм задания граничных условий.....	33
6.5 Алгоритм прохождения регулируемого перекрестка.....	33
6.6 Общий алгоритм работы программы.....	35
7 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО КОМПЛЕКСА	37
7.1 Определение требований к программному комплексу	37
7.2 Проектирование интерфейса.....	37
7.3 Выбор технологий	39
7.3.1 Обоснования выбора языка программирования	39
7.3.2 Обоснования выбора среды программирования.....	40
8 РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА.....	42
8.1 Исходные данные к работе.....	42
8.2 Обзор аналогов	44
8.2.1 Mallenom Systems «Дорожный менеджер»	44

8.2.2	ИндорСофт «IndorRoad».....	48
8.2.2	Демонстрационная имитационная модель «Движения автомобилей по Димитровградскому шоссе». Модель Димитровой.....	50
8.3	Формат хранимых данных.....	54
8.4	Программная реализация комплекса	57
8.5	Анализ и сравнение функционала с разработанным продуктом	60
9	ОТЛАДКА И ТЕСТИРОВАНИЕ ПРОГРАММНОГО КОМПЛЕКСА	62
	ЗАКЛЮЧЕНИЕ	65
	СПИСОК ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ	67
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	68
	ПРИЛОЖЕНИЕ А (справочное) Список научных трудов по теме МД	71
	ПРИЛОЖЕНИЕ Б (справочное) Список научных конференций.....	72
	ПРИЛОЖЕНИЕ В (справочное) Алгоритмы работы программы	74
	ПРИЛОЖЕНИЕ Д (справочное)	80

ВВЕДЕНИЕ

В настоящее время, в связи с большой загруженностью городских дорог, проблема пробок является одной из наиболее значимых проблем автомобилистов любого мегаполиса или крупного города. Моделирование стохастических транспортных потоков с возможностью светофорного регулирования не только поможет увеличить пропускную способности перекрестков, но и позволит провести анализ эффективности работы автоматизированных систем управления уличным движением, что может решить некоторые проблемы перепланировки перекрестков, расширение проезжей части улицы. В итоге, это позволит снизить вероятность появления заторов на улицах городов.

Известны два основных направления построения математических моделей движения автотранспорта: макро и микроскопический подходы. При решении данной проблемы была взята за основу микроскопическая модель Нагеля-Шрекенберга. В данной модели поток транспортных средств формируется в виде одномерной решетки. В каждую ячейку решетки может быть помещена частица, которая в данной модели означает машину. Частицы перемещаются из одной ячейки в другую (свободную) в одном направлении [2].

Таким образом, проектирование и создание программного комплекса на основе микроскопической модели Нагеля – Шрекенберга с целью воссоздания модели стохастического потока движения автотранспорта по улицам города, является *актуальной* задачей на сегодняшний день.

Объектом исследованием является поведение автомобилей в условиях городской среды города Томска.

Предметом исследования служат алгоритмы и модели, позволяющие задать и управлять потоком движения, с целью получения необходимых экспериментальных данных.

Цель данной работы заключается в проектирование и реализации программного комплекса, моделирующего городское дорожное движения, основанного на микроскопической модели Нагеля – Шрекенберга.

Для достижения цели были поставлены и решены следующие *задачи*:

- изучение предметной области;
- сбор реальной информации по загруженности перекрестков города, времени и фазам работы светофоров;
- модификация и последующая программная реализация алгоритмов движения автотранспорта по моделируемой проезжей части;
- анализ существующего программного обеспечения;
- выделение процессов и функций программного обеспечения на основе описанной системы;
- разработка программного комплекса моделирования дорожного движения.

Научная *новизна* исследования заключается в разработке алгоритмов движения транспортных средств с использованием модифицированной модели Нагеля-Шрекенберга.

Теоретическая значимость состоит в том, что разработанные алгоритмы представляет собой вклад в научно-методические разработки для работы с автомобильными потоками, способствующие развитию транспортной инфраструктуры города.

Практическая значимость состоит в возможности применения результатов данной работы различными предприятиями города, чья профессиональная деятельность связана непосредственно с автодорожной сетью города. Программный комплекс будет эффективным организациям для проведения экспериментов с целью выявления пропускной способности участков дорог города, исходя из реальных показателей на данный момент времени. Повышения экономической эффективности, достигаемой за счёт

своевременного анализа получения экспериментальных данных, что может повлиять на дальнейший ход реализации проекта.

Основными *защищаемыми положениями* являются:

- 1 алгоритмы движения автотранспорта, используемые при реализации;
- 2 алгоритмы разбиения карты на линейные участки.

1 ПРОБЛЕМЫ ТРАНСПОРТНОЙ ИНФРАСТРУКТУРЫ РОССИИ

Развитие экономической составляющей любого государства, с благосостоянием и жизнью его граждан, напрямую связано с развитием автодорожной сети этого государства. При недолжном развитии инфраструктуры страны, в том числе - дорожной сети, мешает развитию экономики государства и создает опасные ситуации для жизни жителей. Медленное движение между городами, нерационально спроектированные дорожные развязки, специфические настройки работы светофоров – все это сильно тормозит экономику страны. В России от дорожных происшествий ущерб оцениваются в миллиарды рублей.

Согласно докладам и постановлениям Российской Федерации [3] и [4] проблемами автодорожной сети городов являются:

- высокая вероятность появления пробок на улицах городов;
- большая плотность автотранспорта в потоках;
- высокая перегруженность автодорожной сети городов;
- неправильная расстановка парковок автотранспорта;
- низкий показатель средней скорости автотранспорта в потоке;
- высокий уровень транзита по центральным улицам городов;
- низкий уровень дисциплины и лояльности автомобилистов;
- высокая нагрузка на общественный транспорт городов.

Основными причинами проблем автодорожной сети являются:

- неправильное распределение потоков автотранспорта;
- неоптимальная расстановка светофоров на УДС города;
- недостаточный уровень подготовки водителей;
- нерациональное проектирование автотранспортных развязок;
- высокий прирост автомобилистов;
- неоптимальная расстановка парковочных мест.

2 МОДЕЛИРОВАНИЕ ДОРОЖНОГО ДВИЖЕНИЯ

Создание моделей движения транспортных средств на электронно-вычислительных машинах и проведение экспериментов для выявления проблемных участков и выявления оптимальной структуры дорожно-автомобильной сети, является одним из методов оптимизации дорожного движения. Моделирование дорожного движения методом «прямого» моделирования имеет множество специфических проблем. При моделировании движения автотранспорта через регулируемый и нерегулируемый перекресток, необходимо знать и учитывать распределение автотранспорта по направлению движения. Данные по загруженности перекрестков и распределение движения транспорта можно получить лишь методом эксперимента (п.п 7.1).

Внесение изменений в структуру улично-дорожной сети города позволит распределить движение автотранспортных средств по участкам дорог, что откроет автомобилистам новые точки движения по маршруту, возможно, более выгодные для достижения их конечных целей. При полноценном моделировании, также можно включить поведение автомобилистов на участках дороги: перестроении на другую полосу движения и прохождении перекрёстков. При внесении корректировок модели, транспортный поток автоматически подстраивается под заданные изменения и меняет свои параметры движения под новую структуру автотранспортной сети города.

При создании модели движения автотранспорта необходимо учитывать особенности автомобильной сети в целом:

- увеличение пропускной способности участков дорог при развитии автомобильной инфраструктуры города;
- учёт случайных факторов, независимых от автомобилистов (временных, погодных).

2.1 Цели и задачи моделирования дорожного движения

Центром Исследований Автодорожной Инфраструктуры были проведены разработки требований к программным комплексам для моделирования движения автотранспорта [5]. Были определены решаемые задачи с помощью комплексов, такие как:

1. Задачи на уровне города:

- исследование автодорожной сети при введении новых элементов таких как: автомагистралей, метро и т.д.;
- выявление побочных эффектов внедрения автоматизированных систем управления дорожно-автодорожной сетью города;
- анализ влияния экономических санкций на работу автодорожной сети;
- выявление необходимости строительства новых жилых районов города при увеличении или уменьшении дорожно-автодорожной сети;
- анализ перераспределения транспортных потоков, в том числе и пассажирского автотранспорта при перекрытии элемента улично-дорожной сети.

2. Задачи локального порядка:

- эффект достигаемый средствами перепланировки автодорожной сети.

3. Задачи исследования работы автодорожной сети грузового, легкового и пассажирского автотранспорта.

Сроки моделирования транспортных ситуаций может производиться на любой расчётный срок - от задач моделирования текущего дня, до задач моделирования на долгосрочный момент времени. Такие задачи можно разделить на следующие классы:

- краткосрочные задачи моделирования (анализ последствий намечаемых мероприятий ближайших дней, недель, месяцев);

- среднесрочные задачи моделирования (моделирование сроком около пяти лет);
- долгосрочные задачи моделирования (моделирование ситуаций от десяти и более лет);
- оперативные задачи моделирования (в режиме реального времени).

2.2 Классификация методов исследования дорожного движения

В практике проведения исследований в области дорожного движения, как отечественных, так и зарубежных, известно множество способов, начиная от простейших, доступных одному оператору, без должного программного оснащения и заканчивая сложными исследованиями. В связи с тем, что данные исследования требуют множества расчёта показателей, то использование компьютерной техники, при этом методе исследований – необходимо. Поэтому, сложные исследования требуют некоторых навыков от пользователя, такие как знание технической документации, навыки использования электронно-вычислительной техники (ЭВМ).

Для решения задач, связанных с исследованием дорожного движения необходимы определенные методы. Классификация наиболее распространенных методов исследования характеристик и условий транспортного движения для получения экспериментальной информации для дальнейшего исследования дорожной сети. Разделение основных методов исследования дорожного движения можно разделить на три основные группы, такие как [6]:

1. документальное изучение;
2. натурные исследования;
3. моделирование.

Более детальное представление основных методов исследования дорожного движения приведены на рисунке 2.1.

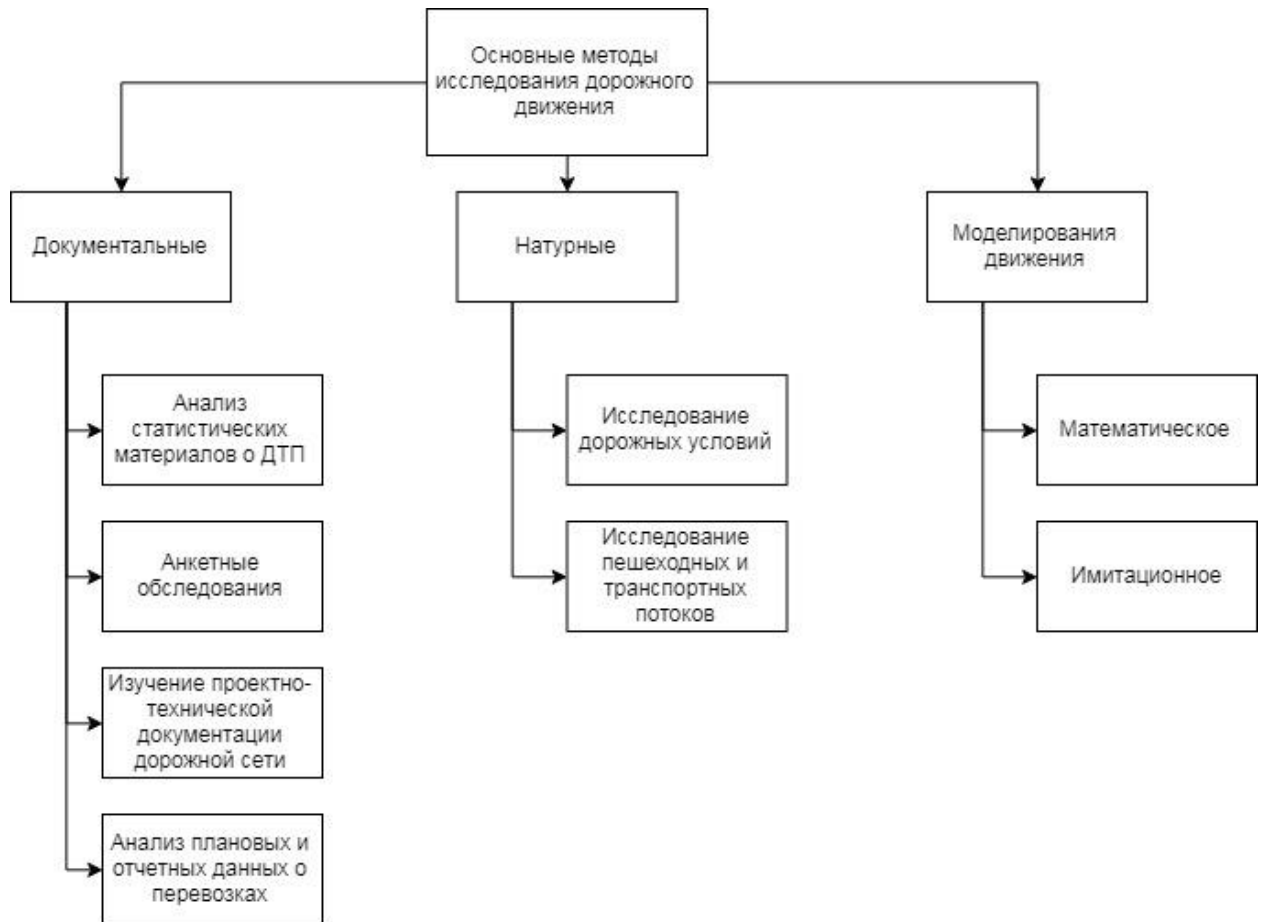


Рисунок 2.1 - Структурная схема классификации основных методов исследования дорожного движения.

2.2.1 Математическое моделирование транспортных потоков

В 1912 году, российский ученый, специалист по городскому планированию Григорий Дмитриевич Дубелир заложил основы математического моделирования закономерностей движения автотранспорта в условиях дорожной сети [8].

Английским ученым Джеймсом Лайтхиллом (Sir Michael James Lighthill) и американским ученым Джеральдом Уиземом была введена первая макроскопическая модель движения автотранспортного потока с позиций механики сплошной среды [9, 10].

Только в 1955 году, американским ученым Ф. Хейтом было проведено математическое исследование транспортных потоков. Далее, благодаря исследованию ученого, исследование транспортных потоков было включено в как самостоятельный раздел в области прикладной математики [11].

Известны два различных подхода моделирования дорожного движения, которые часто комбинируются друг с другом – детерминистический и стохастический подходы.

Моделирование дорожного движения с использованием детерминистического подхода описывает функциональную зависимость между различными факторами, как пример – функциональная зависимость между дистанцией между автомобилями и ускорением каждого из них. При таком моделировании, автотранспортный поток представляется в виде вероятностного процесса.

Классификацию моделей транспортных потоков производится по следующим критериям [7]:

- вероятностные модели;
- модели-аналоги;
- модели следования за лидером.

Моделирование дорожного движения с использованием стохастического подхода представляется как результат взаимодействия участников дорожного движения на автодорожной сети. Так как автомобили в стохастическом потоке подчиняются жестким ограничениям со стороны структурированной автодорожной сети и правил дорожного движения, то, можно: отследить статистические закономерности формирования загруженности по полосам движения, очереди автомобилей, интервалы движения транспорта.

При рассмотрении второго класса, а именно – моделирование на основе моделей-аналогов, движение автотранспортного потока с большим количеством участников движения рассматривается в виде физического потока. Пример такого моделирования можно увидеть при изучении жидкостей и газов, где поток жидкости (в нашем случае – поток автотранспортных средств) – не что иное, как результат движения большого количества молекул. Такой подход к моделированию называется – макроскопическим.

В третьем классе, поведение отдельного автотранспорта моделируется в зависимости от окружающего их состояния на автодорожной сети. Под

окружением понимается связь текущего автомобиля, впередиидущего, и автомобиля, движущегося за лидером. При данном моделировании учитывается поведение участников движения, устойчивость движения, время реакции автомобилистов и исследовался принцип движения на многополосных участках дороги. Такой подход к моделированию называется – микроскопическими.

2.2.2 Имитационное моделирование транспортных потоков

Движение автотранспорта можно рассматривать как сложную систему массового обслуживания, в которой транспорт, поступающий на дорогу, рассматривается как поток входящих заявок, движение автотранспорта от точки начала маршрута до точки назначения – как процесс обслуживания входящей заявки, транспортную сеть – как очереди и каналы обслуживания входящих заявок.

Исходя из вышеописанного, можно выявить основные характеристики данной системы массового обслуживания, такие как:

- средняя длина очереди;
- доля поступивших заявок и заявок на обслуживании;
- среднее время ожидания обслуживания;
- время пребывания заявки в системе т.д.

Но, при таком подходе, классическая система расчёта системы даст грубые и неточные результаты. Так как, основные предположения, лежащие в основе методов расчета классических многоканальных СМО с очередью, могут быть рассчитаны только для отдельно взятого простого элемента их автодорожной сети.

В связи с большой сложностью структуры автодорожной сети, а так же различным поступлением автомобилей в разных участках дороги, невозможно выявить закономерности и прописать формулу, аналогичную формулам Эрланга. Поэтому, наиболее правильным выбором, при решении транспортной задачи, является метод имитационного моделирования процессов дорожного движения (рис 2.2).

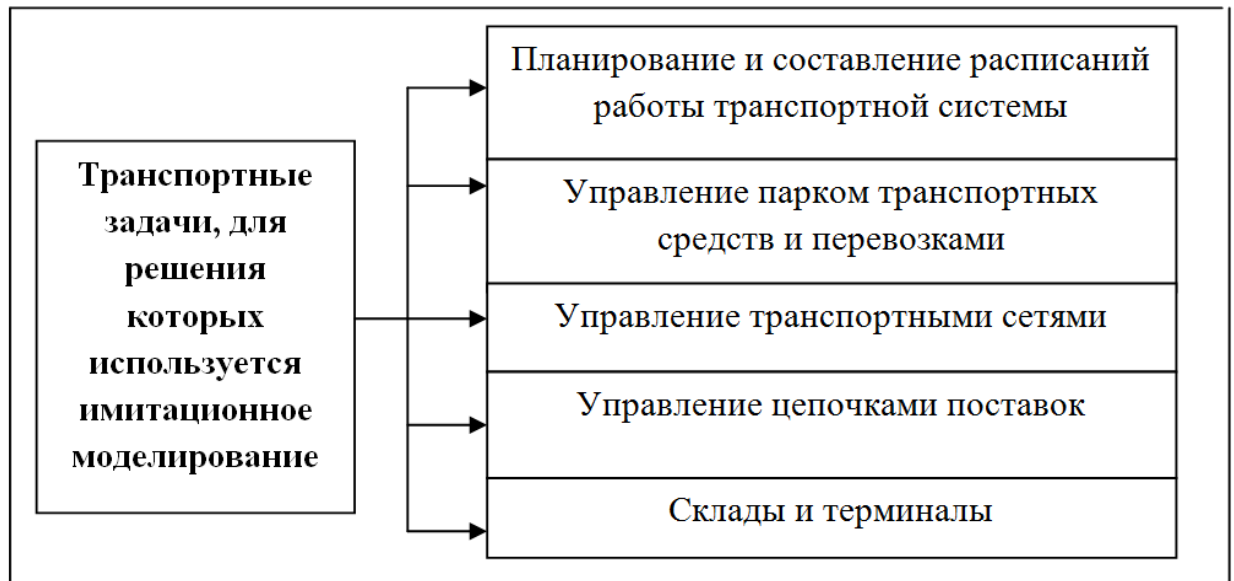


Рисунок 2.2 – Задачи, решаемые имитационным моделированием.

При имитационном моделировании устанавливается строгая структура всей автодорожной сети и правила дорожного движения для каждого из участников. Потоки транспорта стохастически генерируются при помощи генераторов случайных чисел. Транспортное средство при данном подходе является – стохастическим объектом, в роли ПДД выступают алгоритмы изменения состояний системы, карта является детерминированной структурой автодорожной сети. При проведении экспериментов над моделью, можно производить обработку полученных результатов и определить средние показатели автодорожной сети.

Транспортные средства формируют класс объектов, имеющие свои характеристики и параметры такие как:

- тип транспортного средства;
- максимальная скорость движения;
- габариты транспортного средства;
- расстояние тормозного пути;
- стартовая скорость транспортного средства;
- положение на автодорожной сети на текущий момент времени;
- состояние транспортного средства (в движении, остановка);
- начальная и конечная точка маршрута;
- параметры изменения скорости.

Для различного типа транспорта, формируются различные классы. Параметры автомобилей на автодорожной сети задаются по следующему правилу:

1. Строгие параметры:

- тип транспортного средства;
- габариты транспортного средства.

2. Стохастические параметры:

- момент вхождения транспорта в систему;
- начальная скорость движения;
- стартовая точка маршрута транспорта;
- конечная точка маршрута транспорта.

3. Динамические параметры:

- скорость транспортного средства;
- ускорение транспортного средства;
- текущее месторасположение транспортного средства.

Для каждого транспортного средства задаются алгоритмы смены полосы, приоритета движения, возвращение на полосу движения, выбора оптимального маршрута.

С точки зрения математики, транспортная сеть является взвешенным мультиграфом. Каждая вершина графа представляет собой точки сети, такие как:

- начало и конец дороги;
- места стыковки дорог.
- места пересечения дорог;
- места разветвления дорог;

Каждая дуга графа имеет свои координаты и определенными показателями, такие как:

- число полос для каждого направления движения;
- наличие дорожных знаков;
- наличие разметки;
- максимальной скорости движения;
- пешеходных переходов;
- и т.д.

Параметры также задаются и для каждого из перекрестков. Эти параметры задают параметры, к которым обращаются автомобиле при проезде, а именно:

- запрет или разрешение проезда перекрестка с полосы движения, на которой находится транспорт;
- регулируемы или нерегулируемый перекресток
- правила приоритета проезда через перекресток
- правила заезда на перекресток
- и т.д.

При запуске модели, генератор случайных чисел формирует случайные потоки транспорта с заданными параметрами. После чего, производится выполнение алгоритмов. Выполнение алгоритмов влечет за собой изменение системы, другими словами – производится движение автомобилей. Модельное

время может изменяться по двум из основных методов: пособытийно, пошагово.

В первом случае, модельное время изменяется от одного события к другому. Данный метод используется редко для моделей с большим количеством событий.

Во втором случае, модельное время изменяется с заданным шагом. На каждом шаге система эволюционирует, т.е. производится перестроение модели:

- добавление и удаление транспорта с модели;
- смена координат транспорта (движение);
- изменение скорости автомобилей.

После построения и отладки модели проводятся имитационные эксперименты. Результаты экспериментов могут использоваться непосредственной (например, определение среднего времени проезда перекрестка при заданных условиях), или подвергаться дальнейшей статистической обработке для определения степени влияния тех или иных факторов (факторный анализ), связи различных факторов (корреляционный анализ), получения взаимосвязи входных и выходных данных (регрессионный анализ). Для достижения поставленной выше цели этот метод представляется наиболее подходящим.

Удобным математическим аппаратом для реализации таких сложных моделей являются модель Нагеля-Шрекенберга

3 МОДЕЛЬ НАГЕЛЯ-ШРЕКЕНБЕРГА

Модель, разработанная немецкими учёными Каем Нагелем и Майклом Шрекенбергом, представляет собой развитие стохастического одномерного клеточного автомата, предназначенного для моделирования дорожного движения.

Формулировка исходной модели Нагеля-Шрекенберга для однополосного движения заключается в следующем: пусть имеется одномерная сетка, каждая ячейка которой может быть либо свободна, либо занята автомобилем. Размер ячейки принимается равным 7,5 м, что соответствует пространству, занимаемому автомобилем в неподвижном потоке (например, в пробке). Переменные n и V_i - координата и скорость i -го автомобиля соответственно; $g_i = n_{i+1} - l_i - n_i$ - дистанция до лидирующего автомобиля; l - длина i -го автомобиля, в случае данной модели всегда равная единице. Скорость может принимать одно из $(V_{max} + 1)$ допустимых целочисленных значений $V_i = 0, 1, 2, \dots, V_{max}$. На каждом временном шаге t состояние всех автомобилей в системе обновляется в соответствии со следующими правилами (здесь и в дальнейшем для описания правил обновления используется аппарат логики высказываний)[13]:

Ускорение - если $V_i < V_{max}$, то скорость i -го автомобиля увеличивается на единицу; если $V_i = V_{max}$, то скорость не изменяется:

$$v(t) = \min(v(t-1) + 1, V_{max}). \quad (3.1)$$

Торможение - если новая скорость равна или больше расстояния до впереди идущего автомобиля, то значение скорости приравнивается к этому расстоянию:

$$v_i(t) = \min(v(t), g_i(t-1)). \quad (3.2)$$

Случайные возмущения - с заданной вероятностью p водитель уменьшает скорость:

$$\text{if } \xi(t) < p \text{ then } v_t(t) = \max(v(t) - 1, 0). \quad (3.3)$$

Движение - изменение положения автомобиля на сетке автомата в соответствии со вновь вычисленной скоростью:

$$n(t) = n(t-1) + v(t). \quad (3.4)$$

Первое правило (3.1) отражает общее стремление всех водителей ехать с максимальной скоростью, второе (3.2) - гарантирует отсутствие столкновений со впереди идущими автомобилями. Элемент стохастичности, учитывающий случайности в поведении водителей и прочие вероятностные факторы, вносится третьим правилом (3.3). Четвёртое правило (3.4) определяет количество ячеек, на которое переместится автомобиль за одну итерацию.

4 ПОНЯТИЯ И ВИДЫ ПОТОКОВ

Поток — совокупность объектов, воспринимаемых как одно целое, существующих в виде процесса измеряемого в абсолютных единицах в некотором временном интервале [14].

Параметры потока — это параметры, описывающие текущий процесс.

Основные параметры потока:

- начальная точка потока;
- конечная точка потока;
- протяженность потока;
- скорость потока;
- время движения потока от начальной до конечной точек пути;
- интенсивность потока;
- направление движения потока.

Классификацию потоков можно провести по следующим признакам:

1. Со стороны рассматриваемой системы:

- внутренние потоки – потоки, циркулирующие внутри системы;
- внешние потоки – потоки, циркулирующие вне системы.

2. По степени непрерывности:

- непрерывные потоки – потоки, в которых, за момент времени, перемещается определенное количество объектов;

- дискретные потоки – потоки, в которых объекты перемещаются с некоторыми интервалами.

3. По степени регулярности потока:

- детерминированные потоки – потоки с заданными значениями параметров;

- стохастические потоки – потоки, задаваемые случайными величинами, принимающие в определённые промежутки времени параметры, которые возвращают определенное значение с некоторой определенной вероятностью.

4. По степени стабильности потока:

- стабильные потоки – потоки с постоянными значениями параметров в определенный промежуток времени;
- нестабильные потоки – потоки с непостоянными значениями параметров в определенный промежуток времени.

5. По степени изменчивости потоков:

- стационарные потоки – потоки, сформированные во время установившегося процесса. Интенсивность таких процессов постоянная величина;
- нестационарные потоки – потоки, сформированные во время неустановившегося процесса. Интенсивность таких процессов изменяется с течением времени.

6. По характеру перемещения элементов потока:

- равномерные потоки – потоки с постоянной скоростью движения объектов. Объекты в равные промежутки времени, проходят равный путь от начальной до конечной точек маршрута;
- неравномерные потоки – потоки с изменяющейся скоростью движения объектов. Объекты имеют возможность ускориться, остановиться, замедлиться, изменить интервалы отправления и прибытия в конечную точку маршрута.

7. По степени периодичности потоков:

- периодические потоки – потоки с постоянными параметрами или постоянством их изменения через определенный промежуток времени;
- непериодические потоки – потоки с отсутствующей закономерностью изменения параметров.

8. По степени изменения параметров потока заранее заданному режиму:

- ритмичные потоки;
- неритмичные потоки.

9. По степени сложности потоков:

- дифференцированные потоки – однородные потоки;
- интегрированные потоки – разнородные потоки.

10. По степени управляемости потоков:

- управляемые потоки – потоки с возможностью управления со стороны системы;
- неуправляемые потоки – потоки без возможности управления со стороны системы.

11. По характеру образующих объектов выделяются следующие виды потоков:

- материальные;
- денежных средств;
- информационные;
- энергетические;
- транспортные;
- военные и др.

5 АЛГОРИТМЫ РАЗБИЕНИЯ КАРТЫ НА ЛИНЕЙНЫЕ УЧАСТКИ

5.1 Принятые обозначения

Кортежи (упорядоченные наборы фиксированного размера) будем обозначать общепринятым способом. Например, кортеж k из элементов x_1, x_2, x_3 будет записан в виде:

$$k = \langle x_1, x_2, x_3 \rangle. \quad (5.1)$$

При этом для обозначения определённого элемента из этого кортежа будем использовать следующее общепринятое обозначение:

$$x_i = pr_i k \quad \forall i = \overline{1,3}. \quad (5.2)$$

5.2 Точки и операции над точками

Введём множество точек *Points*:

$$Points = \{ \langle x, y \rangle \mid x \in \mathbb{R}, y \in \mathbb{R} \}. \quad (5.3)$$

Для упрощения записи при работе с точками введём операции над элементами множества *Points*:

$$point_1 + point_2 = \langle pr_1 point_1 + pr_1 point_2, pr_2 point_1 + pr_2 point_2 \rangle \quad (5.4)$$

$$point_1 - point_2 = \langle pr_1 point_1 - pr_1 point_2, pr_2 point_1 - pr_2 point_2 \rangle, \quad (5.5)$$

$$point * scalar = \langle pr_1 point * scalar, pr_2 point * scalar \rangle, \quad (5.6)$$

$$point / scalar = \langle pr_1 point / scalar, pr_2 point / scalar \rangle, \quad (5.7)$$

$$veclen(vec) = \langle \sqrt{pr_1 vec * pr_1 vec + pr_2 vec * pr_2 vec} \rangle, \quad (5.8)$$

где $point_1 \in Points$, $point_2 \in Points$, $point \in Points$, $scalar \in \mathbb{R}$, $vec \in Points$.

5.3 Определение центров ячеек полос участков дорог

Далее, все координаты точек подразумеваются в пикселях.

Предположим, что пользователь добавил участок дороги на карту, проведя координатным манипулятором из точки A в точку B .

Прямая AB разбивает плоскость карты на 2 полуплоскости ($planeOnewayR$, $planeOnewayL$). При этом точка C , расположена в

полуплоскости $planeOnewayR$ (эта полуплоскость расположена «справа» относительно вектора AB)(рис. 5.1).

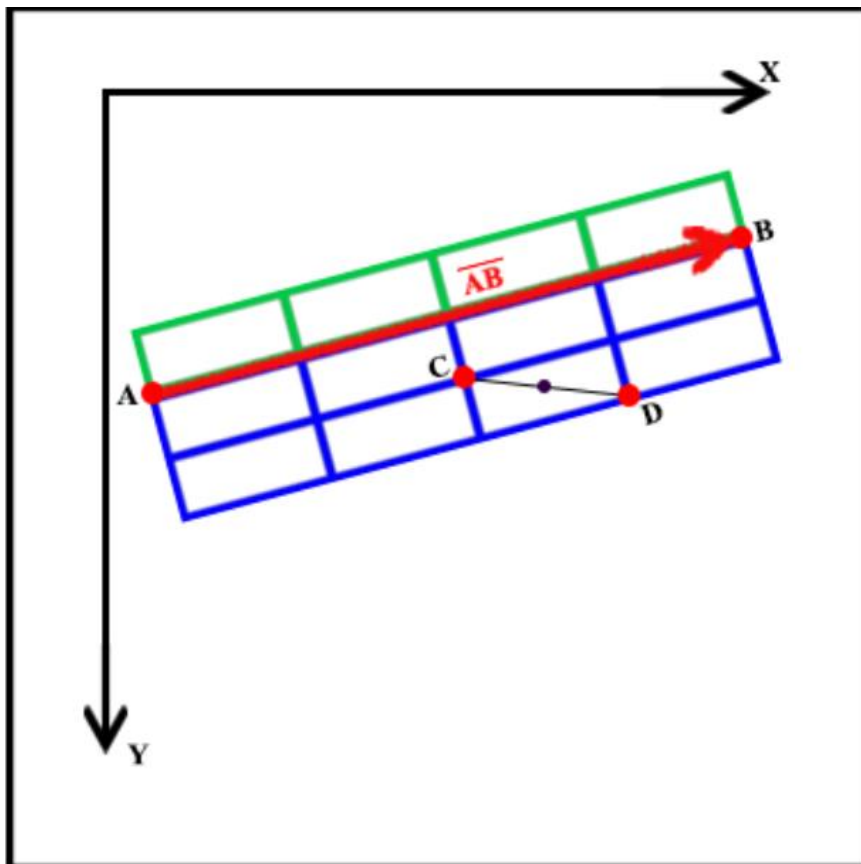


Рисунок 5.1 – Построенный участок моделируемой трассы.

Полосы, расположенные в полуплоскости $planeOnewayR$ группируются в $onewayR \in OnewaysR \subset Oneways$, а полосы, расположенные в полуплоскости $planeOnewayL$ группируются в $onewayL \in OnewaysL \subset Oneways$.

$Oneways$ это множество сущностей, которые заключают в себе полосы участка дороги, предназначенные для движения по ним в одном направлении. Множество $Oneways$ разбивается на 2 подмножества: $OnewaysR$ и $OnewaysL$.

Пользователь далее указывает количество полос в $onewayR$ и количество полос в $onewayL$.

На рисунке X предполагается, что пользователь указал 2 полосы для $onewayR$ и 1 полосу для $onewayL$.

Для каждого из участков $road \in Roads$ дорог после добавления их пользователем существуют следующие отображения (заданные пользователем неявно):

$$roadOnewayR: Roads \rightarrow Oneways, \quad (5.9)$$

$$roadOnewayL: Roads \rightarrow Oneways. \quad (5.10)$$

С помощью этих отображений мы можем получить oneway'и каждой из дорог. Так как пользователь указывает количество полос для каждого из oneway'ев, существуют также следующие отображения:

$$roadwayCnt: Oneways \rightarrow \mathbb{N}. \quad (5.11)$$

Для каждого из oneway'ев имеется 2 точки: точка начала и точка конца. При этом, для *onewayR* точкой начала является точка *A* и точкой конца является точка *B*, а для *onewayL* точкой начала является точка *B* и точкой конца является точка *A*, что математически можно описать следующим образом:

$$roadStart: Roads \rightarrow Points, \quad (5.12)$$

$$roadEnd: Roads \rightarrow Points, \quad (5.13)$$

Для заданных пользователем точек начала и конца участков дорог получить точки начала и конца, соответствующих каждому участку дороги *road oneway'ев*, можно с помощью следующего отображения:

$$onewayStart: Oneways \rightarrow Points, \quad (5.14)$$

$$onewayEnd: Oneways \rightarrow Points. \quad (5.15)$$

При этом:

$$onewayStart(oneway) = roadStart(road) \forall oneway \in OnewaysR,$$

$$onewayEnd(oneway) = roadEnd(road) \forall oneway \in OnewaysR,$$

$$onewayStart(oneway) = roadEnd(road) \forall oneway \in OnewaysL,$$

$$onewayEnd(oneway) = roadStart(road) \forall oneway \in OnewaysL.$$

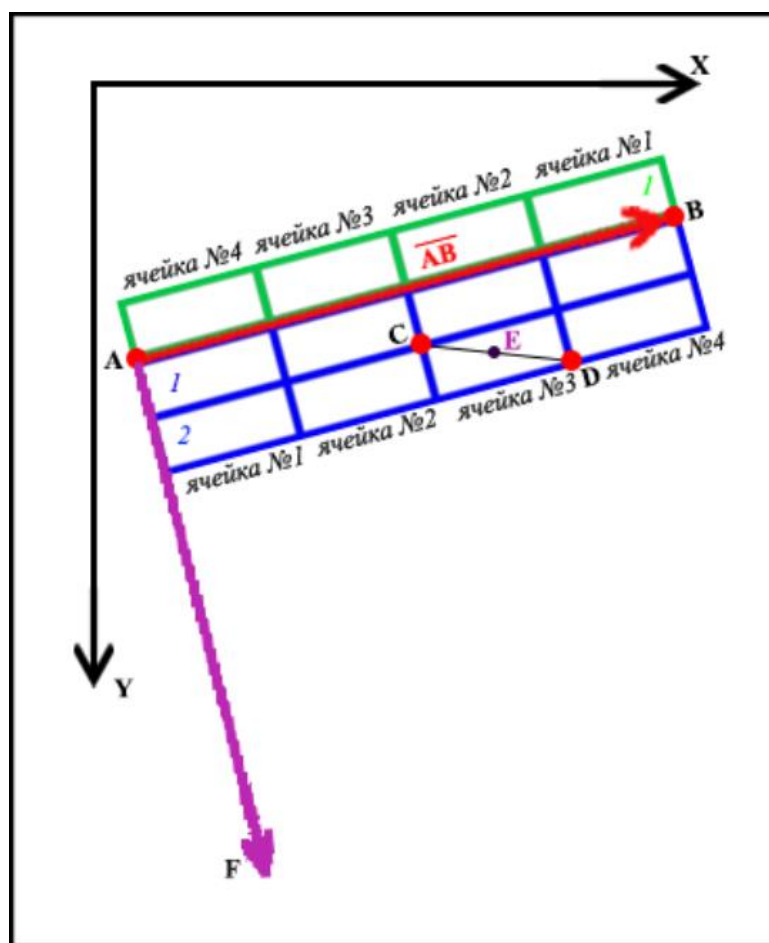


Рисунок 5.2 – Участок моделируемой дороги с обозначениями, после добавления вектора \overline{AF} .

На рисунке 5.2 *onewayR* имеет 2 полосы, а *onewayL* имеет 1 полосу, причём, каждая из полос имеет 4 ячейки, например полоса *onewayL* имеет следующие ячейки: ячейка № 1, ячейка № 2, ячейка № 3, ячейка № 4. *onewayR* имеет следующие полосы: полоса № 1 и полоса № 2. Полоса № 1 расположена ближе к вектору \overline{AB} . Ячейки с меньшими номерами располагаются ближе к точке начала *oneway*'я.

Определить центр E ячейки №3 полосы №2 *oneway*'я *onewayR* можно, если знать координаты точек C и D , а именно $E = (C + D)/2$. Для получения точек C и D можем использовать следующее отображение:

$$\text{cellPointC}(\text{oneway}, \text{roadwayNumber}, \text{cellNumber}) = \frac{\text{onewayStart}(\text{oneway}) + \text{vec}(\text{oneway}) * (\text{cellNumber} - 1)}{\text{cellCnt}(\text{oneway}) + \text{offset}(\text{oneway}) * (\text{roadwayNumber} - 1)}, \quad (5.16)$$

где $\text{vec}(\text{oneway}) = \text{onewayEnd}(\text{oneway}) - \text{onewayStart}(\text{oneway})$.

Ширина каждой полосы принимается фиксированной и равной $roadwayWidthPx = 7,5$ пикселей.

Получить вектор $vecRotA$ можно путём поворота вправо на 90 градусов вектора $vecA$. Этот вектор поможет определить смещение, учитывающее номер полосы. Для поворота вектора можно использовать следующее отображение:

$$vecRot(vec) = \langle -pr2vec, pr1vec \rangle, \quad (5.17)$$

где $vec \in Points$.

Для получения смещение по полосам, необходимо использовать следующее отображение:

$$offset(oneway) = \frac{vecRot(vec(oneway)) * roadwayWidthPx}{veclen(vecRot(vec(oneway)))}. \quad (5.18)$$

Для уточнения количества ячеек в каждой из полос любого $oneway$ 'я необходимо использовать следующее отображение:

$$cellCnt(oneway) = \frac{veclen(vec(oneway)) * scale}{cellLength}, \quad (5.19)$$

где $scale = 0,3304$, $cellLength = 7,5$.

Для данной карты $scale$ — это сколько метров соответствует одному пикселю, а $cellLength$ — это длина одной ячейки в метрах участка дороги, частично соответствует длине автомобиля.

Для получения центров ячеек используем следующее отображение:

$$\begin{aligned} cellCenter(oneway, roadwayNumber, cellNumber) \\ = (cellPointC(oneway, roadwayNumber, cellNumber) + \\ cellPointC(oneway, roadwayNumber + 1, cellNumber + 1))/2. \end{aligned} \quad (5.20)$$

5.4 Расположение автомобилей в произвольный момент времени

Алгоритм каждую секунду времени получает ячейку, в которой должен появиться автомобиль в следующую секунду. Представим, что в момент времени $t_1 = 4$ с. автомобиль располагался в ячейке № 2. До начала следующей секунды, предположим, что алгоритм выдаёт, что в следующую секунду автомобиль будет располагаться в ячейке № 4. То есть известно, что в момент времени $t_2 = 5$ с. автомобиль располагается в ячейке № 4 (рис 5.3).

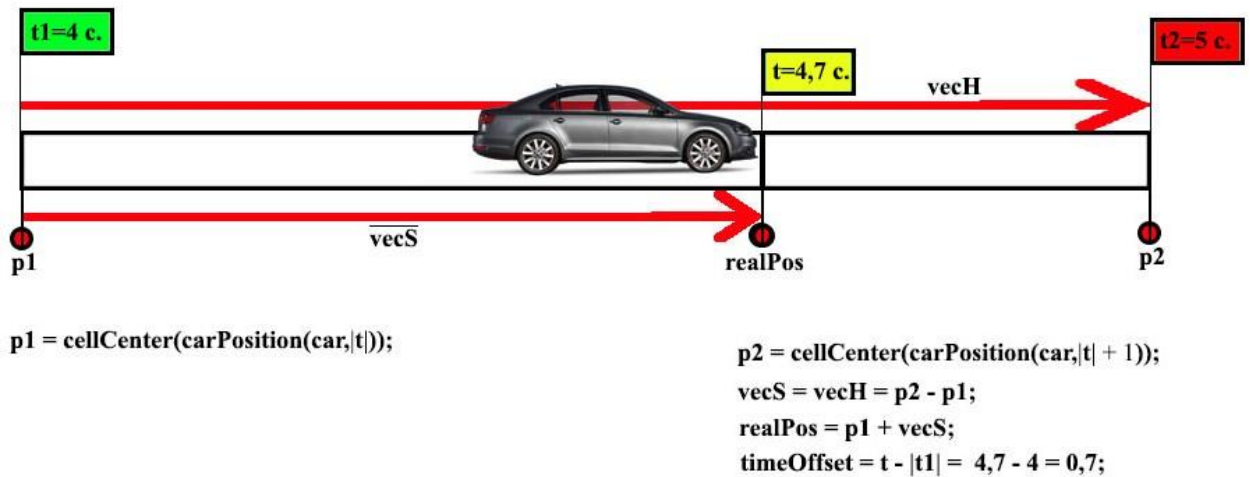


Рисунок 5.3 – Вычисление расположения автомобиля в момент t .

Для того, чтобы в любой произвольный момент времени t (например, 4,7с) отобразить расположение автомобиля на карте, необходимо использовать следующее отображение:

$$\begin{aligned}
 & carRealPosition(car, t) \\
 &= \left(cellCenter(carPosition(car, \lfloor t \rfloor)) \right. \\
 &\quad \left. - cellCenter(carPosition(car, \lfloor t \rfloor + 1)) \right) * (t - \lfloor t \rfloor) + \\
 &\quad cellCenter(carPosition(car, \lfloor t \rfloor)), \tag{5.21}
 \end{aligned}$$

где $\lfloor x \rfloor \forall x \in \mathbb{R}$ — операции получения целой части числа x .

Для получения позиции ($\langle oneway, roadwayNumber, cellNumber \rangle$) автомобиля на секунде $step$ будем использовать следующее отображение:

$$carPosition: Cars \times \mathbb{N}_0 \rightarrow Oneways \times \mathbb{N} \times \mathbb{N}. \tag{5.22}$$

6 АЛГОРИТМЫ ДВИЖЕНИЯ ТРАНСПОРТА

6.1 Алгоритм однополосного движения

Алгоритм построен в соответствии с моделью Нагеля-Шрекенберга. Для каждой полосы (Roadway) в зависимости от загруженности полосы движения, с некоторой вероятностью p формируется очередь автомобилей. Для каждого автомобиля задаются следующие параметры:

- скорость автомобиля - Car.Velocity. Скорость варьируется от 0 до величины Car.MAX_VELOCITY = 4. Единица измерения Car.Velocity, в реальном времени, соответствует 7.5 м/с. При этом, 7,5 метров – длина ячейки;
- расстояние до следующей ячейки впереди идущего автомобиля - Car.Distance. При отсутствии впереди идущего автомобиля, Car.Distance = MAX_DISTANCE.

Правило обновления шага алгоритма Step:

1. **Ускорение текущего автомобиля:** если Car.Velocity < Car.MAX_VELOCITY, скорость текущего автомобиля увеличивается на единицу, в противном случае Car.Velocity = Car.MAX_VELOCITY и ускорение не происходит.
2. **Торможение текущего автомобиля.** Если Car.Distance <= Car.Velocity, то скорость текущего автомобиля Car.Distance - 1.
3. **Стохастическое торможение текущего автомобиля.** Если Car.Velocity > 0, то скорость текущего автомобиля может быть снижена с некоторой вероятностью p .
4. **Движение текущего автомобиля.** Автомобиль совершает движение на количество ячеек сетки, соответствующее своей скорости, после выполнения предыдущих шагов алгоритма.

6.2 Алгоритм смены полосы

Проверка выполнения условий перестроения:

- локальная цель автомобиля Car.LocalTarget является поворотом;
 - имеется соседняя полоса для перестроения;
 - ячейка для перестроения свободна;
 - расстояние впереди идущего автомобиля больше Car.MAX_VELOCITY и позади идущего автомобиля больше или равно Car.Velocity текущего автомобиля.
- При выполнении условий перестроения, текущий автомобиль меняет полосу движения. В противном случае продолжает движение без смены полосы.

6.3 Алгоритм движения по многополосному участку

1. Основой работы данного алгоритма является алгоритм однополосного движения Нагеля–Шрекенберга [2].
2. У каждого автотранспортного средства существует его реальная цель движения Car.LocalTarget. Цель автомобиля обозначается цветом. Синий цвет – движение автомобиля с дальнейшим поворотом направо. Зеленый цвет – прямолинейное движение автомобилем по полосе. Красный цвет – движение автомобиля с дальнейшим поворотом налево.
3. Для каждой машины выясняется необходимость и целесообразность смены его текущей полосы движения, выполняется алгоритм смены полосы.
4. При количестве полос движения Roadway.Count > 2, для разрешения ситуаций в которых автомобили пытаются перестроиться с крайних полос на центральную и попасть в одну ячейку моделированной трасы, шаги перестроения для крайних правых участников движения выполняются в четный шаг алгоритма, для крайних левых – в нечетные.
5. Выполнение подшага выполняется параллельно для всех участников движения.
6. Движение продолжается по алгоритму однополосного движения

автомобиля.

6.4 Алгоритм задания граничных условий

На каждом шаге работы программы выполняется установка граничных условий для спроектированной трассы. Для каждой полосы (Roadway) в зависимости от заданной загруженности полосы движения, с некоторой вероятностью «р», формируется очередь автомобилей, представляющих собой транспортный поток. У каждого транспортного средства существует его реальная цель движения Car.LocalTarget. Цель автомобиля обозначается цветом. Синий цвет – движение автомобиля с дальнейшим поворотом направо. Зеленый цвет – прямолинейное движение автомобилем по полосе. Красный цвет – движение автомобиля с дальнейшим поворотом налево. Основываясь на этих данных, добавляются автомобили с точек заезда на трассу, определенные целью движения для данного участка дороги. В случае, когда трасса перегружена, автомобиль ожидает в очереди до момента освобождения точки въезда на трассу. При выезде автомобиля за границы трассы, автомобиль удаляется из очереди. На каждом шаге работы программы, выполняется проверка на отсутствие машины в точке заезда.

6.5 Алгоритм прохождения регулируемого перекрестка.

1. При приближении к светофору на 200 метров, если автомобиль имеет цель – поворот, автомобиль выполняет перестроение по алгоритму смены полосы.
2. Смена скорости автомобиля на выбранной полосе регулируется согласно алгоритму однополосного движения.
3. Снижение скорости также может потребоваться при следующих условиях:
 - если автомобиль, подъезжая к перекрестку, имеет цель – поворот, то, при приближении к перекрестку за 15 клеток ее скорость Car.Velocity = 2. При повороте на перекрестке Car.Velocity = 0, автомобиль останавливается;

- при условии, что свет светофора красный и скорость машины $Car.Velocity$ больше дистанции до светофора $Distance-1$, то $Car.Velocity = Car.Distance - 1$;

- защита от столкновений, если скорость автомобиля $Car.Velocity$ больше дистанции до ближайшего автомобиля на полосе, то $Car.Distance - 1$, то $Car.Velocity = Car.Distance - 1$.

4. Случайные возмущения.

5. Если автомобиль имеет цель поворота – выполняется поворот через перекресток на другой участок дороги;

6. Новый шаг цикла $Step$.

6.6 Общий алгоритм работы программы

1. Загрузка основного модуля комплекса.
2. Чтение входных данных из файла:
 - a) построение автодорожной сети в виде множества двумерных решеток, которые накладываются на изображение моделируемого участка карты;
 - b) распределение нагруженности участков трассы;
 - c) расстановка светофоров по участкам трассы;
 - d) установка времени и фаз работы светофоров;
 - e) подгрузка модулей добавления и редактирования элементов трассы.
3. Запуск комплекса.
4. Формирование очереди автомобилей с параметрами скорости Car.Velocity, целей автомобилей Car.LocalTarget, запуск таймера отсчета времени Timer() и т.д.
5. Запуск алгоритма формирования граничных условий;
6. Выбор алгоритма автомобиля для прохождения определенного участка трассы:
 - a) при проезде по прямолинейному участку трассы запуск алгоритма однополосного движения;
 - b) при проезде по многополосному участку трассы запуск алгоритма многополосного участка трассы:
 - запуск алгоритма смены полосы;
 - c) при подъезде к светофору, установленному на перекрестке, запускается алгоритм прохождения регулируемого участка трассы:
 - запуск алгоритма смены полосы;
 - d) при подъезде к перекрестку запуск алгоритма прохождения нерегулированного участка трассы:
 - запуск алгоритма смены полосы;

7. Пока не завершена работа программы – обновление состояний ячеек.



Рисунок 6.1 – Алгоритм работы программы.

7 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО КОМПЛЕКСА

7.1 Определение требований к программному комплексу

Программный комплекс необходимо спроектировать на основе следующих требований и принципов:

- интерактивность – это способность информационно-коммуникационной системы, активно и адекватно реагировать на действия пользователя;
- удобство использования – пользователю должно быть удобно работать с системой: удобное размещение элементов интерфейса, уместная подсветка, подсказки, побуждающие посетителя к нужным действиям, будут обеспечивать легкость в анализе полученных данных и комфортную работу с ее источником;
- запуск комплекса в учебных аудиториях на любом персональном компьютере;
- быстрое действие - работа модулей и алгоритмов не должна занимать большой промежуток времени.

7.2 Проектирование интерфейса

Интерфейс должен быть понятен, т.е. пользователю должно быть удобно получать информацию от источника знаний. От размещения элементов интерфейса до подсказок, побуждающих посетителя к нужным действиям, будет зависеть легкость в освоении информации и комфортная работа с ее источником.

К интерфейсу предъявлены следующие требования:

- структура интерфейса должна быть построена в главном окне, без дополнительных окон: включать меню для выбора.
- отображение алгоритмов должно проходить в контексте главного окна;
- отображение информации о соответствующем алгоритме должно сопровождаться графиками, таблицами и/или виджетами с интерактивным взаимодействием;

- информация должна быть доступной, читаемой, адаптированной;
- при большом объеме информации, превышающего размер монитора, необходимо реализовать переход к следующему блоку информации, имеет место быть полоса прокрутки;
- интерфейс должен быть конструктивно информативен, минималистичен, но не перегружен.

При запуске программного комплекса происходит загрузка основных компонентов программного модуля, построение форм, формирование основной страницы интерфейса, которая содержит меню для выбора элементов дорожного движения. Результатом диалога пользователя с программным комплексом при помощи интерфейса, является взаимодействие последнего с ядром программы. Отправляется запрос, после чего, от формы передающей данный запрос, отправляются управляющие параметры в программный код, для запуска процедуры, после чего выполняется вызов процедуры, выполняющей расчёт и визуальное построение данного алгоритма. Визуальное представление работы алгоритма формируется ядром программы при помощи встроенных процедур и отображается в соответствующей области интерфейса.

7.3 Выбор технологий

7.3.1 Обоснования выбора языка программирования

Среда и язык программирования должны максимально точно подходить для реализации требований к программному комплексу.

Призвание языка программирования C# состоит в реализации компонентно–ориентированного подхода к программированию, способствующему наименьшей машинно–архитектурной зависимости полученного кода программы, большей программной гибкости, легкости при переносе, легкости при повторном использовании программ. Так же, C# получил широкое применение в создании программных приложений для установки на компьютер конечного пользователя программной системы. Для простоты диалога программного продукта с пользователем системы используются элементы управления, описанные в библиотеке Windows Forms [15].

Элементы управления — объекты, находящиеся внутри объектов формы. Каждому типу элемента управления соответствует набор некоторых свойств, событий, методов, с определенным назначением. Работа с элементами управления осуществляется при помощи конструктора или при динамическом добавлении во время выполнения программного кода [16].

Можно выделить несколько сильных сторон языка C#:

- типобезопасность кода программы;
- поддержка событийно–ориентированного программирования;
- C# является интегрированным в среду для создания приложения Microsoft .NET.

По вышеописанным причинам, был выбран язык программирования C#. Т.к. основной задачей является обучение конечного пользователя, а именно – студента, необходимо реализовать продукт для более простого усвоения учебного материала. Решение каждого алгоритма должно быть визуализировано, т.е. результаты выполнения работы алгоритмов должны быть

выведены на экран программного продукта. Так же, пользователю программного продукта должен быть удобен и понятен алгоритм работы алгоритма и программы в целом. Ввод пользовательских данных должен соответствовать данным необходимым для корректного выполнения работы алгоритмов.

Для работы с подобным языком программирования необходимо предварительно установить интегрированную среду разработки (IDE), которая включает в себя текстовый редактор, компилятор и/или интерпретатор, средства автоматизации сборки, отладчик. Многие IDE являются платными. Для каждой операционной системы необходимо иметь свою среду разработки. Так же при работе уже с программным комплексом может возникнуть необходимость в установке дополнительных файлов или библиотек на рабочую машину для корректной работы комплекса. При расширении или модификации алгоритмов вновь потребуются IDE, а программный комплекс необходимо будет перекомпилировать и проводить повторную сборку.

7.3.2 Обоснования выбора среды программирования

Microsoft Visual Studio 2015 — представляет собой набор инструментов, направленных на создание программного обеспечения, начиная от планирования пользовательского интерфейса до его разработки, анализа качества программного кода и его производительности, написания программного кода, тестирования, отладки, развертывания в средах конечных пользователей, сбора документации по использованию. С помощью описанных инструментов достигается максимальная эффективность совместного использования в интегрированной среде разработки (IDE) Visual Studio 2015 [17].

Интегрированная среда разработки Visual Studio 2015 была выбрана по нижеописанным причинам:

1. Среда поддерживает программную платформу. NET Framework, позволяющую выполнять, отлаживать, компилировать, перестраивать программы, написанные на объектно–ориентированном языке C#.
2. Удобный интерфейс и поддержка фирмы–производителя Microsoft. Компания Microsoft является крупнейшей компанией, производящей операционные системы семейства Windows.
3. Большой ассортимент статей для обучения и примеров разработки по программному обеспечению.
4. Предоставление бесплатной лицензии для студентов технических вузов в рамках программы «Dreamspark» от компании Microsoft.

8 РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА

8.1 Исходные данные к работе

Исходными данными к работе являются: техническое задание, включающее требования к программному комплексу; выдержки; а так же дополнительная литература, ГОСТы и интернет-ресурсы, приведенные в разделе «Список использованных источников».

На первоначальном этапе была составлена условная схема участка города Томска, с целью формирования граничных условий для дальнейшей реализации программного комплекса. Условные обозначения на схеме отображают перекрестки (П) и участки дорог (УД) города. Данные обозначения были взяты за исходные для дальнейшей программной реализации (рис.8.1).

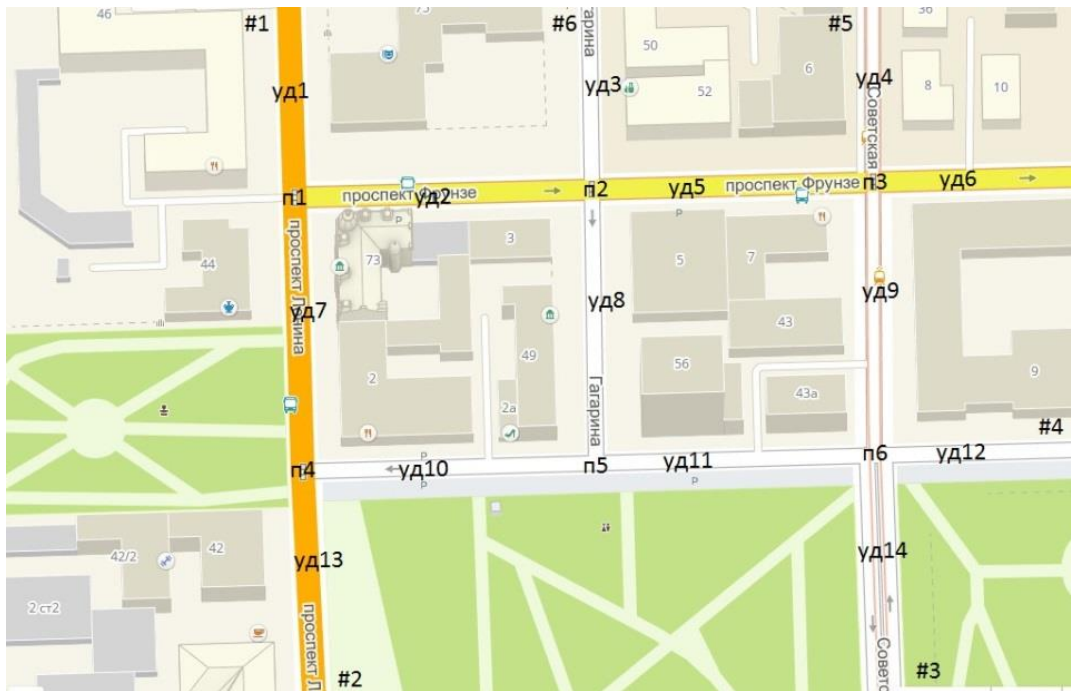


Рисунок 8.1 - Схема граничных условий.

На следующем этапе была осуществлена видеосъемка дорожных потоков города. Видеофиксация производилась на перекрестках с целью отслеживания локальных целей автомобилей, времени и режимов работы светофоров. Съемка проводилась в равные промежутки времени в 10 минут. Места съёмки располагались на перекрестках для того, чтобы рассчитать количество автомобилей, которые проезжают по каждой из полос движения, количество

автомобилей, которые желают повернуть или перестроится в другой ряд движения (рис 8.2).

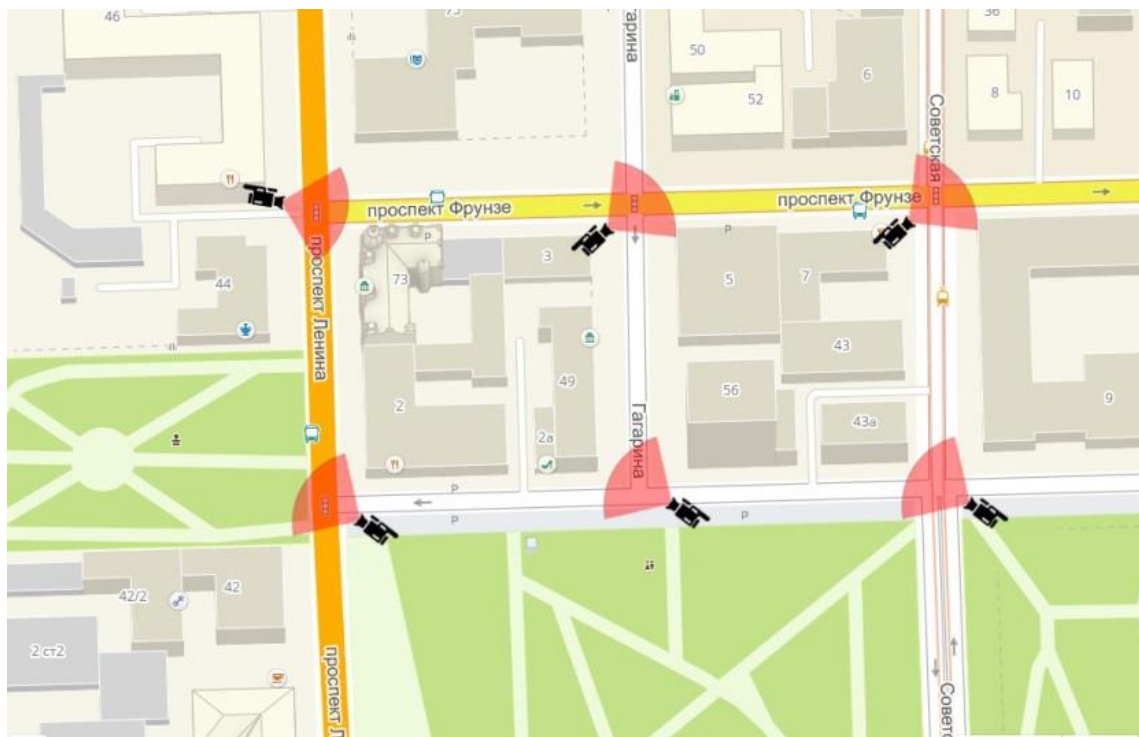


Рисунок 8.2 - Схема расстановки камер.

Результаты съемки были приведены к табличному виду, отражающему распределение автомобилей по локальным целям для каждого из участков дорог, времени работы режимов светофоров. Далее было сформировано схематическое отображение транспортного потока для каждого из перекрестков. Цвет автомобилей (рис. 8.3) определяет намерения дальнейшего движения автомобиля на ближайшем перекрестке. Автомобили синего цвета проезжают перекресток прямо. Зеленые автомобили выполняют перестроение и поворот налево, относительно движения автомобиля. Здесь отображены только светофоры, предназначенные для транспорта. Светофоры для пешеходов в данной модели не учитываются.

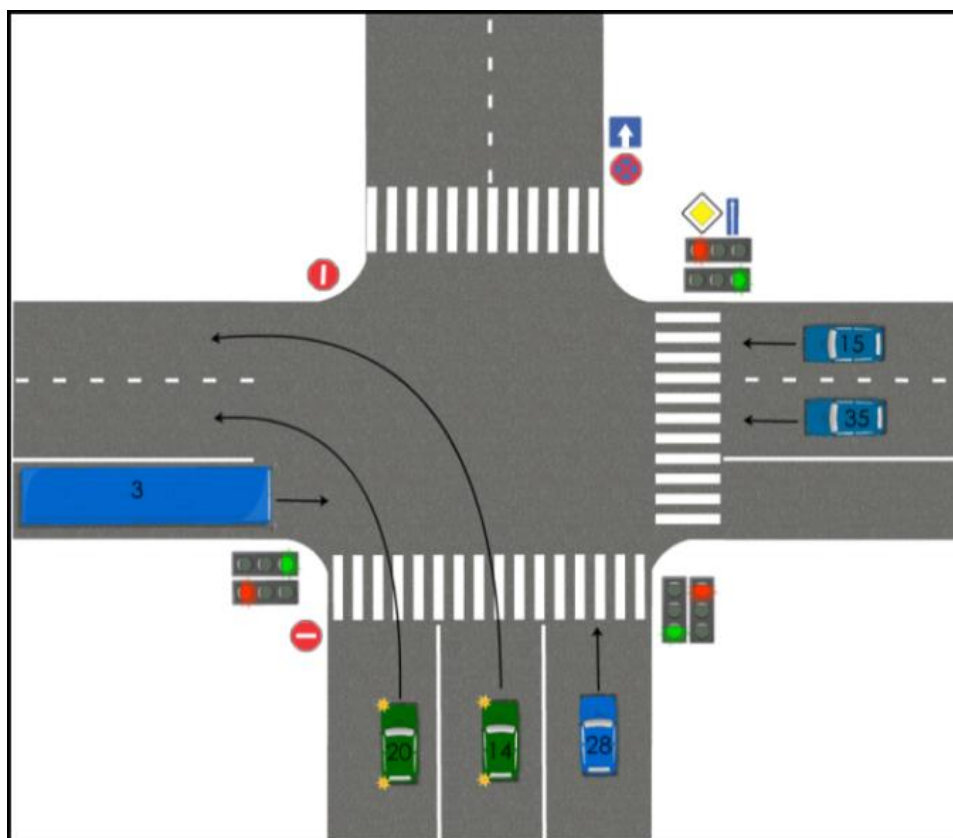


Рисунок 8.3 - Схема потока транспорта для перекрестка П2.

8.2 Обзор аналогов

В настоящее время известно несколько аналогов разрабатываемому комплексу. Для анализа необходимости разработки продукта, который бы мог составить некоторую конкуренцию уже имеющимся аналогам, следует учесть их достоинства и недостатки. При разработке программного комплекса были рассмотрены несколько аналогов, с целью выявления их функционала, такие как:

- mallenom Systems «Дорожный менеджер»;
- индорСофт «IndorRoad»;
- демонстрационная имитационная модель «Движения автомобилей по Димитровградскому шоссе. Город Ульяновск». Модель Димитровой.

8.2.1 Mallenom Systems «Дорожный менеджер»

«Дорожный менеджер» от компании Mallenom Systems, представляет собой ПО для создания моделей автодорожной сети города или региона.

Программный продукт отображает реальную обстановку автодорожной сети в различных дорожных ситуациях на данный момент времени и возможных (благополучных и неблагоприятных) последствий при реконструировании имеющейся сети [18].

Функционал данного продукта весьма обширен, что позволяет пользователям использовать его для следующих целей:

1. Создание и редактирование модели улично-дорожной сети (УДС) города или региона:

- построенная улично-дорожная сеть города в модели полностью отражает реальную улично-дорожную сеть города и включает редактирование при изменении;

- улично-дорожная сеть города в модели содержит в себе основные объекты, такие как:

- остановки маршрутного транспорта;
- светофоры;
- перекрестки;
- участки дорог;
- пешеходные переходы;
- районы;
- начальные и конечные точки маршрута.

- для улично-дорожной сети города устанавливаются: ограничения проезда транспорта по его типу, минимальная и максимальная скорости, приоритет движения, заторы, профили;

- для улично-дорожной сети города устанавливаются светофоры с реальными фазами работы.

2. Калибровка транспортных потоков на модели, базирующаяся на работе автотранспортных детекторов:

- при калибровке модели, учитывается дорожный трафик улично-дорожной сети, на которых размещены транспортные детекторы с заданными

значениями дорожного трафика для каждого из участков. Потоки распределяются так, что разница между эталонными и получаемыми значениями будет минимальна;

- параметры, полученные с реальных автотранспортных детекторов, могут быть приняты за эталонные значения. Так же, эталонные значения могут быть вычислены эмпирическим путем;

- результатом калибровки являются матрицы корреспонденций, составленные на основе реальных потоков улично-дорожной сети города или региона.

3. Моделирование в визуальном режиме наблюдения существующих и прогнозируемых потоков автотранспорта:

- пользователь может наблюдать отображение реального состояние улично-дорожной сети города в разное время;

- работа модели может быть отображена в двух возможных режимах: ускоренного построения и реального времени.

4. Выполнение расчета параметров автотранспортного потока во время работы программы:

- отображение транспортных потоков в виде тепловой карты на отдельных участках улично-дорожной сети в ходе симуляции;

- отображение загруженности отдельных участков улично-дорожной сети по окончании симуляции потока.

5. Выполнение расчета параметров автотранспортного потока в визуальном режиме наблюдения.

Ниже, представлен пример работы программы «Дорожный менеджер» от компании Mallenom Systems с некоторым своим функционалом (рис. 8.4).

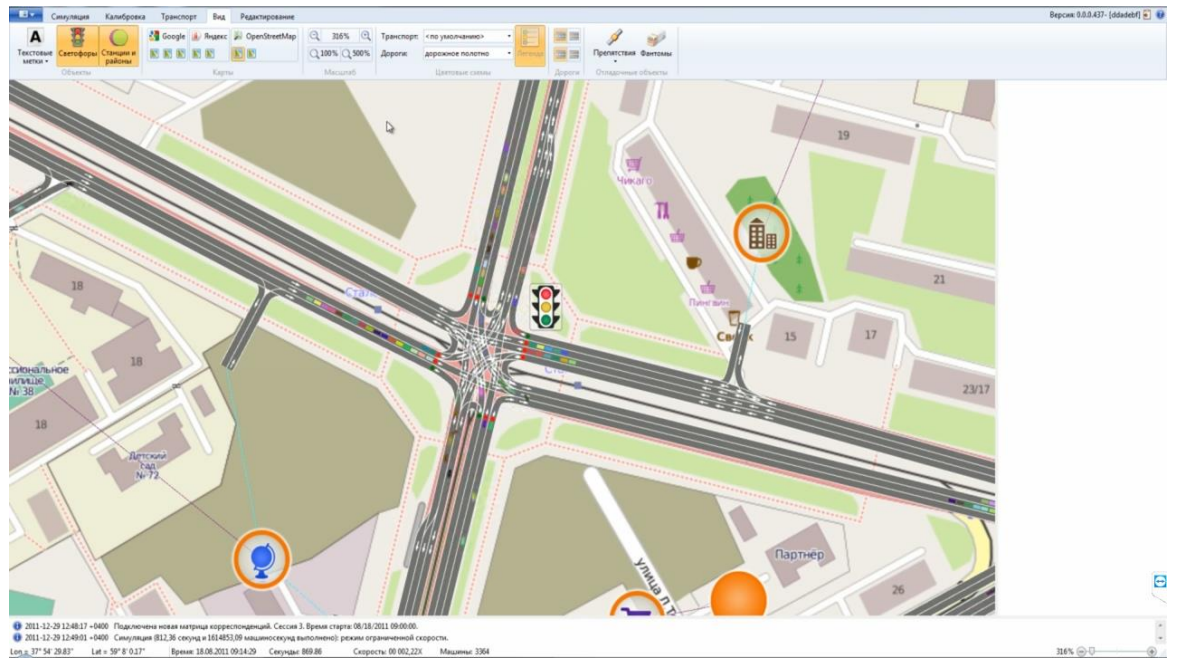


Рисунок 8.4 – Окно программы «Дорожный менеджер».

8.2.2 ИндорСофт «IndorRoad»

Геоинформационная система дорожно-уличной сети города «IndorRoad» от Томской компании ИндорСофт применяется в органах, чья профессиональная деятельность связана с управлением дорожным хозяйством на всех уровнях и организациях-подрядчиков. Данная система может быть использована для управления автотранспортными дорогами вне городской и сельских зонах, так и для отслеживания состояния улично-дорожной сети в городских условиях [19].

Функционал «IndorRoad» весьма обширен, что позволяет пользователям использовать его для следующих целей:

- управления эксплуатаций дорожного полотна;
- паспортизации дорог;
- учёта автомобильных дорог;
- изменения масштабов отображаемой карты: детального и общего плана (обзорной карты). Карты отображают автомобильные дороги в реальных размерах;
- просмотра паспортов и карточек для любого из элемента автомобильной дороги;
- просмотра панорамного видеоряда участков дорожного полотна;
- отображение дорог в матричном виде и в виде отчета диагностики;
- построения плана дороги исходя из пространственных данных, для решения других задач;
- представления информации по автомобильной дороге в виде графика;
- построения высокоточной 3D-модели, которая предоставляет возможность: размеры земельного участка, выявить некоторые дефекты дорожного полотна, уклонов дороги, габариты насыпей;

Ниже, представлен пример работы геоинформационной системы дорожно-уличной сети города «IndorRoad» (рис. 8.5).

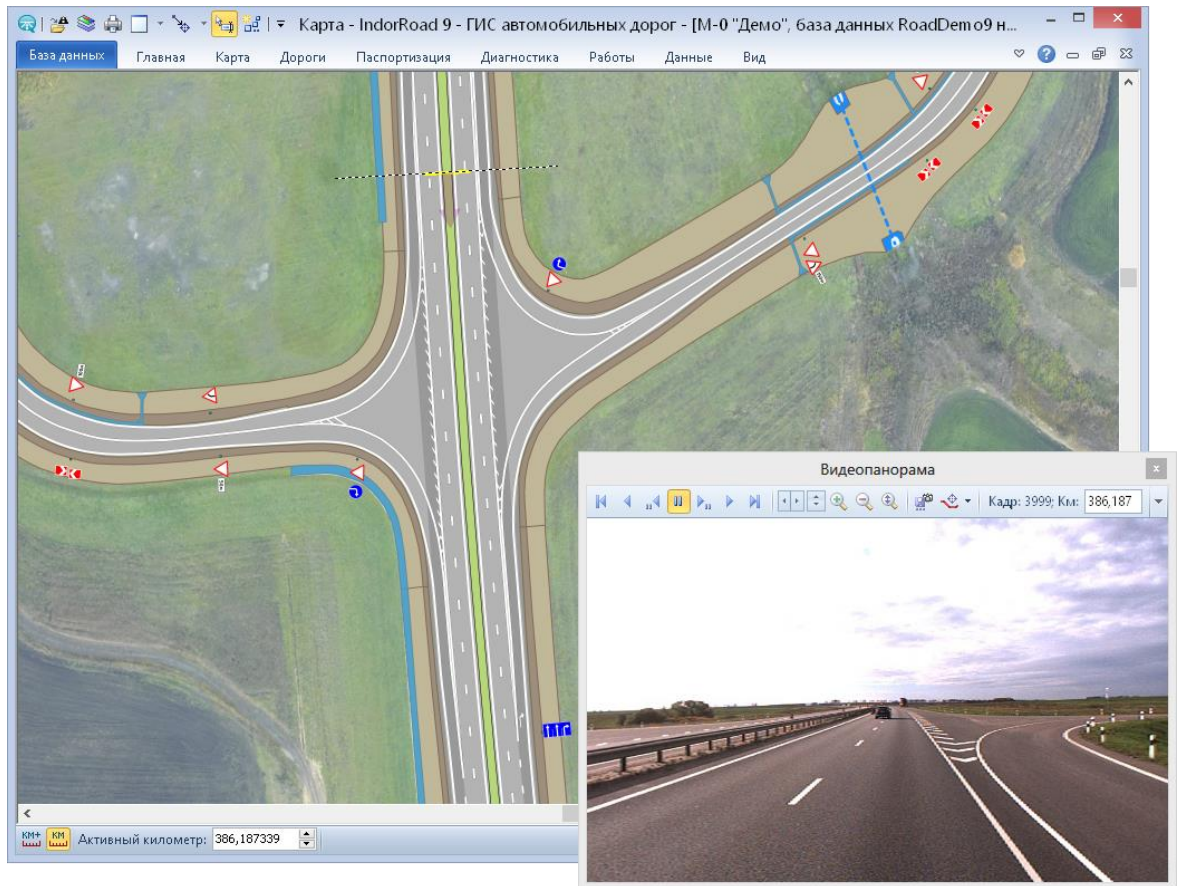


Рисунок 8.5 – Окно программы «IndorRoad».

8.2.2 Демонстрационная имитационная модель «Движения автомобилей по Димитровградскому шоссе». Модель Demitrova.

Демонстрационная имитационная модель «Движения автомобилей по Димитровградскому шоссе» выполнена на базе программного обеспечения AnyLogic. AnyLogic – платформа для создания имитационных моделей любых бизнес-систем. Функционал данного программного продукта включает в себя большой функционал и может использоваться в следующих областях:

- цепи поставок (проектирование, оптимизация политик, оценка риска);
- здравоохранения (проектирование клиники, улучшение процессов, оптимизация политик);
- пассажиропотоки (вместимость зданий, пункты обслуживания, планы эвакуации);
- дорожного движения (планировка дорог, анализ потока транспорта, оптимизация сети);
- железнодорожного движения (проектирование железнодорожной сети, оптимизация графика, испытание политик);
- военно-промышленный комплекс (техническое обслуживание, планирование операций, цепи поставок и логистика).

Моделирование в AnyLogic производится при помощи всех трех современных подходов:

- дискретно-событийного;
- агентного;
- системной динамики.

Данные методы могут быть использованы в любой комбинации на базе одного программного обеспечения, для моделирования разной сложности бизнес-системы. Также, в AnyLogic поддерживает различные виртуальные языки моделирования диаграммы процессов, диаграммы состояния, блок-схемы и диаграммы потоков и накопителей [20].

AnyLogic содержит множества библиотек для большого спектра деятельности:

- библиотека моделирования процессов для типовых производственных и бизнес-процессов.

- библиотека моделирования потоков для процессов транспортировки насыпных и жидких грузов в таких отраслях как горная промышленность или добыча нефти и газа.

- железнодорожная библиотека для железнодорожных перевозок, терминалов и сортировочных станций.

- пешеходная библиотека для пешеходных потоков в аэропортах, торговых центрах, на стадионах или вокзалах.

- библиотека дорожного движения для моделирования машин, грузовиков и автобусов на дорогах, парковках и производственных площадках.

- производственно-складская библиотека для моделирования процессов на заводах и в распределительных центрах.

- библиотека дорожного движения, а также пешеходная и железнодорожная библиотеки позволяют моделировать перемещение и взаимодействие объектов на физическом уровне, что невозможно с другим ПО общего назначения для моделирования.

Демонстрационная имитационная модель «Движения автомобилей по Димитровградскому шоссе» выполнена на базе библиотеки дорожного движения, которая позволяет:

- выполнять проектировку транспортных дорог и автомагистралей с возможностью моделирования производимых временных изменений, перекрытий проезжей части;

- анализировать пропускную способность транспортных сетей, включая статистических данных о пробках и заторах на улицах города;

- производить формирования светофоров на улицах города и оптимизировать фазы работы светофоров;

Модель Димитровой включает весь базовый функционал дорожной библиотеки AnyLogic. При открытии окна программы, пользователю предоставляется возможность просмотреть существующие фазы работы светофоров, установленных на модели. Для более простой работы пользователя, предоставляется возможность выбора уже оптимизированных фаз работы светофоров (рис 8.6).

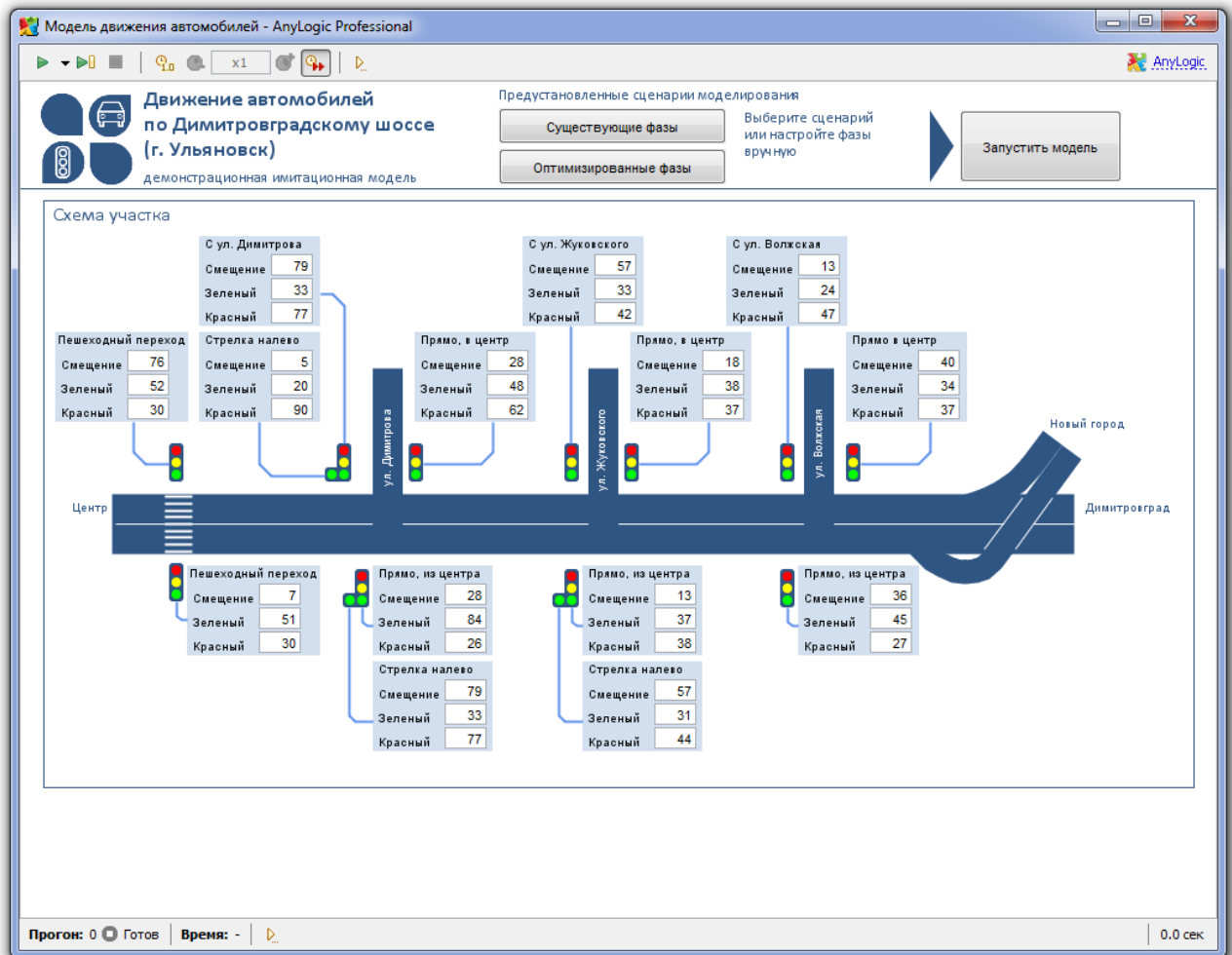


Рисунок 8.6 – Меню редактирования светофорных фаз.

При выборе фаз работы светофоров, программный комплекс готов для построения модели движения.

Модель Димитровой включает в себя следующий функционал, доступный пользователю (рис. 8.7):

- редактировать фазы работы светофоров;
- увеличивать и уменьшать скорость моделирования;
- выбирать необходимый участок трассы;
- увеличивать и уменьшать размер карты;
- отображать время работы светофоров;
- отображать потоки автомобилей;
- выполнять множественные прогоны моделирования;
- выводить на экран статистические данные о потоке;
- информационное окно о продукте.

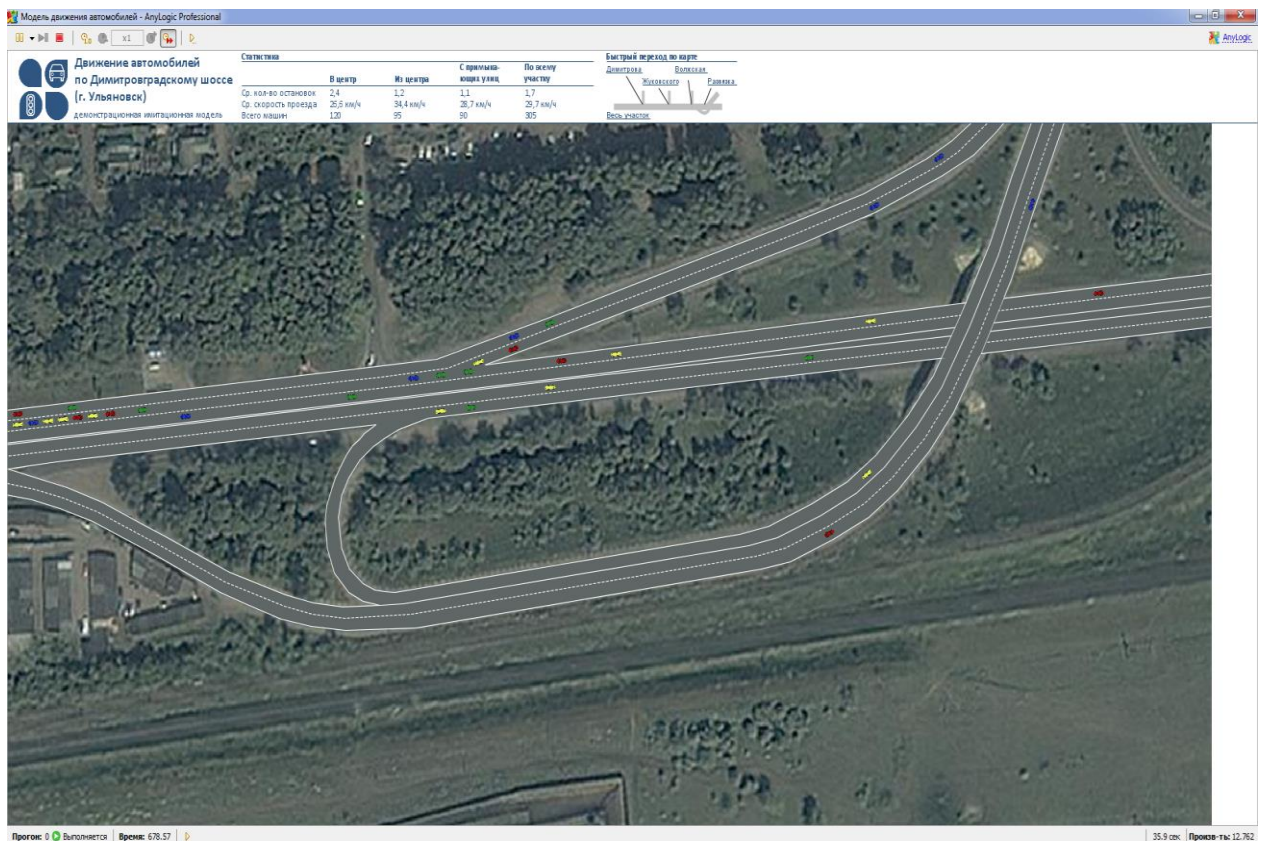


Рисунок 8.7 – Пример работы программы.

8.3 Формат хранимых данных

Реализация данного программного обеспечения предусматривает возможность хранения данных участка карты на долговременном носителе. В нашем случае было принято решение хранить информацию об отредактированном участке карты в файле формата MAP.

Данный формат был разработан исключительно для данного программного продукта.

Описание формата файла MAP:

m n

<Секция светофоров>

<Секция участков дорог>

Где *m* - это число светофоров, *n* - число участков дорог.

Секция светофоров:

<Описание светофора>

<Описание светофора>

...

<Описание светофора>

Секция участков дорог:

<Описание участка дороги>

<Описание участка дороги>

...

<Описание участка дороги>

Описание светофора:

id has_left has_right x y p

<Описание состояния 1>

<Описание состояния 2>

...

<Описание состояния N>

Где *id* - идентификатор светофора; *has_left* - символы "true", если светофор имеет дополнительную секцию слева (иначе - символы "false"); *has_right* - то же для дополнительной секции справа; *x*, *y* - координаты размещения центра жёлтого сигнала светофора относительно левого верхнего угла участка карты; *p* - число состояний для данного светофора.

Описание состояния:

sec left center right

Где *sec* - число секунд, в течение которого светофор находится в данном состоянии; *left* - "true", если дополнительная секция слева имеет разрешающий сигнал, "false", если дополнительная секция слева имеет запрещающий сигнал, "null", если дополнительная секция слева отсутствует; *right* - то же, что и *left*, но для дополнительной секции справа; *center* - "true", если основной сигнал светофора разрешающий, "false" - в противном случае.

Описание участка дороги:

id

x0 y0 x1 y1

<Описание зелёных однонаправленных полос участка дороги>

<Описание синих однонаправленных полос участка дороги>

Где *id* - идентификатор участка дороги; *x0*, *y0* - координаты начала дороги; *x1*, *y1* - координаты конца дороги.

Описание однонаправленных полос участка дороги:

left_cnt straight_cnt right_cnt roadways_cnt

<Описание полосы 1 дороги>

<Описание полосы 2 дороги>

...

Где *left_cnt* - число машин, которые поворачивают налево в течение периода *t*; *straight_cnt* - прямо; *right_cnt* - направо; *roadways_cnt* - число полос в данном направлении.

Описание полосы дороги:

car_cnt period lights_id left_id center_id right_id

Где *car_cnt* - число машин, которые появляются на данной полосе в течение периода *period*; *lights_id* - светофор, на который необходимо ориентироваться водителям автомобилей в конце полосы этого участка дороги; *left_id* - идентификатор однонаправленных полос участка дороги, на которые необходимо переместиться автомобилям, проезжающим перекрёсток в конце полосы налево; *center_id* - прямо; *right_id* - направо.

В случаях, когда автомобили не имеют возможности проехать в одном из этих направлений, вместо идентификатора однонаправленных полос участка дороги указываются символы "null".

Функции преобразования объектов-сущностей программы в текстовое представление соответствующих секций имеют название `serialize()` во всём проекте. Функции, осуществляющие обратные преобразования, названы `parse()`. При этом все функции `serialize()` являются функциями экземпляра классов и возвращают значение типа `string`. Все функции `parse()` - являются статическими принимают 1 параметр типа `string` и возвращают объект соответствующего класса. Сериализация объекта карты иерархически вызывает сериализацию объектов, расположенных на карте. То же касается процесса десериализации.

8.4 Программная реализация комплекса

В результате проведенных исследований и программной реализации алгоритмов движения транспортных средств был создан программный комплекс моделирования дорожного городского движения (рис. 8.6).

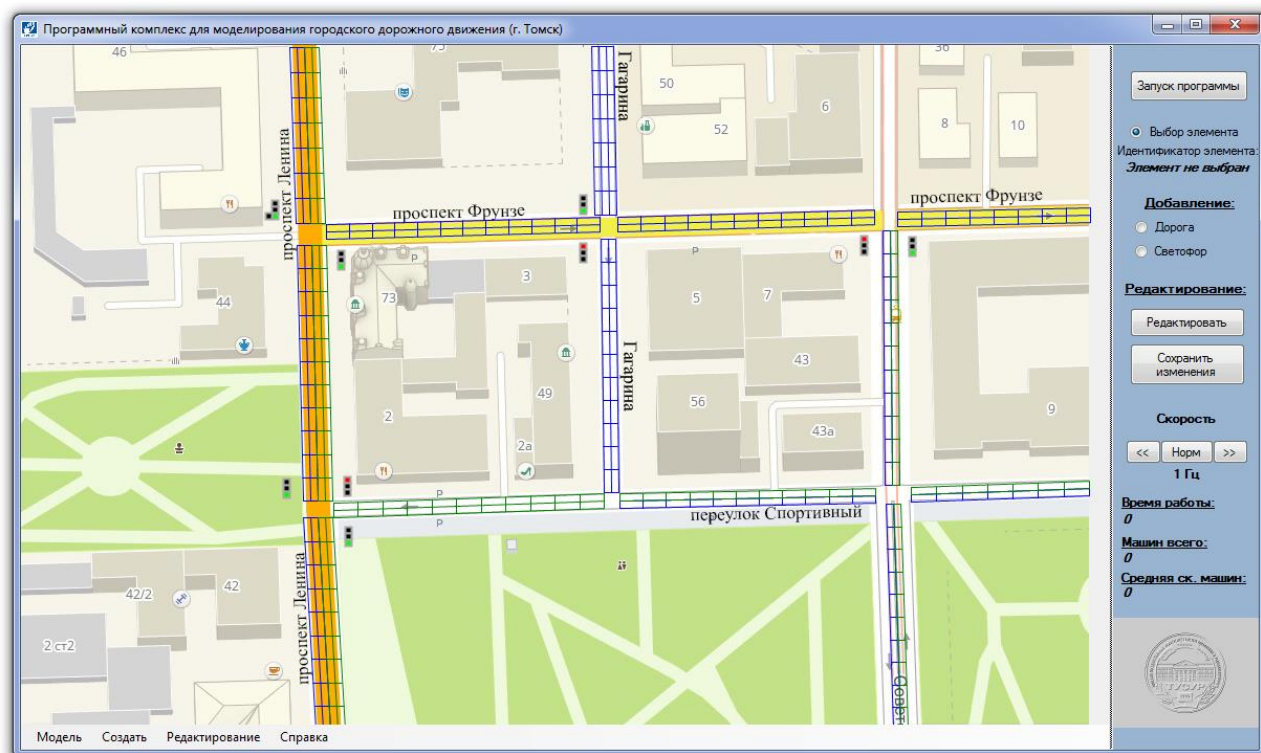


Рисунок 8.6 - Главное меню программного комплекса.

При запуске программного комплекса загружаются все необходимые данные для моделирования движения автомобилей на выбранном участке карты г. Томска. Данные о загруженности перекрестков и целях автомобилей, полученные экспериментальным путем, были занесены и сохранены в программе (рис.8.7).

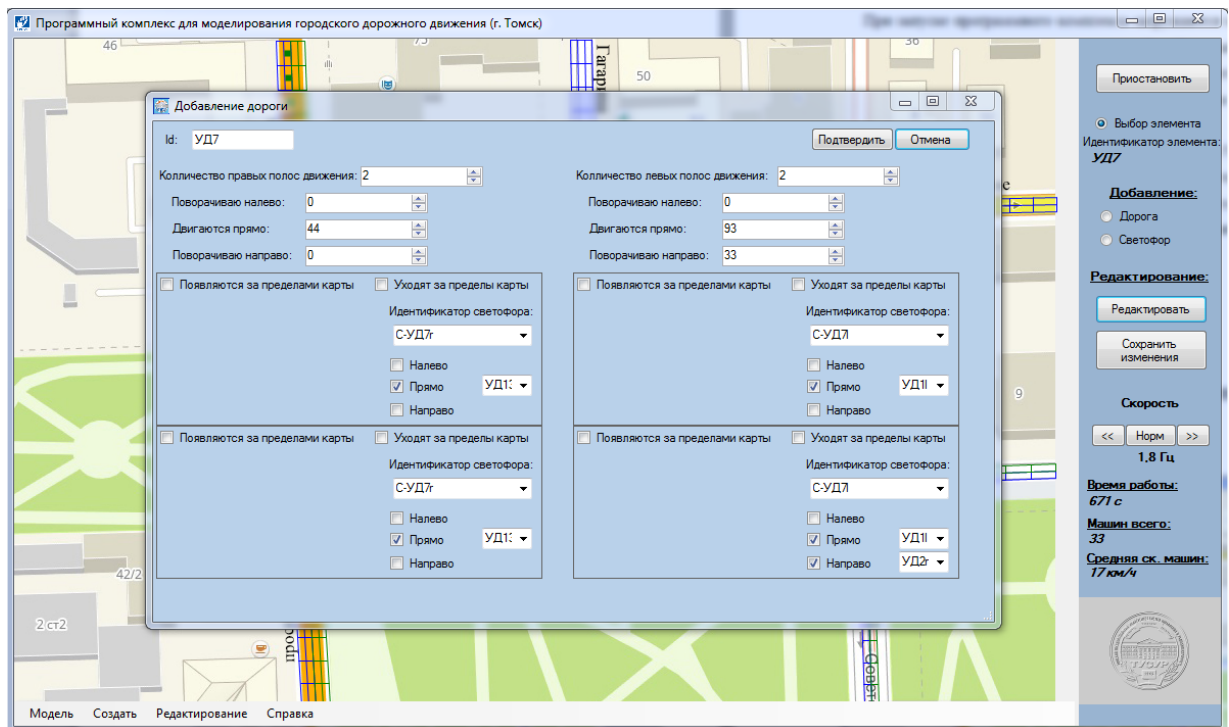


Рисунок 8.7 - Нагруженность участка дороги УД3.

Время и режимы работы светофоров на всех перекрестках города перенесены на моделируемый участок карты. Режимы работы светофоров были синхронизированы между собой, что позволило отобразить их работу на модели, соответствующую реальной схеме движения (рис. 8.8).

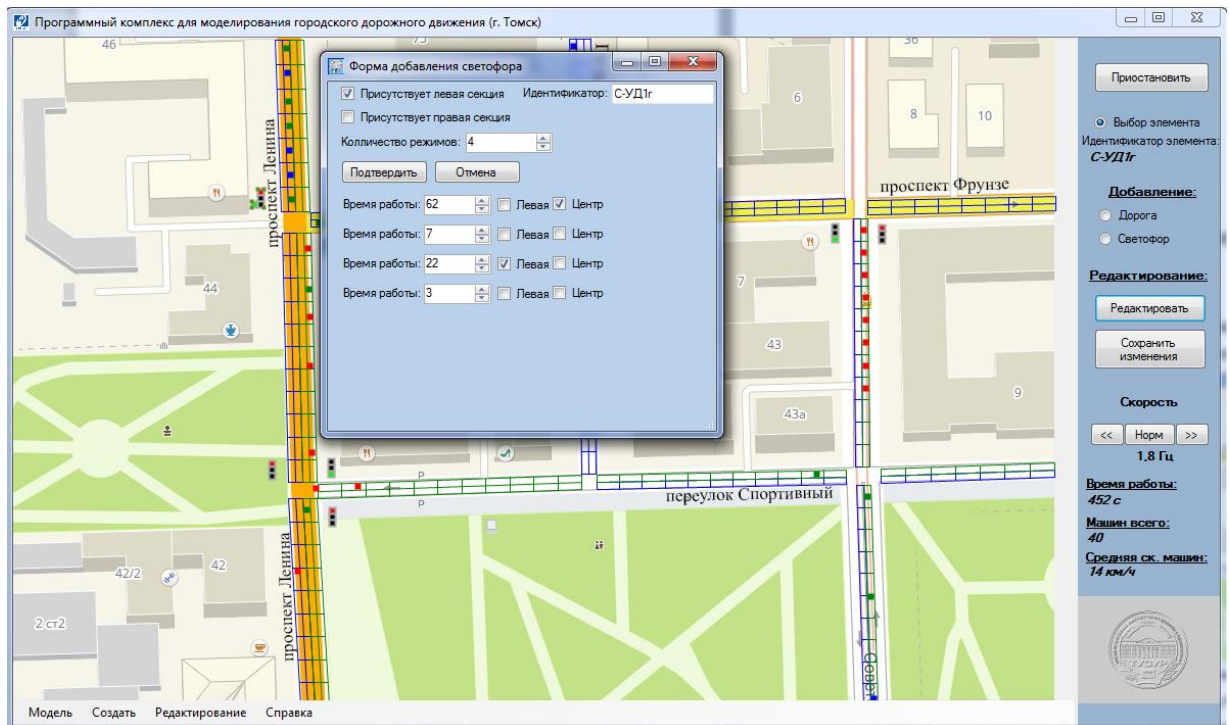


Рисунок 8.8 - Время и режимы работы светофора на перекрестке П1.

Запуск работы модели влечёт за собой запуск таймера, отсчитывающий периоды в 50 м/с. Данный таймер необходим для обновления расположения автомобилей на карте. При этом каждую секунду, выполняется вызов метода Step() для карты, который определяет новую позицию (ячейку) для каждой из машин, расположенных на карте по ранее приведённым алгоритмам (рис. 8.9).

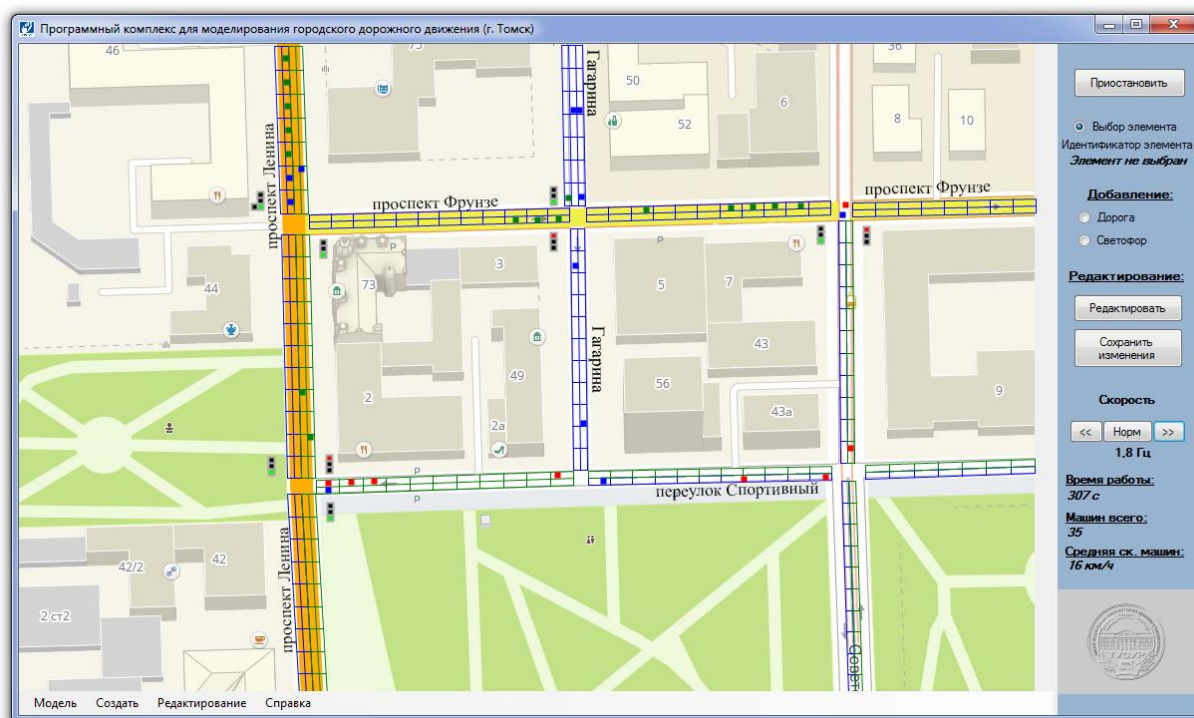


Рисунок 8.9 - Пример работы программы.

Реализованный программный комплекс предусматривает построение модели пользователя на его участках дороги, с ее дальнейшим сохранением. Пример работы программы, описанный выше, был сохранен с целью демонстрации работы комплекса.

8.5 Анализ и сравнение функционала с разработанным продуктом

После обзорной части уже имеющихся на рынке программных продуктов, была составлена таблица, отображающая недостатки и достоинства аналоговых систем (табл. 8.1).

Таблица 8.1 – Анализ функционала аналогов разрабатываемому программному продукту.

№	Функционал	Дорожный менеджер	IndorRoad	Модель Димитровой	TomRoad
1	Возможность редактирования участка карты	+	+	-	+
2	Бесплатное использование ПО	-	-	+	+
3	Возможность регулирования работы светофоров (времени, циклов)	+	+	+	+
4	Наглядное отображение локальных целей автомобилей	+	-	-	+
5	Изменение скорости моделирования	+	-	+	+
6	Интуитивно понятный интерфейс	-	-	+	+
7	Производительность	-	-	+	+
8	Возможность добавления пользовательских карт	+	+	-	+
9	Задание периодов появления автомобилей на моделируемом участке	+	-	-	+
10	Масштабирование участков дорог	+	+	-	-
11	Моделирование движения автотранспорта на реальных данных	-	+	-	+
12	Построение пользовательских участков трассы	+	+	-	+

Существующие аналоги имеют ряд недостатков, которые можно выявить исходя из данных, приведённых в таблице 8.1. При проектировании системы "TomRoad" большое значение уделялось устранению вышеприведённых недостатков с целью повышения конкурентоспособности создаваемого продукта. Результатом подобного подхода к проектированию стало наличие следующих специфических качеств разработанного программного обеспечения:

- экономическая составляющая. Данный продукт предоставляется на безвозмездной основе и не требует платных подписок и платного обслуживания. Все обновления программного комплекса будут предоставляться пользователям на бесплатной основе после установки программного продукта;

- низкая ресурсозатратность. Программный продукт выполнен на основе графических примитивов, что сказывается на скорости обработки алгоритмов и отображения движения автотранспорта на модели.

- простой интерфейс. Программный продукт был разработан в минималистическом стиле. Интерфейс интуитивно понятен для простого пользователя, не нагружен элементами управления;

- удобство использования. Разделение автомобилей по цветам, которые характеризуют локальные цели автомобилей, позволяет пользователю анализировать поток. Автомобили, которые не могут достигнуть своей локальной цели, меняют цель на более выгодную для автомобилиста в данной ситуации;

- построение модели на реальных данных. Программный комплекс позволяет задать пользователю для каждой полосы движения: количество автомобилей, которые появляются на данной полосе в течение заданного промежутка времени, локальные цели автомобилей, режимы работы светофоров.

9 ОТЛАДКА И ТЕСТИРОВАНИЕ ПРОГРАММНОГО КОМПЛЕКСА

Для завершения реализации программного комплекса было проведено комплексное тестирование, а именно:

- проверка работоспособности всех модулей и алгоритмов программы;
- проверка работоспособности программного модуля на разных рабочих машинах и операционных системах;
- корректное отображение объектов на экране;
- корректное отображение выводимой информации на экран пользователя.

Для проверки работоспособности всех модулей, каждый из них был запущен в составе программного комплекса через меню разделов.

Для проверки работоспособности модуля в целом, он был запущен на разных рабочих машинах с разными техническими характеристиками. Результаты тестирования приведены в таблице 8.1.

Таблица 9.1 – Зависимость работы программного комплекса (ПК) от технических характеристик рабочей машины (РМ)

№	Наименование РМ	Технические характеристики РМ	Замечания по результатам запуска программного комплекса
1	Ноутбук SAMSUNG NP-305V5A	Intel Core i5-2450M 2.50 GHz; 19.0" 1600x900; RAM 4 Gb, HDD 500 Gb, GeForce GT940M 1Gb, MS Windows 7	ПК показал высокую устойчивость в работе, точность выполнения предписанных функций, высокую скорость загрузки программы, эффективность и бесшбойность. Выводимые на экран подсказки и предупреждения выводятся корректно. Программа адекватно реагирует на действия пользователя.
2	Стационарный ПК	Intel Core i5-2550 3.5 GHz; 23.0" 1920x1080; RAM 12 Gb, HDD 1 Tb, NVIDIA GeForce GTX 1050 Ti 2Gb, MS Windows 7	ПК показал высокую устойчивость в работе, точность выполнения предписанных функций, высокую скорость загрузки программы, эффективность и бесшбойность. Выводимые на экран подсказки и предупреждения выводятся корректно. Программа адекватно реагирует на действия пользователя.

Продолжение Таблицы 9.1

3	Стационарный ПК каф. АСУ	Intel Core 2Quad Q9300 2.5GHz 20" 1680x1050; RAM 4 Gb, SSD 128 Gb, HDD 1 Tb, GeForce GTX 7500, Ubuntu 14.04 Mate	ПК показал функциональную полноту и законченность реализуемых функций обработки, адаптивный интерфейс
4	Стационарный ПК каф. АСУ	Intel Core i5–2320 3GHz; 18.5" 1366x768; RAM 3.42 Gb, HDD 320 Gb, Intel HD, MS Windows XP	Запуск ПК требует большего времени, в виду нехватки вычислительной мощности компьютера.
5	Стационарный ПК каф. АСУ	Intel Core 2 CPU 6400 2.13 GHz; 19" 1280x1024; RAM 1 Gb, HDD 160 Gb, Intel HD, MS Windows XP	Запуск ПК требует большего времени, в виду нехватки вычислительной мощности компьютера.

В ходе выполнения тестирования, были выявлены следующие ошибки:

- при отсутствующем файле с сохраненными данными пользователя (map.map), не выдается ошибка с предупреждением. Программный комплекс запускается в стартовом режиме, загружается пустое окно с картой города;
- при загрузке файла карты с расширением, отличным от *.jpg, программный комплекс выдаёт программное исключение. Дальнейшая работа с программой невозможна;
- при создании пустого участка дороги, такое возникает, когда пользователь указывает начальную и конечную точку формирования трассы в одной координаты, другими словами, происходит одинарное нажатие, комплекс создает невидимый участок. Работа комплекса протекает в обычном режиме;
- программа игнорирует проверку на ввод нулевых параметров, выполняя некорректный прогон алгоритмов.
- при ручном вводе данных в файл с загружаемыми параметрами, заданными пользователем, программа выдает исключение. Дальнейшая работа с комплексом невозможна;

- при остановке экспериментальных данных, результаты предыдущих измерений не фиксируются пользователем. В связи с данной ошибкой, дальнейший анализ данных всех прогонов программы произвести невозможно.

Все вышеописанные ошибки были устранены. При повторной проверке на корректность расчетов всех алгоритмов, результаты вычислений совпали с тестовыми значениями. Программный комплекс соответствует предъявляемым к разработке требованиям в разделе 7.1, обладает интерактивными элементами, эргономичным интерфейсом, наличием достаточного справочного материала, выполняет корректную загрузку данных пользователей, осуществляет выдачу сообщений при возникновении ошибок с описанием, корректно осуществляет работу всех алгоритмов описанных в разделах 5-6. Обеспечена работа программного комплекса без установки дополнительного ПО, запуск может быть осуществлен в учебных аудиториях на любом персональном компьютере, соответствующем требованиям к программному обеспечению, приведенному в разделе 6.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной магистерской диссертации были решены поставленные задачи и получены следующие результаты:

- 1) изучена предметная область;
- 2) сделана постановка задачи магистерской диссертации;
- 3) спроектирован программный комплекс с учетом всех предъявляемых требований;
- 4) написан программный код;
- 5) проведена отладка и тестирование программного комплекса;
- 6) разработана документация программного продукта.

Для завершения реализации программного продукта было проведено комплексное тестирование: дополнен иллюстрационный материал, внесены коррективы в интерфейс, устранены незначительные недоработки, предложено решение по увеличению эффективности программного комплекса.

В результате выполнения вышеуказанных задач, была успешно достигнута основная цель работы, а именно: реализован программный комплекс, который соответствует предъявляемым к разработке требованиям в разделе 7.1, обладает интерактивными элементами, интуитивно-понятным интерфейсом, позволяет построить модель движения транспорта. Информация доступна, читабельна, адаптирована. Обеспечена работа программного комплекса без установки дополнительного ПО, запуск может быть осуществлен на любом персональном компьютере, соответствующем требованиям к программному обеспечению, приведенному в разделе 7.1.

В результате проведенных исследований, были разработаны алгоритмы управления потоком автотранспорта в условиях городской среды, на основании которых может быть реализован программный комплекс. Данный комплекс позволит принять решения о перепланировке перекрестков, расширении проезжей части улицы и др., что в итоге уменьшает вероятность появления заторов на улицах городов. Таким образом, была достигнута основная цель -

создание программного комплекса моделирования дорожного движения, который был упомянут во введении и, как следствие, цель магистерской диссертации была достигнута в полном объёме.

Пояснительная записка выполнена в соответствии с ОС ТУСУР 01–2013 «Работы студенческие по направлениям подготовки и специальностям технического профиля» [21].

СПИСОК ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ

- АТС – автотранспортное средство;
- АСУ – автоматизированные системы управления;
- ВКР – выпускная квалификационная работа;
- МД – магистерская диссертация;
- ПО – программное обеспечение;
- РМ – рабочая машина;
- СМО – система массового обслуживания;
- ТЗ – техническое задание;
- ТС – транспортная сеть;
- УДС – улично-дорожная сеть;
- ЦИАИ – центр исследований автодорожной инфраструктуры;
- ЭВМ – электронно-вычислительная машина;
- IDE – интегрированная среда разработки.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Nagel K., Schreckenberg M. A cellular automaton model for freeway traffic // J. Phys. I France. 1992. Vol. 2. Pp. 2221-2229.
- 2 М. А. Трапезникова, А. А. Чечина, Н. Г. Чурбанова, «Описание динамики транспортных потоков на элементах улично-дорожной сети с использованием двумерных математических моделей». Препринты ИПМ им. М. В. Келдыша, 2016, 093, 20 с.
- 3 Государственный доклад от 11 октября 2003 г. N 181 «О состоянии безопасности дорожного движения в Российской Федерации» [электронный ресурс]. – Режим доступа: <http://docs.cntd.ru/document/901873810> , свободный. – Загл. с экрана (дата обращения: 10.04.18).
- 4 Отчет о проведении Международного семинара "Решение проблем организации автомобильного движения в центре Москвы" (г. Москва, 20-21 февраля 2002 г.) [электронный ресурс]. – Режим доступа: <http://docs.cntd.ru/document/90187345> , свободный. – Загл. с экрана (дата обращения: 10.04.18).
- 5 Информационный – образовательный портал Ханадеевой Е.А. Понятия и виды потоков [электронный ресурс]. – Режим доступа: http://www.hanadeeva.ru/logictica/lekci/vopros_1_8/index.html , свободный. – Загл. с экрана (дата обращения: 10.04.18).
- 6 Проект ПРООН RUS/97/029 „Стратегия развития транспорта. Программа действий и кратко/среднесрочных инвестиций" [электронный ресурс]. – Режим доступа: <http://docs.cntd.ru/document/90187345> , свободный. – Загл. с экрана (дата обращения: 12.05.18).
- 7 Методы исследования дорожного движения [электронный ресурс]. – Режим доступа: https://studopedia.ru/9_56781_dorogi-i-ih-klassifikatsiya.html, свободный. – Загл. с экрана (дата обращения: 12.05.18).
- 8 Большая биографическая энциклопедия Дубелир Георгий Дмитриевич [электронный ресурс]. – Режим доступа:

<http://www.biografija.ru/biography/dubelir-georgij-dmitrievich.htm>, свободный. – Загл. с экрана (дата обращения: 16.05.18).

9 Уизем Дж. Линейные и нелинейные волны : Учебное пособие. – М.: Мир, 1977 г. , 300 с. [электр. ресурс]. – Режим доступа: http://citforum.ck.ua/book/c_net/otryvok.shtml, свободный. – Загл. с экрана (дата обращения: 10.04.18).

10 Lighthill Institute of Mathematical Sciences. Sir James Lighthill [электронный ресурс]. – Режим доступа: <http://www.ucl.ac.uk/lims/jameslighthill.htm>, свободный. – Загл. с экрана (дата обращения: 16.05.18).

11 Хейт Ф. Математическая теория транспортных потоков : Учебное пособие. – М.: Мир, 1966 г. - 286 с. [электр. ресурс]. – Режим доступа: http://citforum.ck.ua/book/c_net/otryvok.shtml, свободный. – Загл. с экрана (дата обращения: 10.04.18).

12 Смирнов Н.Н., Киселев А.Б., Никитин В.Ф., Юмашев М.В. Математическое моделирование автотранспортных потоков: Учебное пособие. – М.: мех-мат МГУ, 1999 г. - 286 с. [электр. ресурс]. – Режим доступа: http://citforum.ck.ua/book/c_net/otryvok.shtml, свободный. – Загл. с экрана (дата обращения: 10.04.18).

13 Nagel, K.; Schreckenberg, M. A cellular automaton model for freeway traffic // J. Phys. I France. – 1992. – V. 2. – P. 2221-2229.

14 Информационно-образовательный портал Ханадеевой Е.А. Понятия и виды потоков: Учебное пособие. – М.: "Питер", 2013 г. , 800 с. [электр. ресурс]. – Режим доступа: http://www.hanadeeva.ru/logictica/lekui/vopros_1_8/index.html, свободный. – Загл. с экрана (дата обращения: 10.04.18).

15 Троелсен Э. С# и платформа .NET. Библиотека программиста: Учебное пособие. – М.: "Питер", 2013 г. , 800 с. [электр. ресурс]. – Режим доступа: http://citforum.ck.ua/book/c_net/otryvok.shtml, свободный. – Загл. с экрана (дата обращения: 10.04.18).

16 Microsoft. Developer Network. Элементы управления Windows Forms [электронный ресурс]. – Режим доступа: [https://msdn.microsoft.com/ru-ru/library/ettb6e2a\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/ettb6e2a(v=vs.110).aspx), свободный. – Загл. с экрана (дата обращения: 10.04.18).

17 Microsoft. Developer Network. Интегрированная среда разработки Visual Studio 2015 [электронный ресурс]. – Режим доступа: <https://msdn.microsoft.com/ru-ru/library/dn762121.aspx>, свободный. – Загл. с экрана (дата обращения: 10.04.18).

18 Mallenom Systems «Дорожный менеджер» [электронный ресурс]. – Режим доступа: <http://www.mallenom.ru/products/modelirovanie-i-iskusstvennyi-intelekt/dorozhnyi-menedzher/>, свободный. – Загл. с экрана (дата обращения: 05.06.18).

19 ИндорСофт – разработка программного обеспечения для проектирования, строительства, эксплуатации автомобильных дорог и электрических сетей. «IndorRoad» [электронный ресурс]. – Режим доступа: <https://www.indorsoft.ru/products/road/>, свободный. – Загл. с экрана (дата обращения: 05.06.18).

20 Anylogic: имитационное моделирование для бизнеса [электронный ресурс]. – Режим доступа: <https://www.anylogic.ru/>, свободный. – Загл. с экрана (дата обращения: 05.06.18).

21 ОС ТУСУР 01–2013 «Работы студенческие по направлениям подготовки и специальностям технического профиля» Общие требования и правила оформления – Томск: ТУСУР, 2013. – 57 с.

ПРИЛОЖЕНИЕ А**(справочное)****Список научных трудов по теме МД****СПИСОК****научных трудов Разумова Дмитрия Сергеевича**

№ пп	Название труда, открытия, изобретения	Печатный или на правах рукописи	Издательство, журнал (название, номер, год) или номер авторского свидетельства	Год издания	Кол-во печат. листов или страниц	Примечание (указать соавторов)
1	Программный комплекс моделирования городского дорожного движения.	печ.	XV Международная научно-практическая конференция "Eurasiascience", 31 мая 2018 года. СЕКЦИЯ 07. Технические науки.	2018	5	Катаев М.Ю., Шелестов А.А.
2	Алгоритмы управления потоком автотранспорта в городских условиях	печ.	XXV Международная научно-практическая конференция «Вопросы современных научных исследований»	2018	6	Катаев М.Ю., Шелестов А.А.

Соискатель (подпись) _____ Разумов Д.С.

ПРИЛОЖЕНИЕ Б

(справочное)

Список научных конференций



Рисунок П.1 - Диплом участника XV Международной научно-практической конференции «Eurasiascience».



Рисунок П.2 - Сертификат участника Международная научно-практическая конференция «Вопросы современных научных исследований».

ПРИЛОЖЕНИЕ В

(справочное)

Алгоритмы работы программы

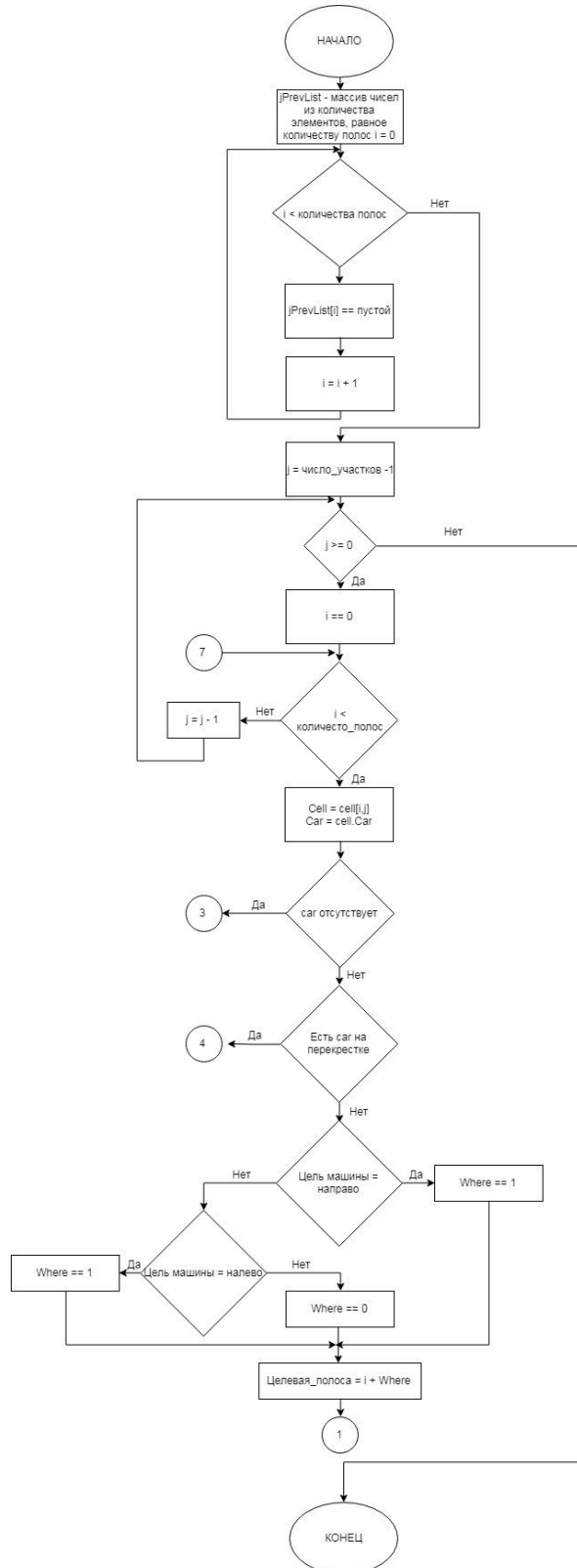


Рисунок В.1 – Алгоритм смены полосы.

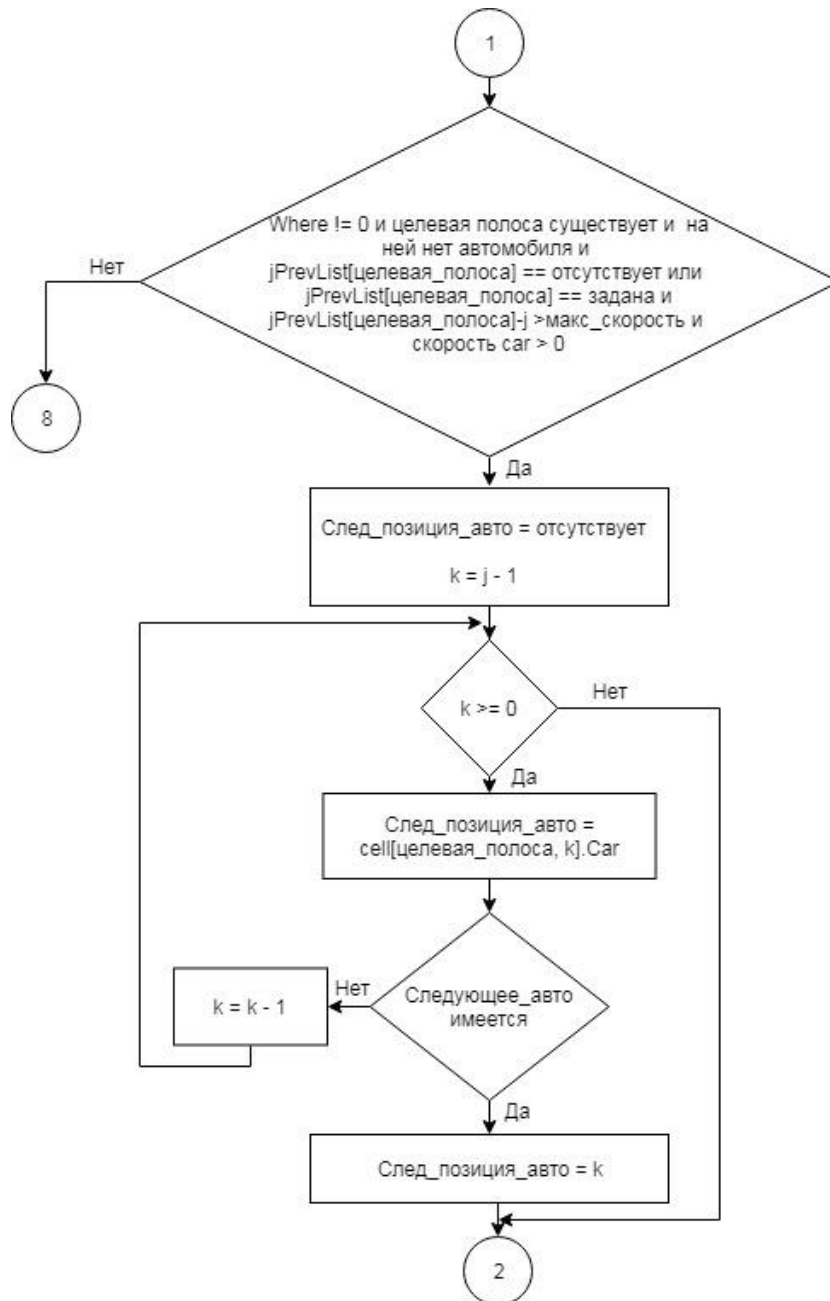


Рисунок В.2 – Алгоритм вычисления позиции автомобиля.

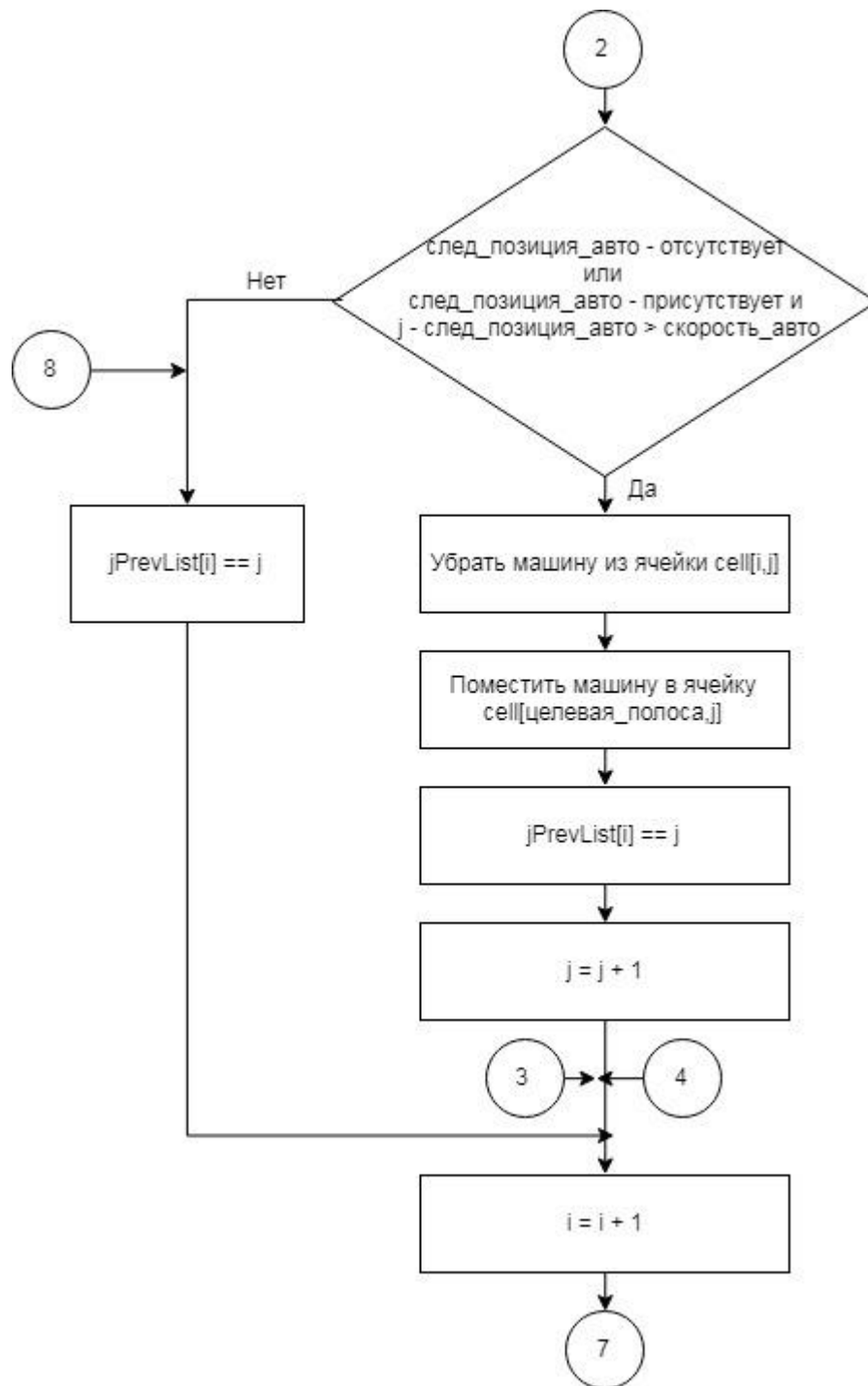


Рисунок В.3 – Алгоритм добавления автомобиля на трассу.

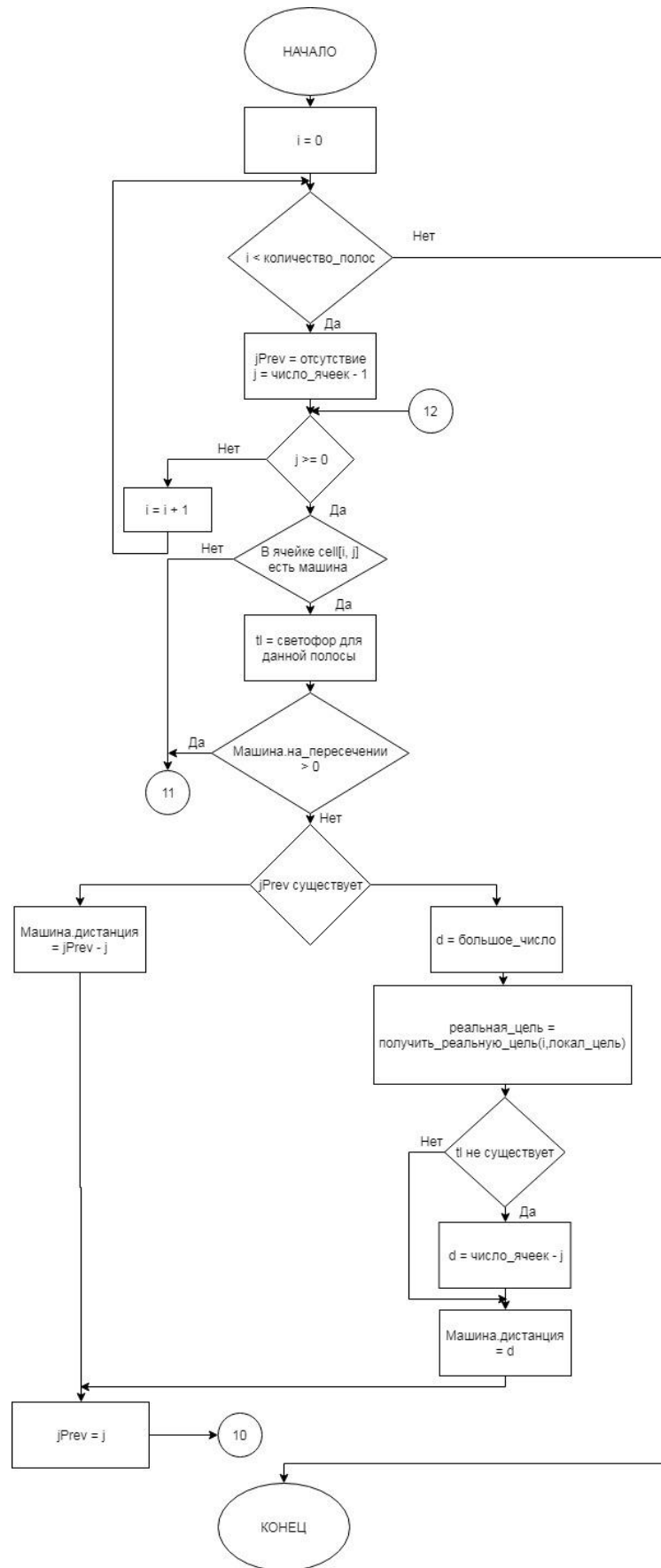


Рисунок В.4 – Алгоритм движения по полосе.

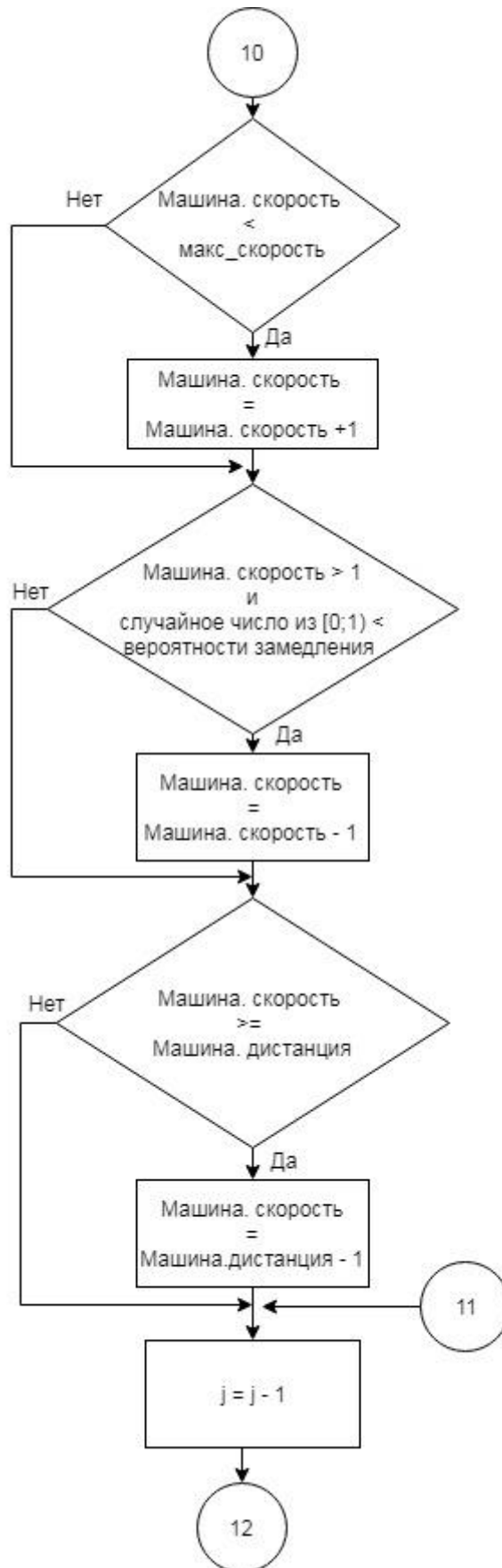


Рисунок В.5 – Алгоритм изменения скорости автомобиля.

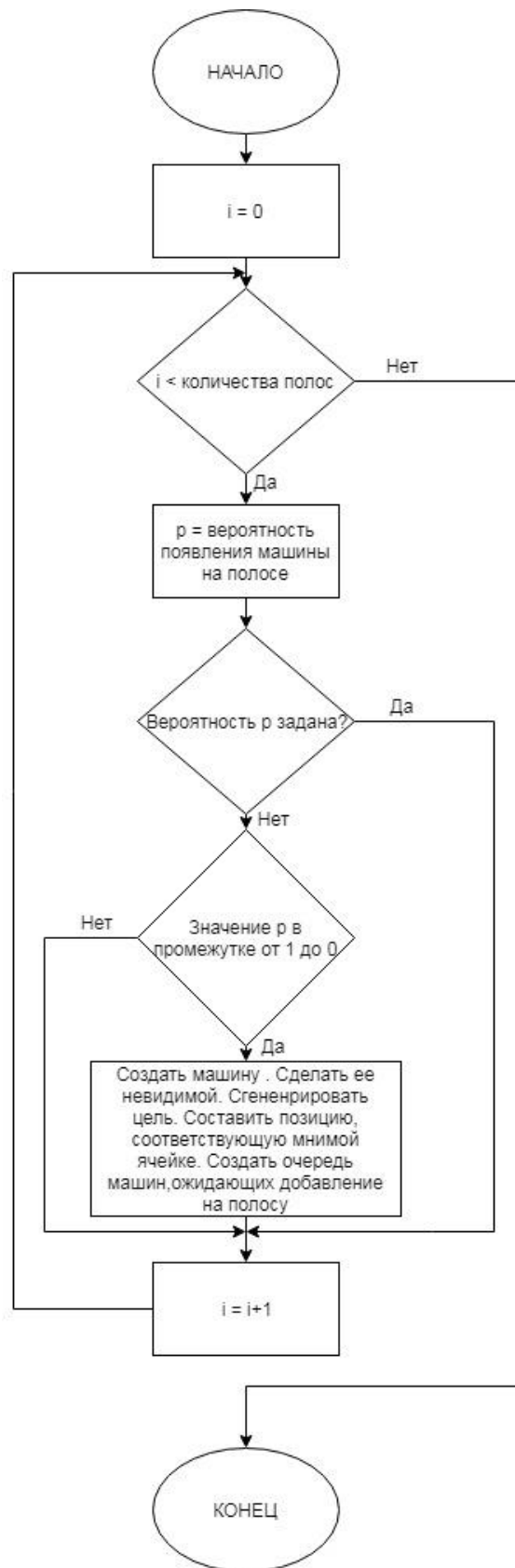


Рисунок В.6 – Алгоритм генерации автомобиля.

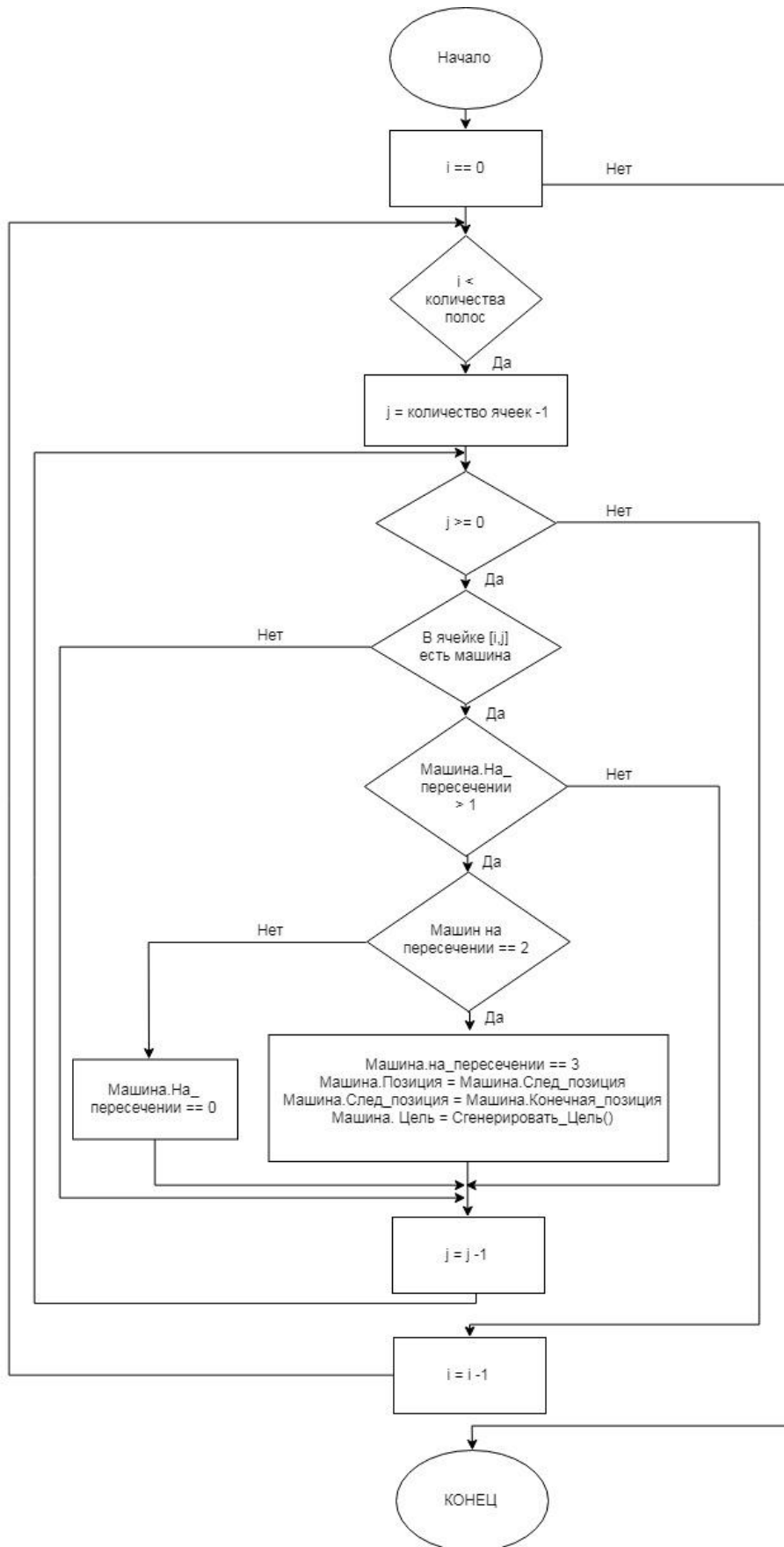


Рисунок В.7 – Алгоритм шага проезда через перекрёсток.

ПРИЛОЖЕНИЕ Д

(справочное)

Руководство пользователя

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)**

Кафедра автоматизированных систем управления (АСУ)

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «ТОМROAD»

Содержание

1. ВВЕДЕНИЕ.....	84
1.1 Область применения.....	84
1.2 Краткое описание возможностей.....	84
1.3 Уровень подготовки пользователя	85
1.4 Перечень эксплуатационной документации	85
2 НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ.....	86
2.1 Виды деятельности, функции	86
2.2 Программные и аппаратные требования к системе	86
3 ПОДГОТОВКА К РАБОТЕ	87
3.1 Состав дистрибутива	87
3.2 Запуск системы	87
3.3 Проверка работоспособности системы	88
4 ОПИСАНИЕ ОПЕРАЦИЙ.....	89
4.1 Наименование операции	89
4.2 Условия выполнения операции.....	89
4.3 Подготовительные действия	89
4.4 Заключительные действия	89
4.5 Ресурсы, расходуемые на операцию.....	90
5 АВАРИЙНЫЕ СИТУАЦИИ. ВОССТАНОВЛЕНИЕ СИСТЕМЫ	91
6 РЕКОМЕНДАЦИИ ПО ОСВОЕНИЮ	92
6.1 Описание функционала программного комплекса	92
6.1.1 Загрузка исходных данных для моделирования	92
6.1.2 Создание пользовательской модели	93
6.1.3 Загрузка существующей модели.....	95
6.1.4 Создание участка дороги на модели.....	97

6.1.5 Создание светофора на модели	100
6.1.6 Редактирование участка дороги	103
6.1.7 Редактирование светофора	104
6.1.8 Привязка светофора к полосе движения.....	105
6.1.9 Перестроение автомобилей на другой участок дороги.....	107
6.1.10 Сохранение пользовательских изменений	108
6.2 Завершение работы с программой	109

1. ВВЕДЕНИЕ

1.1 Область применения

Программное обеспечение (ПО) «TOMROAD» предназначено для моделирования движения автотранспорта в условиях городской среды с возможностью использовать результаты измерений для дальнейшей работы.

Программное обеспечение «TOMROAD» направлено на применения результатов данной работы различными предприятиями города, чья профессиональная деятельность связана непосредственно с автодорожной сетью города. Программный комплекс будет эффективным для организации для проведения экспериментов с целью выявления пропускной способности участков дорог города, исходя из реальных показателей на данный момент времени. Повышения экономической эффективности, достигаемой за счёт своевременного анализа получения экспериментальной информации, что может повлиять на дальнейший ход реализации проекта.

1.2 Краткое описание возможностей

Программное обеспечение «TOMROAD» выполняет следующие функции:

- просмотр справочной информации о разработчике и версии программы;
- просмотр справочной информации о работе с программным комплексом;
- построение имитированной дорожной сети;
- создание объектов «светофор»;
- задание параметров нагрузки на каждом участке дорог;
- привязка дорог к фазам работы светофоров;
- загрузка сохраненных данных из текстового файла;
- ручной ввод данных для работы алгоритмов;
- расчёт алгоритмов, представленных в программе;

- изменение начальных параметров;
- вывод на экран результатов работы алгоритмов;
- сохранение результатов работы программы в текстовый файл.

1.3 Уровень подготовки пользователя

Программное обеспечение «TOMROAD» должен обладать минимальными навыками пользования компьютером, а именно:

- использование мыши и клавиатуры;
- работа со справочной информацией;
- выборка файлов.

1.4 Перечень эксплуатационной документации

Перечень эксплуатационных документов, с которым необходимо ознакомиться:

- «TOMROAD» "Руководство пользователя".

2 НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ

2.1 Виды деятельности, функции

ПО «TOMROAD» предназначено для автоматизации следующих видов деятельности:

- формирование потоков автомобилей на реальных данных;
- регулирование фаз работы светофоров, основываясь на сформированном потоке автомобилей;
- расчёта пропускной способности перекрестков.

2.2 Программные и аппаратные требования к системе

- CPU Intel Core i3 3.4 GHz /1024 Mb;
- HDD 120Gb SATA;
- графическая карта Intel® HD Graphics Family с актуальными драйверами;
- ОС MS Windows 7/8/10;
- компьютерная мышь/тачпад;
- клавиатура.

3 ПОДГОТОВКА К РАБОТЕ

3.1 Состав дистрибутива

В состав дистрибутива ПО «TOMROAD» входит:

- Приложение установки программного обеспечения setup.exe;
- Дистрибутив системы ПО «TOMROAD»;
- Документацию к программному продукту;
- Файлы с сохраненными значениями для построения пользовательской трассы в формате (*.map);

3.2 Запуск системы

1. Перейдите в дистрибутив, содержащий файлы TOMROAD.exe
2. Перед Вами откроется рабочее окно программы (Рисунок 3.1).

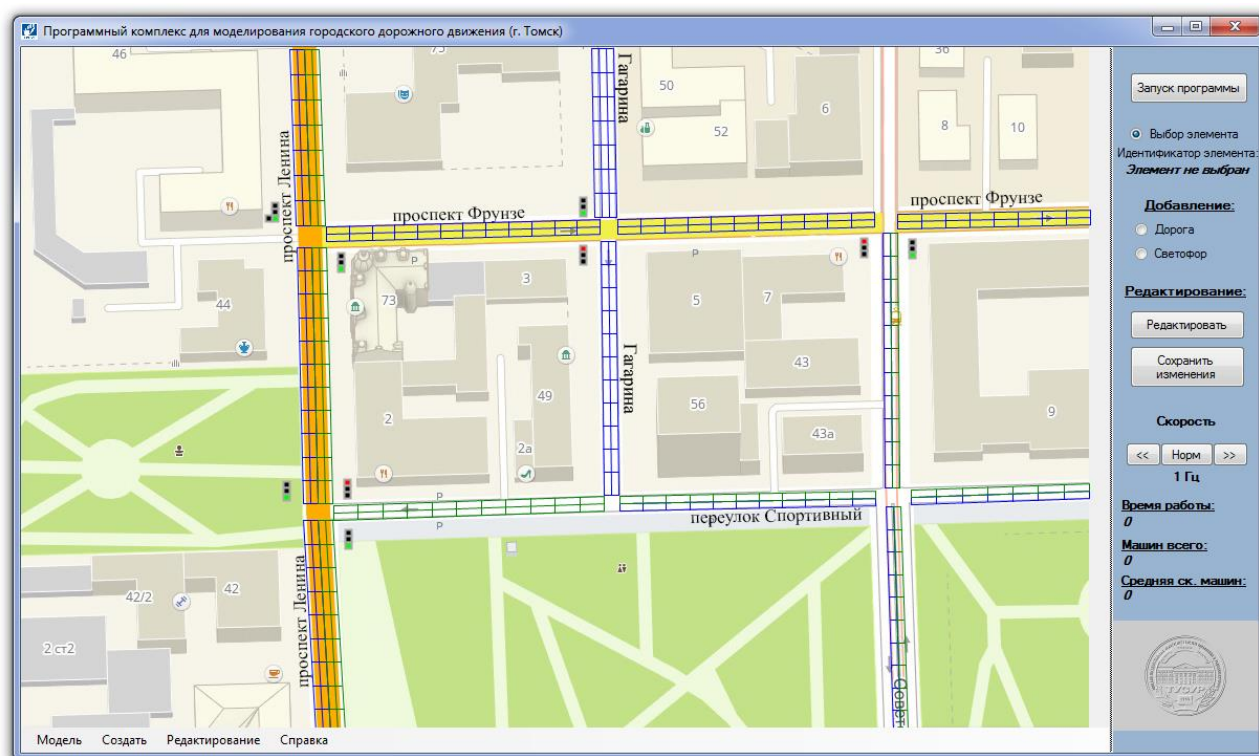


Рисунок 3.1 – Окно клиентского приложения.

В случае необходимости установки к персональному компьютеру, предварительно необходимо выполнить установку системы.

3.3 Проверка работоспособности системы

Программное обеспечение работоспособно, если в результате действий пользователя, изложенных в п.п.4.2, на экране монитора отобразилось главное окно клиентского приложения без выдачи пользователю сообщений о сбое в работе.

4 ОПИСАНИЕ ОПЕРАЦИЙ

4.1 Наименование операции

- загрузка исходных данных из файла;
- ручной ввод данных;
- изменение начальных параметров;
- алгоритм однополосного движения;
- алгоритм смены полосы;
- алгоритм движения по многополосному участку;
- алгоритм задания граничных условий;
- алгоритм прохождения регулируемого перекрестка;
- вывод на экран результатов работы алгоритмов;
- сохранение результатов работы программы в текстовый файл;
- просмотр справочной информации о разработчике и версии программы;
- просмотр справочной информации о работе с программным комплексом;
- просмотр справочной информации об алгоритмах, описанных в программе.

4.2 Условия выполнения операции

Для нормального функционирования ПО «TOMROAD» необходим доступ к файлам или сетевым ресурсам, на которых расположены файлы для работы программы (map.map, map.jpg). Для сохранения результатов выполнения алгоритма, необходимы права администратора. Построение модели и выполнение алгоритмов выполняется при условии, что был загружен файл с корректными значениями. Примеры участков дорог с корректными значениями находятся в поставляемом дистрибутиве (map.map, map.jpg).

4.3 Подготовительные действия

Отсутствуют.

4.4 Заключительные действия

После завершения работы с приложением, закрыть основное окно.

4.5 Ресурсы, расходуемые на операцию

Отсутствуют.

5 АВАРИЙНЫЕ СИТУАЦИИ. ВОССТАНОВЛЕНИЕ СИСТЕМЫ

При сбое в работе аппаратуры восстановление нормальной работы системы должно производиться после:

- перезагрузки операционной системы;
- запуска исполняемого файла системы.

При ошибках в работе аппаратных средств (кроме носителей данных и программ) восстановление функции системы возлагается на ОС. При ошибках, связанных с программным обеспечением (ОС и драйверы устройств), восстановление работоспособности возлагается на ОС. При неверных действиях пользователей, неверных форматах или недопустимых значениях входных данных, система выдает пользователю соответствующие сообщения, после чего возвращается в рабочее состояние.

6 РЕКОМЕНДАЦИИ ПО ОСВОЕНИЮ

Для успешного освоения приложения ПО «TOMROAD» необходимо иметь навыки работы с ПК и изучить настоящее «Руководство пользователя».

6.1 Описание функционала программного комплекса

Запустите систему. Для этого перейдите в папку, в которой находится программа. Выберите файл «TOMROAD.exe и кликните (нажмите) по нему 2 раза левой кнопкой мыши. При запуске программы откроется рабочее окно программы (Рисунок 6.1).

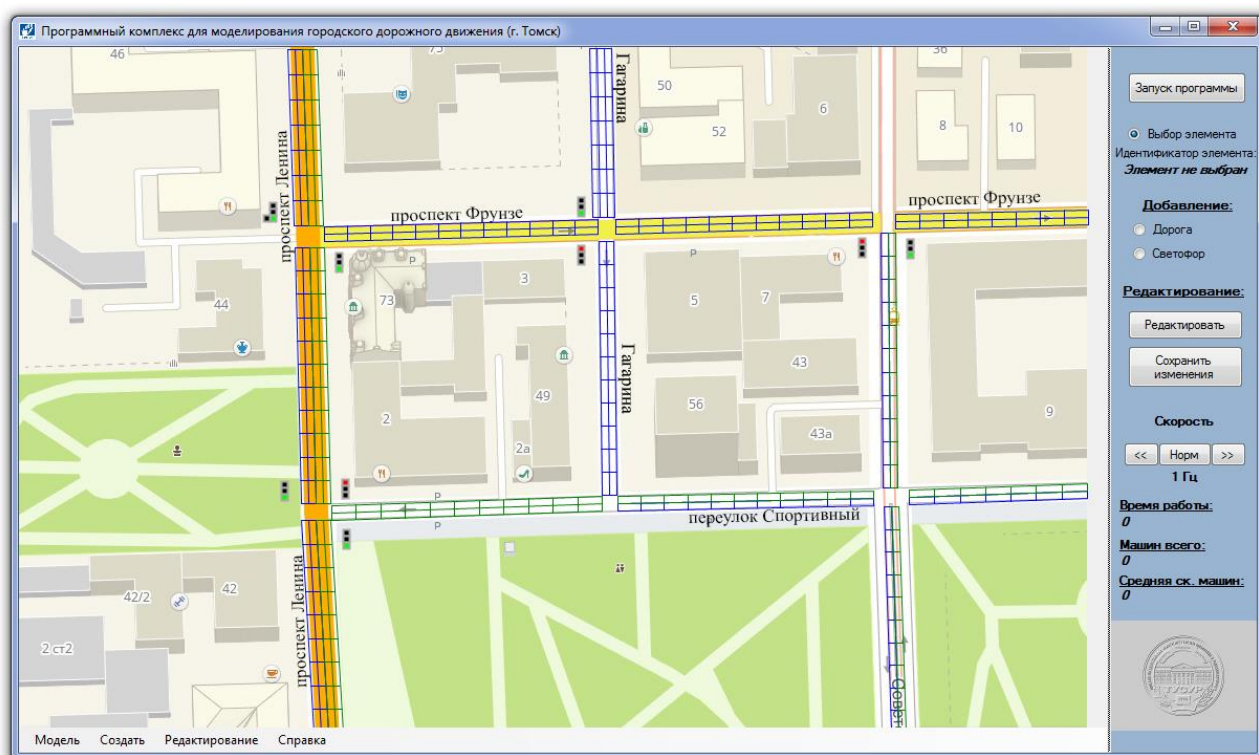


Рисунок 6.1 – Рабочее окно программы.

6.1.1 Загрузка исходных данных для моделирования

В случае, если необходимо осуществить ручной ввод данных в программу, выберите раздел «Модель – Создать новую модель». Для этого, дважды нажмите поле программы

ВНИМАНИЕ: Данные для работы алгоритмов различны. Для корректного ввода данных, ознакомьтесь с информацией, представленной в п.п. 4.2 настоящего руководства.

После чего, перед Вами откроется новое диалоговое окно для выбора файла с расширением (*.jpg) (Рисунок 6.2).

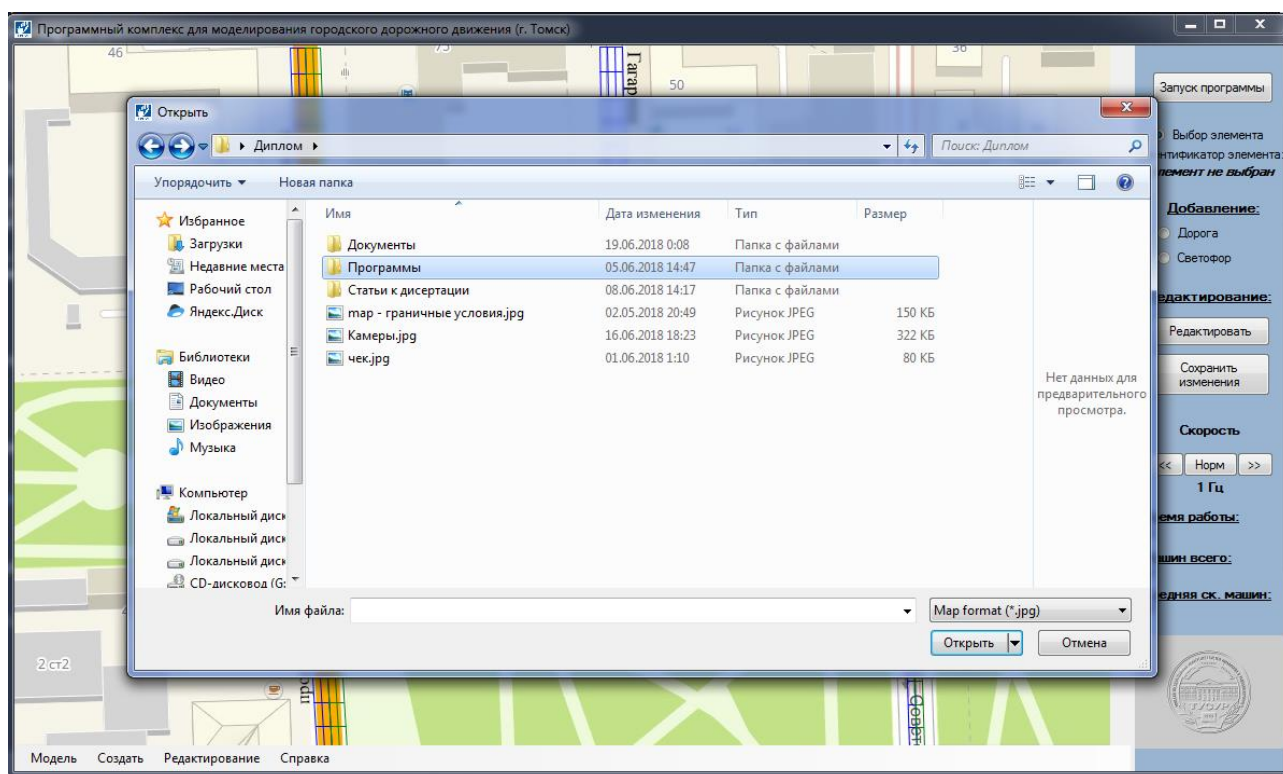


Рисунок 6.2– Окно выбора файла.

При выборе необходимого файла, нажмите кнопку «Открыть».

ВНИМАНИЕ: Для корректной работы программы должен быть выбран файл с расширением (*.jpg), для других расширений работа программы завершится неудачей и будет выведена соответствующая ошибка.

6.1.2 Создание пользовательской модели

В случае, если необходимо осуществить ручной ввод данных в программу, выберите раздел «Модель – Создать новую модель». Для этого, дважды нажмите поле программы. После, поле должно пометиться галочкой (Рисунок 6.3).

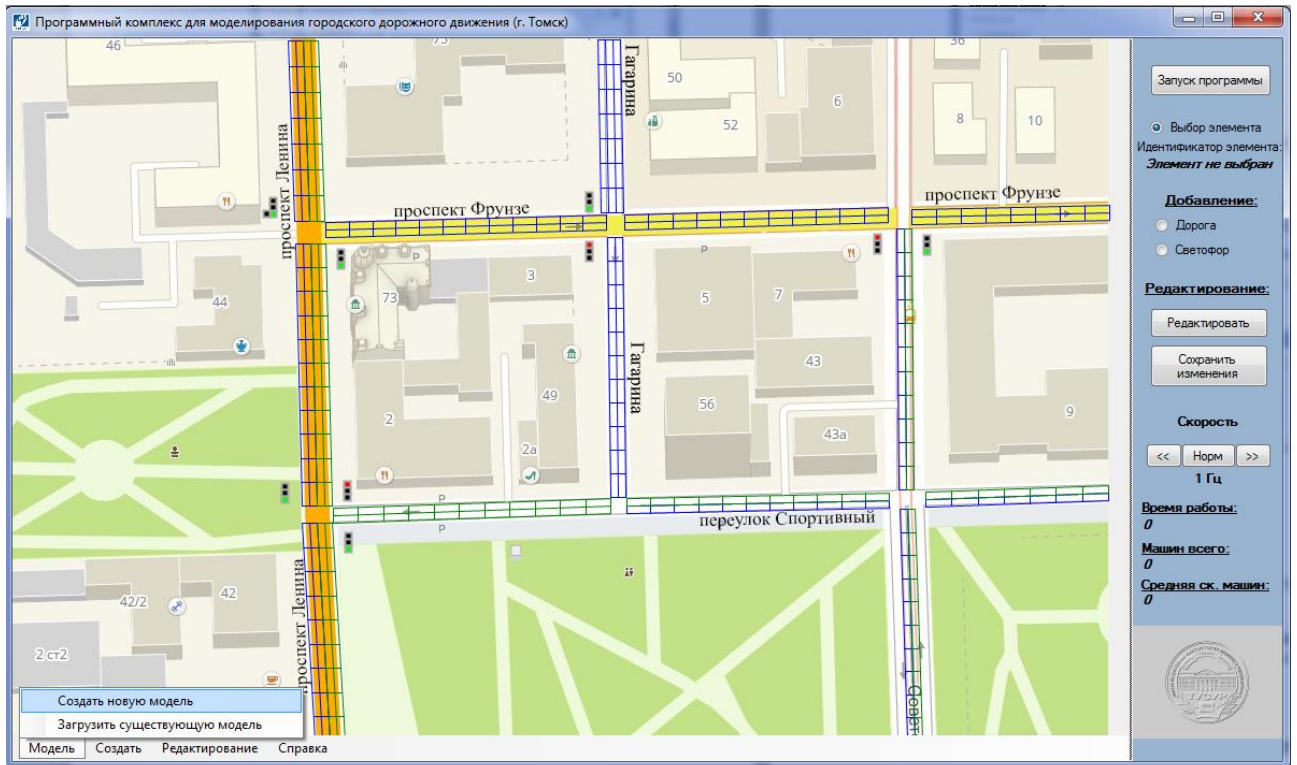


Рисунок 6.3–Выбор параметра «Создать модель».

В активном окне выберите файл с картой и подтвердите выбор. После, нажмите подтвердить. Откроется новая карта для создания новой модели. (Рисунок 6.4).

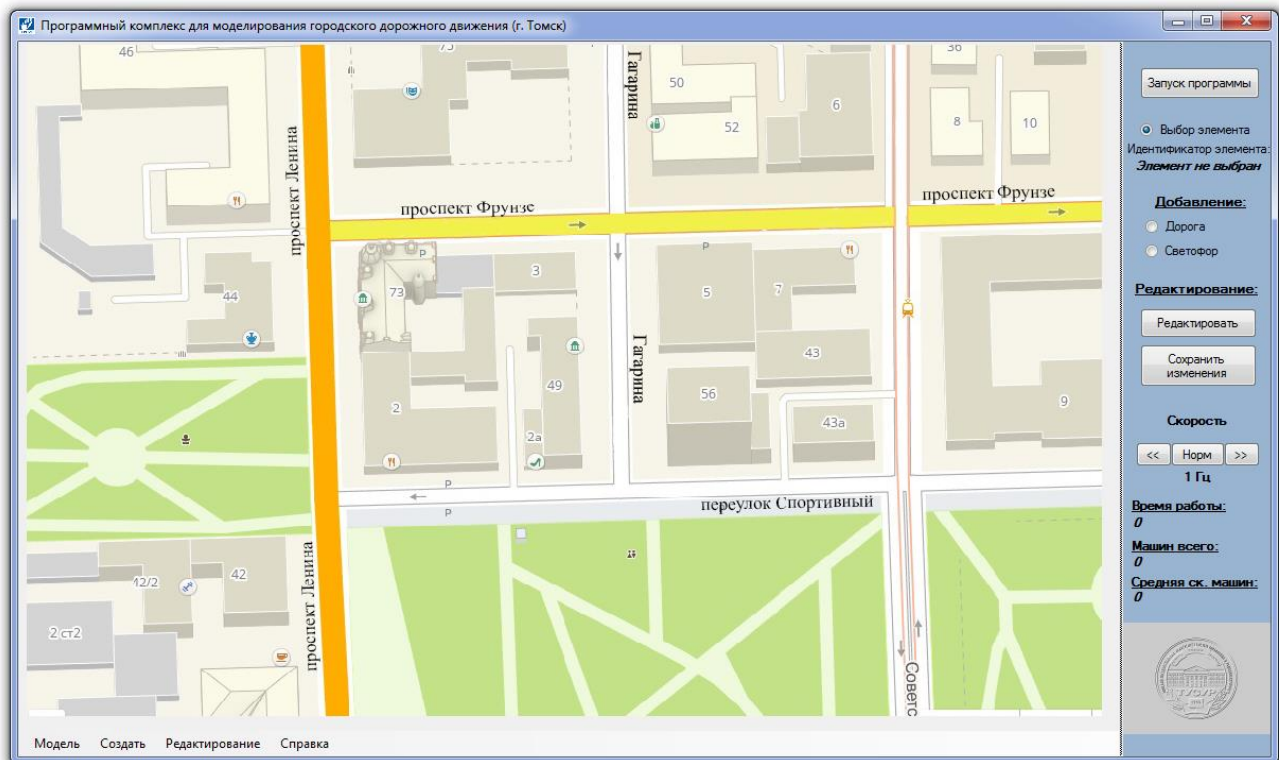


Рисунок 6.4–Загрузка пользовательской карты.

6.1.3 Загрузка существующей модели

В случае, если необходимо осуществить загрузку уже построенных моделей данных в программу, выберите раздел «Модель – Загрузить существующую модель» (Рисунок 6.5).

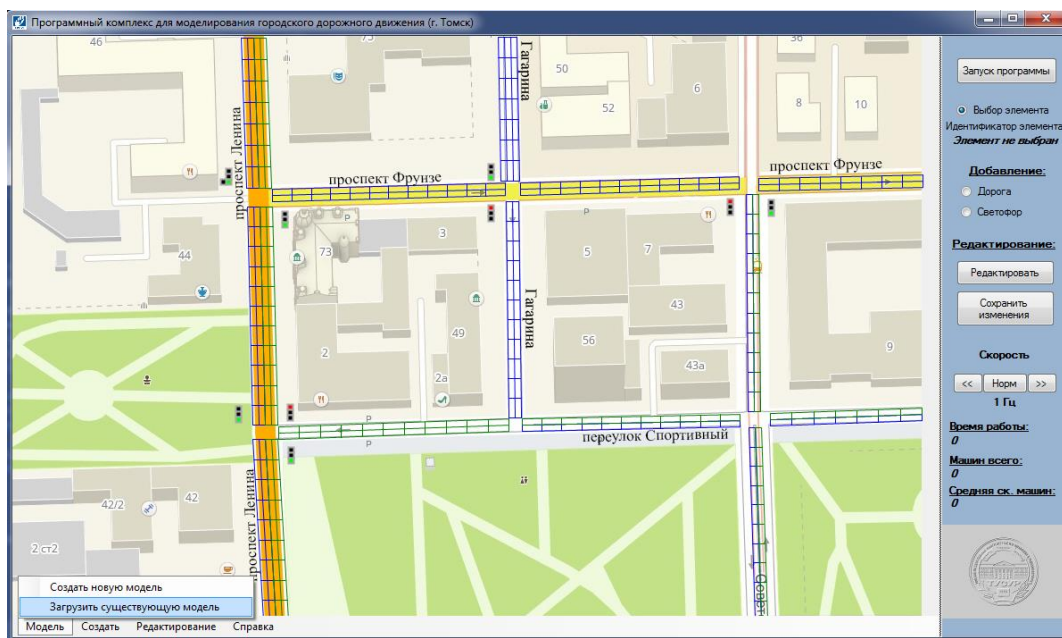


Рисунок 6.5 – Выбор параметра «Загрузить существующую модель».

В активном окне выберите файл с картой и подтвердите выбор. После, нажмите подтвердить. Откроется окно выбора файла (Рисунок 6.6).

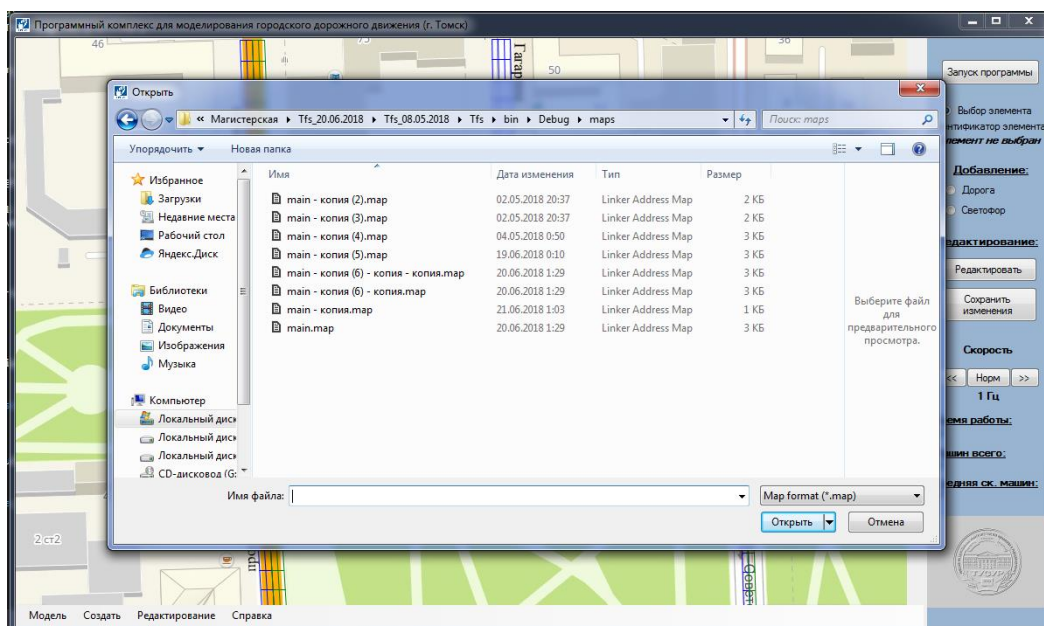


Рисунок 6.6 – Выбор параметра «Выбор файла карты существующей модели».

ВНИМАНИЕ: Для корректной работы программы должен быть выбран файл с расширением (*.mar), для других расширений работа программы завершится неудачей и будет выведена соответствующая ошибка.

После выполнения действий, описанных выше, перед пользователем откроется окно уже созданной модели (Рисунок 6.7).

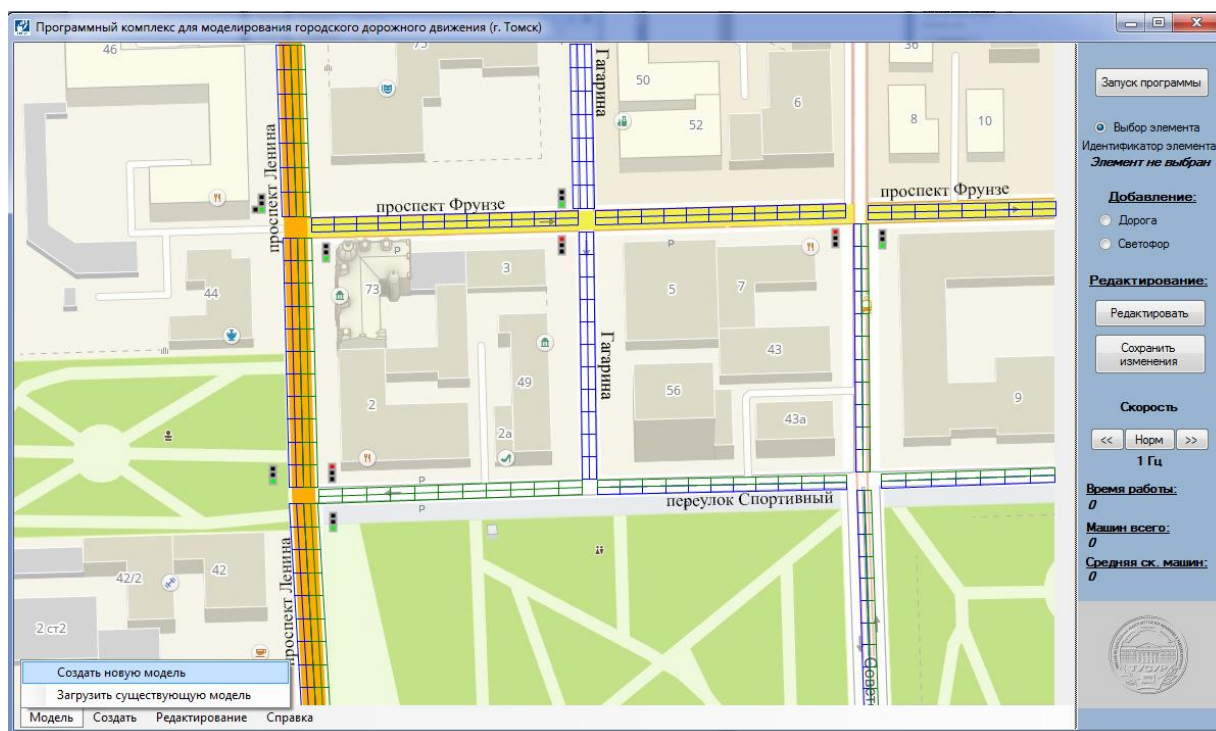


Рисунок 6.7 – Загрузка существующей карты.

6.1.4 Создание участка дороги на модели

Для создания участка дороги на модели, выберите компонент «Дорога» во вкладке создание. После чего, наведите указатель мыши на карту и нажмите левой кнопкой мыши на место, где будет располагаться начало участка дороги. После, не отпуская кнопки мыши, проведите курсором до места карты, где будет заканчиваться участок трассы. Перед вами откроется окно задания параметров (Рисунок 6.8).

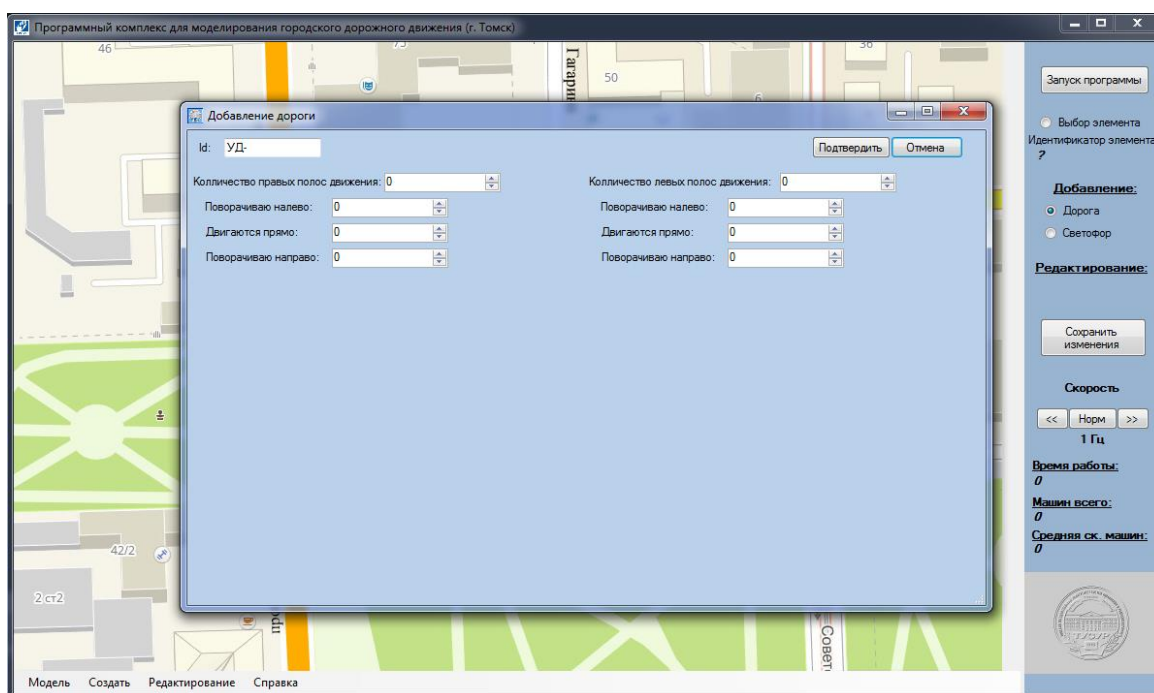


Рисунок 6.8 – Окно создания Участка дороги.

Далее, задаем уникальный идентификатор для участка дорог. Следующий шаг, задаем количество полос движения для каждой стороны (при отсутствии полос движения, оставляем значение 0). Левая часть окна отвечает за движение по левой полосе движения, правая часть окна – за движение по правой стороне. Правая сторона движения выделяется синим цветом. Встречная полоса движения выделяется зеленым цветом.

После указания количества полос, появится форма заполнения параметров каждой полосы (Рисунок 6.9).

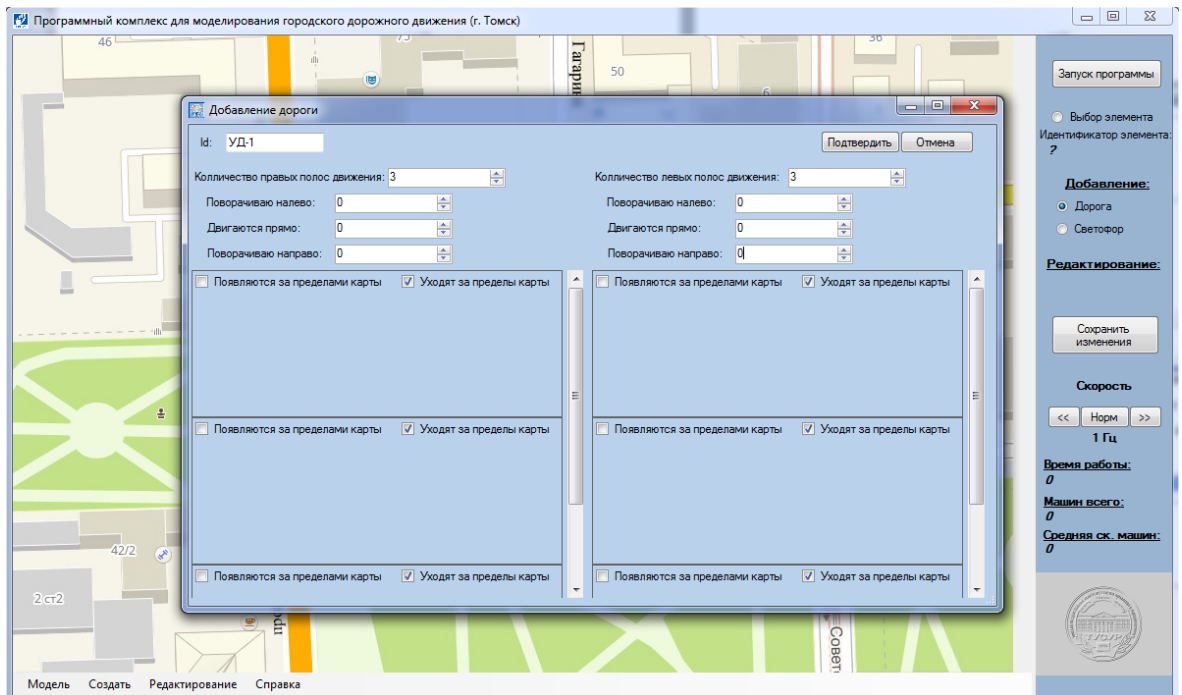


Рисунок 6.9 – Окно задания параметров для полос движения.

Далее, указываем, откуда должны появляться автомобили. Т.к. у нас создан только один участок трассы, указываем, что машины появляются за пределами карты, ставим галочку. Появятся поля с параметрами появления машин за промежуток времени. Указываем время и количество автомобилей (Рисунок 6.10).

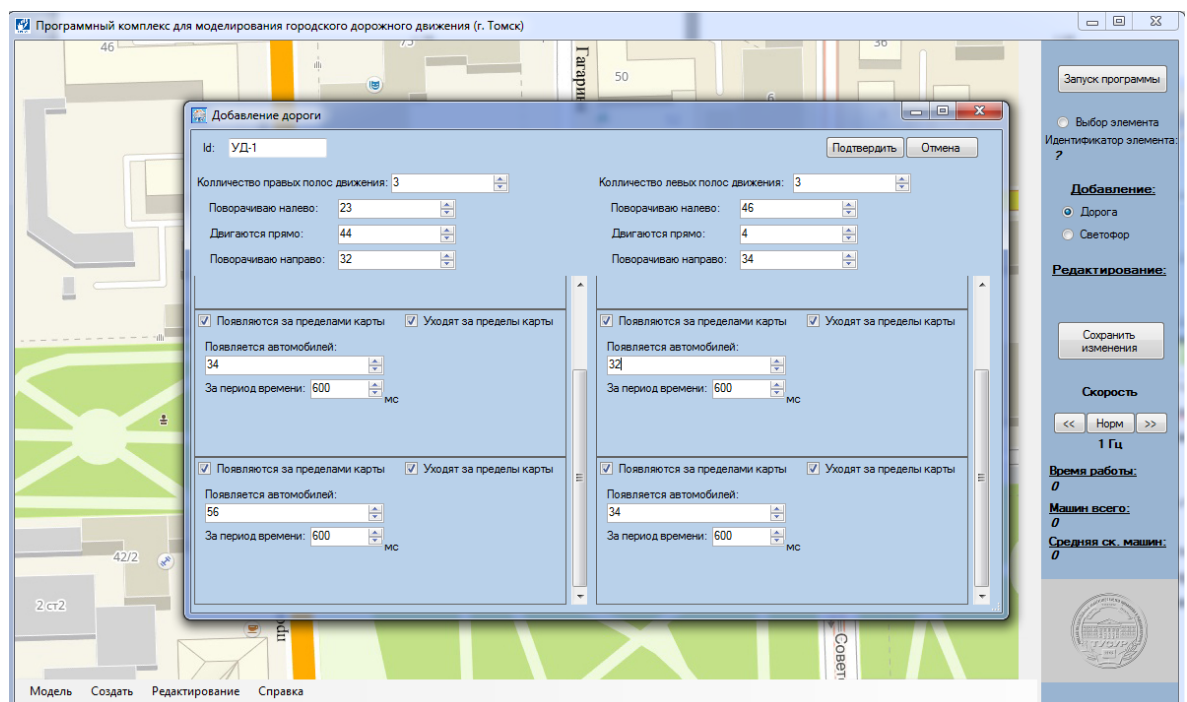


Рисунок 6.10 – Окно задания параметров для полос движения.

После заполнения всех необходимых полей, выбираем «Подтвердить». Участок трассы отобразится на карте (Рисунок 6.11).

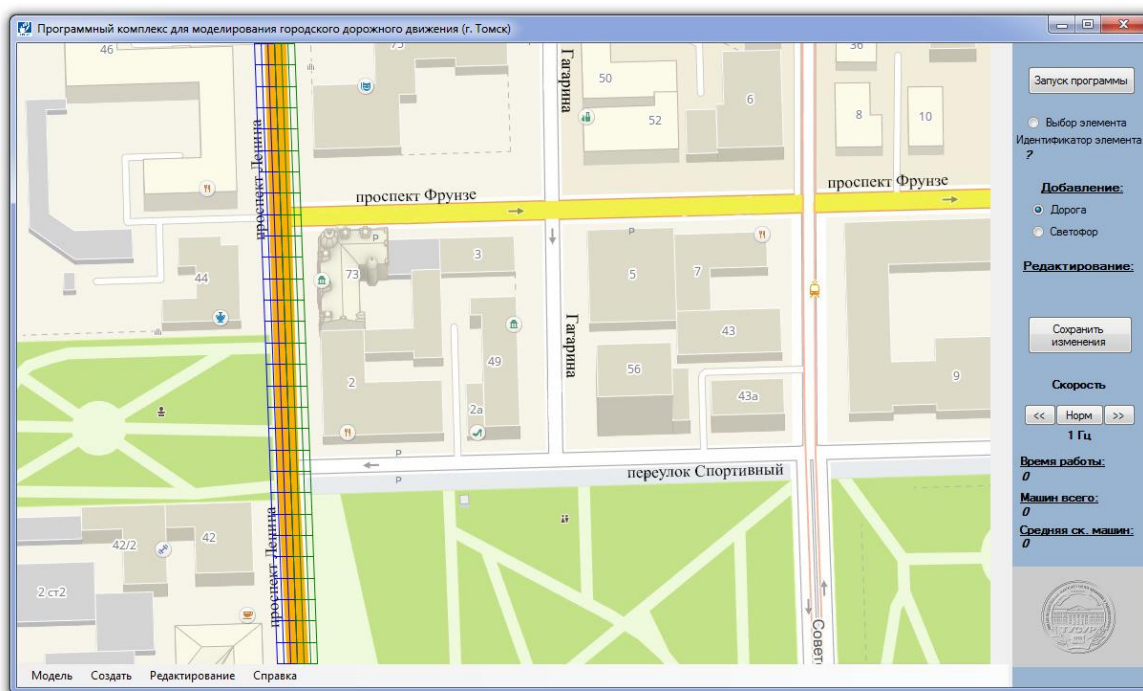


Рисунок 6.11 – Созданный участок трассы.

Запускаем модель кнопкой «Запуск программы». После чего, выполняется расчёт алгоритмов и на карте отображается автотранспорт по указанным ранее параметрам.

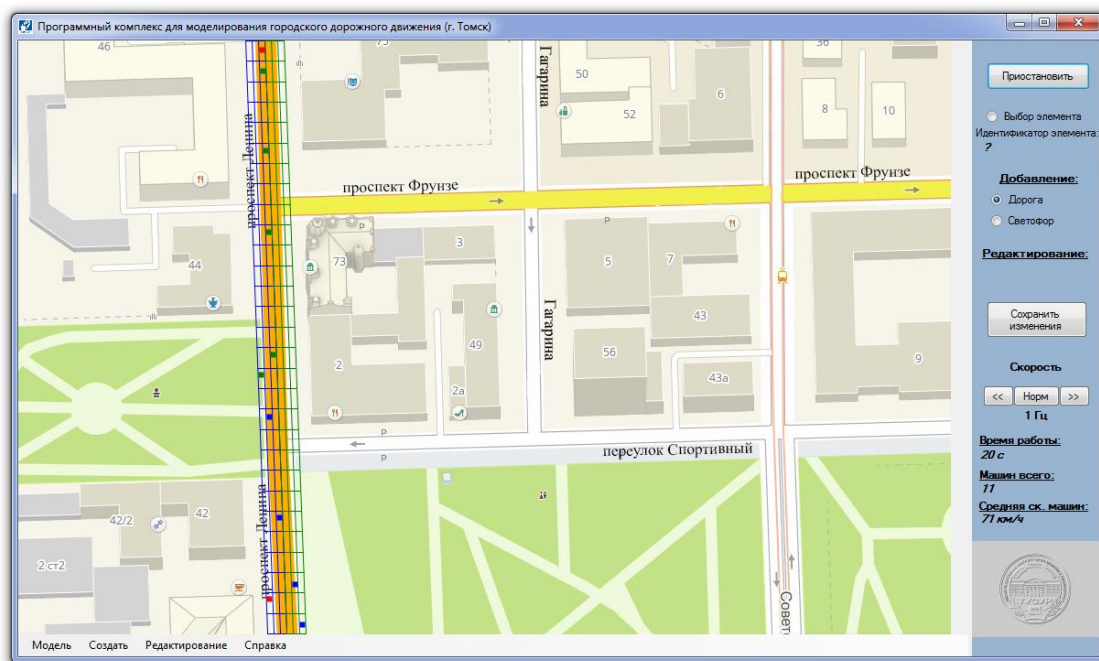


Рисунок 6.12 – Проверка работоспособности модели.

6.1.5 Создание светофора на модели

Для создания светофора на модели, выберите компонент «Светофор» во вкладке создание. После чего, наведите указатель мыши на карту и нажмите левой кнопкой мыши на место, где будет располагаться светофор. Перед вами откроется окно задания параметров светофора (Рисунок 6.13).

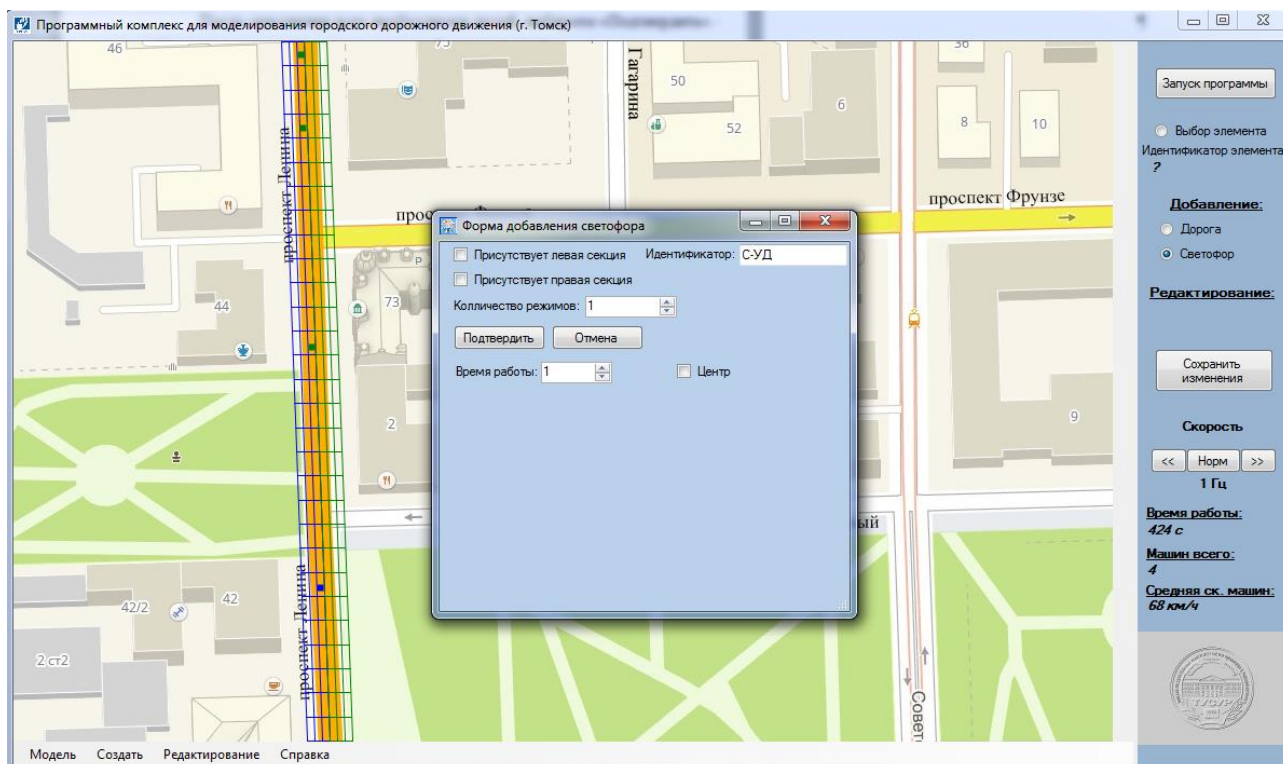


Рисунок 6.13 – Окно создания Светофора.

Далее, задаем уникальный идентификатор для светофора. Следующий шаг, указываем параметры присутствия секций у светофоров. Для этого, ставим галочку напротив имеющейся секции (при дополнительных секций, оставляем значение в первоначальном виде). При указании присутствия секций, открывается форма настройки фаз работы светофора (Рисунок 6.14).

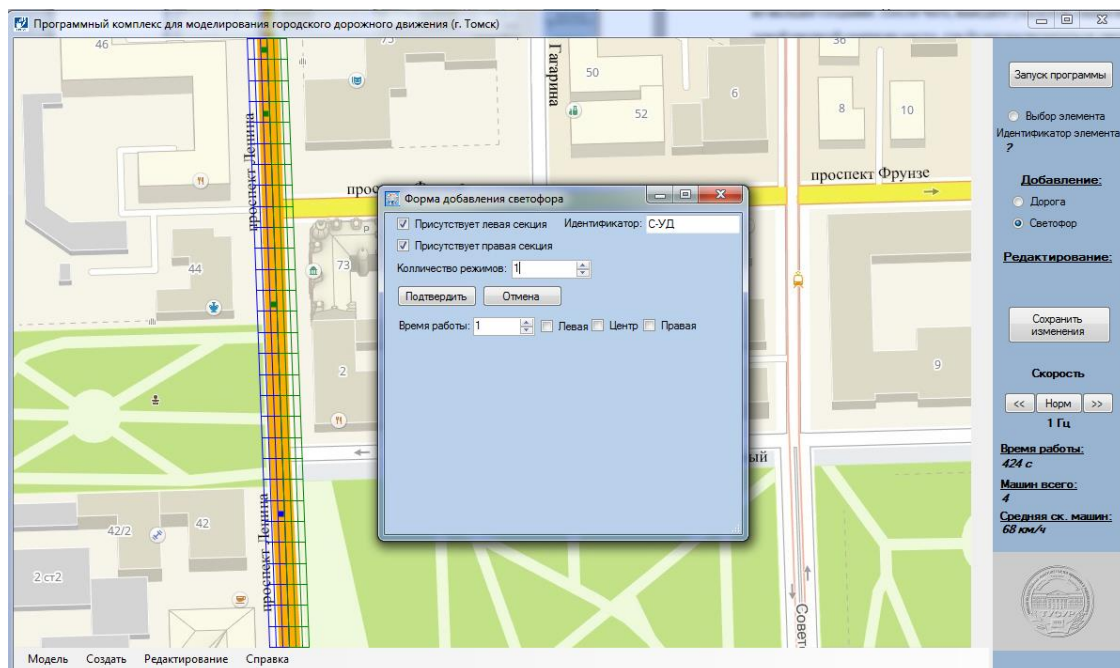


Рисунок 6.14 – Окно настройки светофора.

Далее, указываем количество режимов работы светофоров. Режимы определяют сколько секунд и в какой последовательности должны загораться секции светофора. Центральная секция отвечает за зеленый сигнал, отсутствие выбора параметров говорит о том, что светофор загорается красным (Рисунок 6.15).

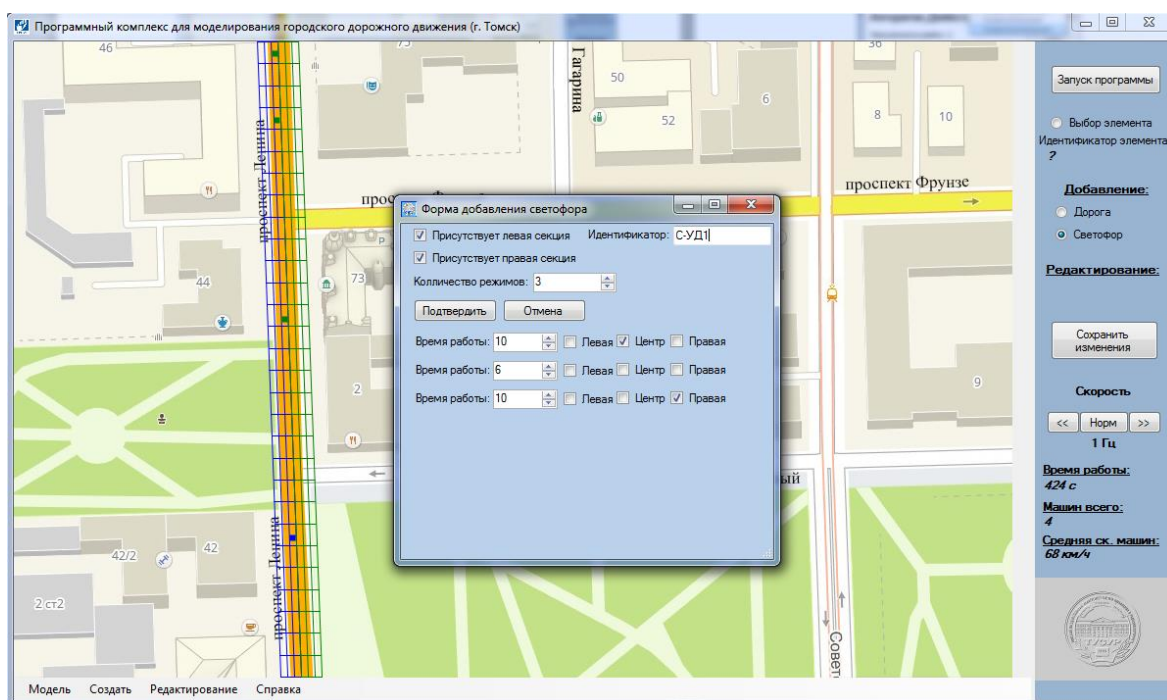


Рисунок 6.14 – Окно настройки светофора.

После заполнения всех необходимых полей, выбираем «Подтвердить». Светофор отобразится на карте (Рисунок 6.15).

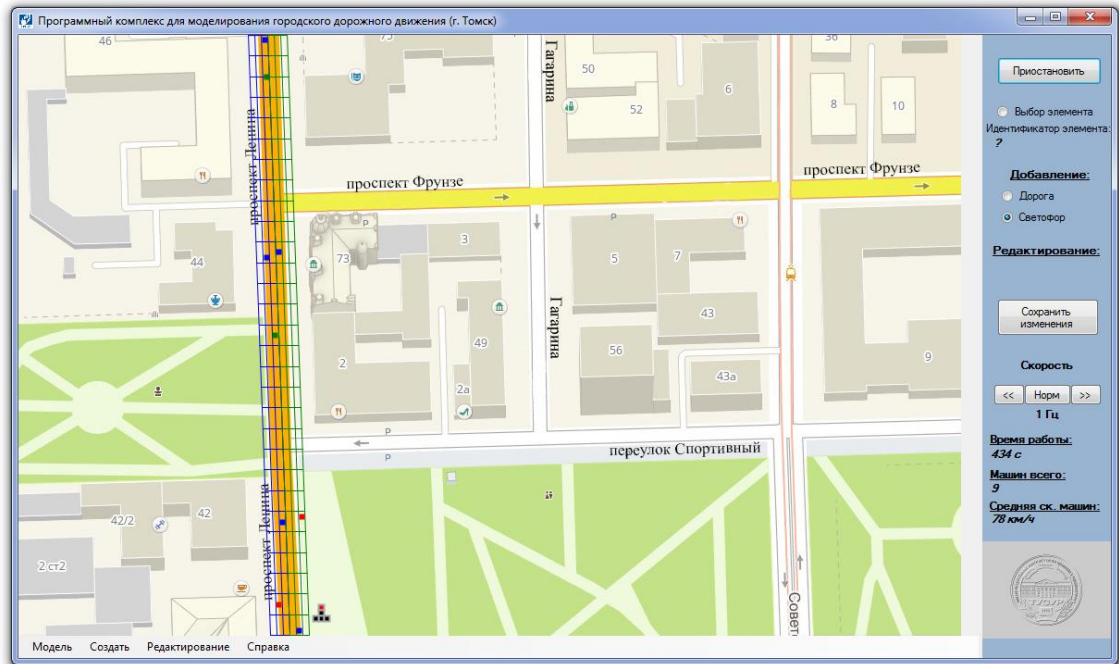


Рисунок 6.15 – Проверка работоспособности модели.

6.1.6 Редактирование участка дороги

Для редактирования участка дороги на модели, выберите компонент «Выбор элемента». Выберите необходимый элемент. После чего, нажмите правой кнопкой мыши по карте и выберите «Редактировать». Перед вами откроется окно задания параметров (Рисунок 6.16).

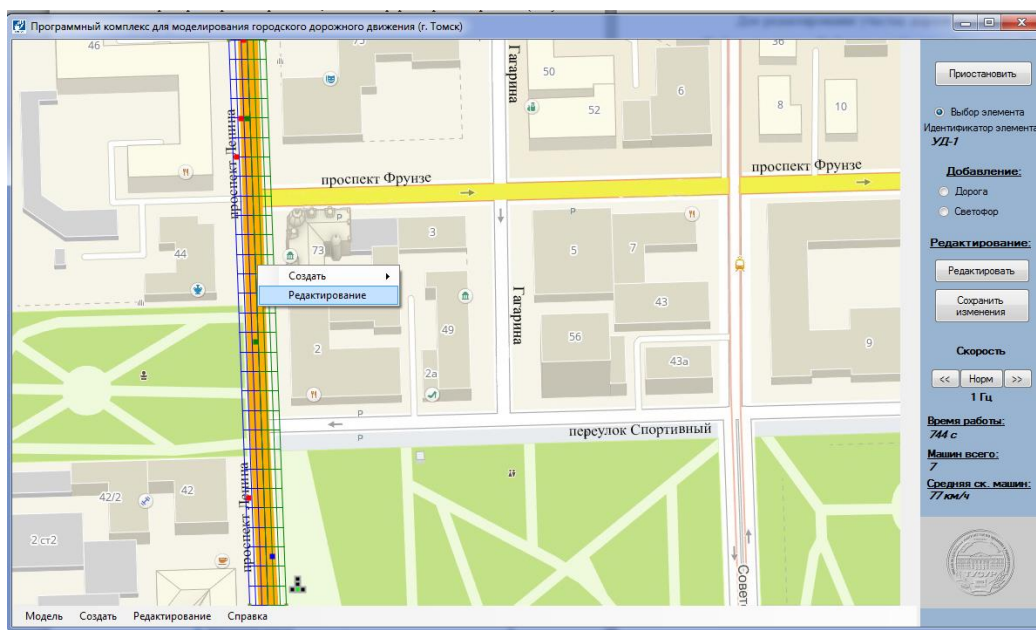


Рисунок 6.16 – Выбор элемента для редактирования.

Далее, откроется окно с параметрами Участка дороги и его идентификатор в правой части программы (Рисунок 6.17).

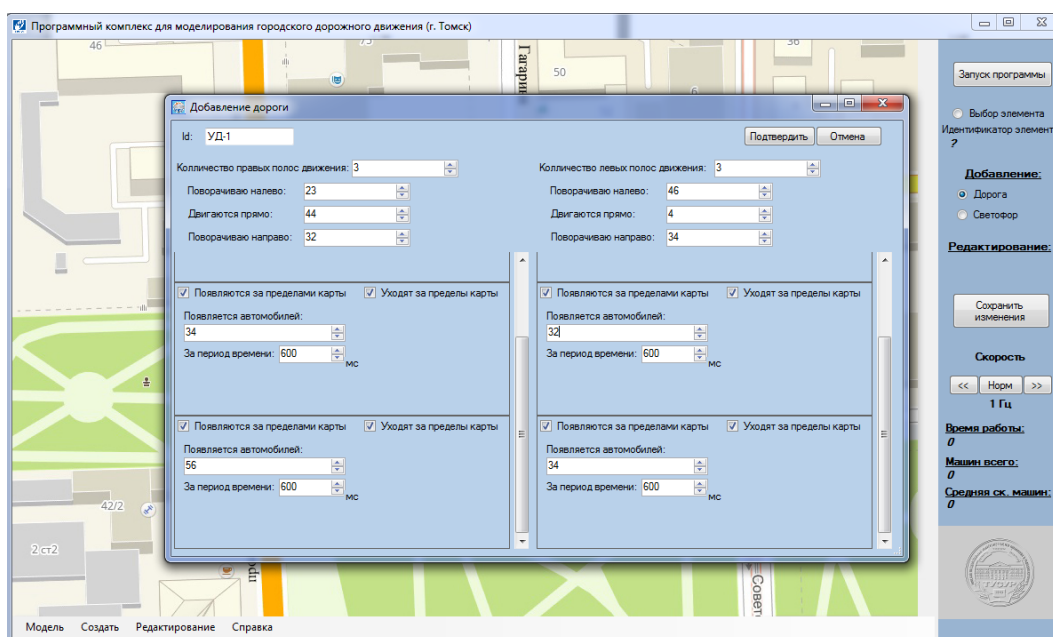


Рисунок 6.17 – Окно участка дороги для редактирования.

6.1.7 Редактирование светофора

Для редактирования светофора на модели, выберите компонент «Выбор элемента». Выберите необходимый элемент светофора. После чего, нажмите правой кнопкой мыши по карте и выберите «Редактировать». Перед вами откроется окно задания параметров (Рисунок 6.18).

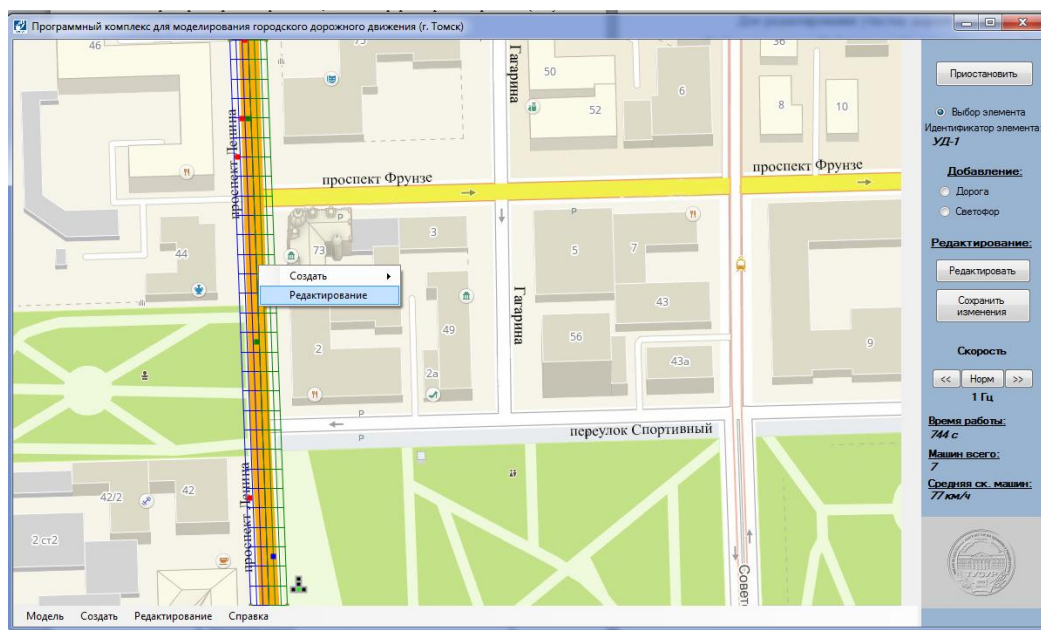


Рисунок 6.18 – Выбор элемента для редактирования.

Далее, откроется окно с параметрами Светофора и его идентификатор в правой части программы (Рисунок 6.19).

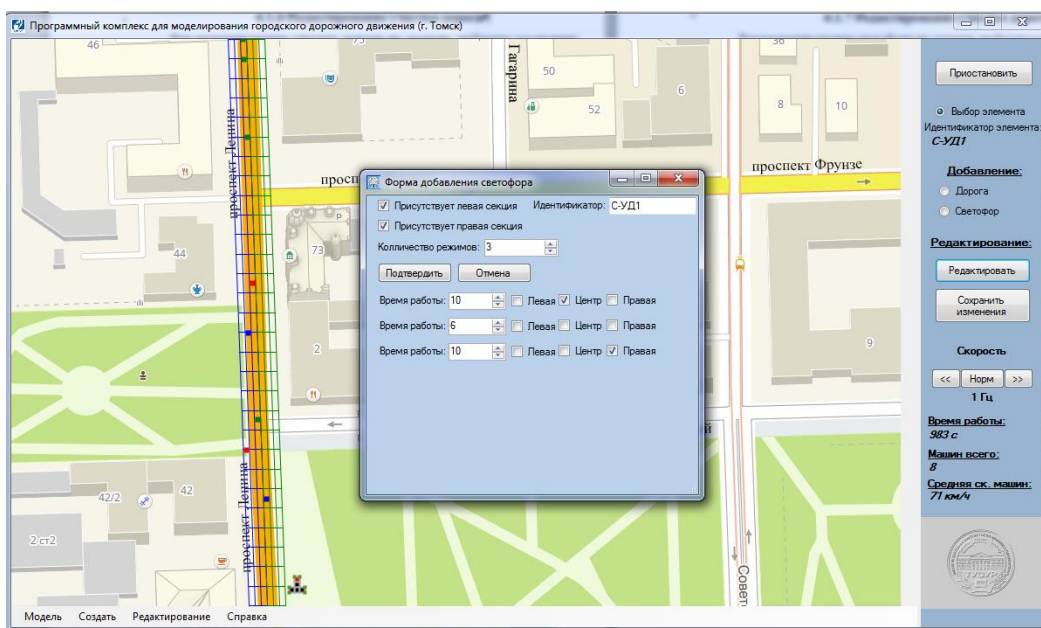


Рисунок 6.19 – Окно светофора для редактирования.

6.1.8 Привязка светофора к полосе движения

Для привязки светофора к участку дороги, выберите компонент «Выбор элемента». Выберите необходимый Участок дороги. После чего, нажмите правой кнопкой мыши по карте и выберите «Редактировать». Перед вами откроется окно задания параметров (Рисунок 6.20).

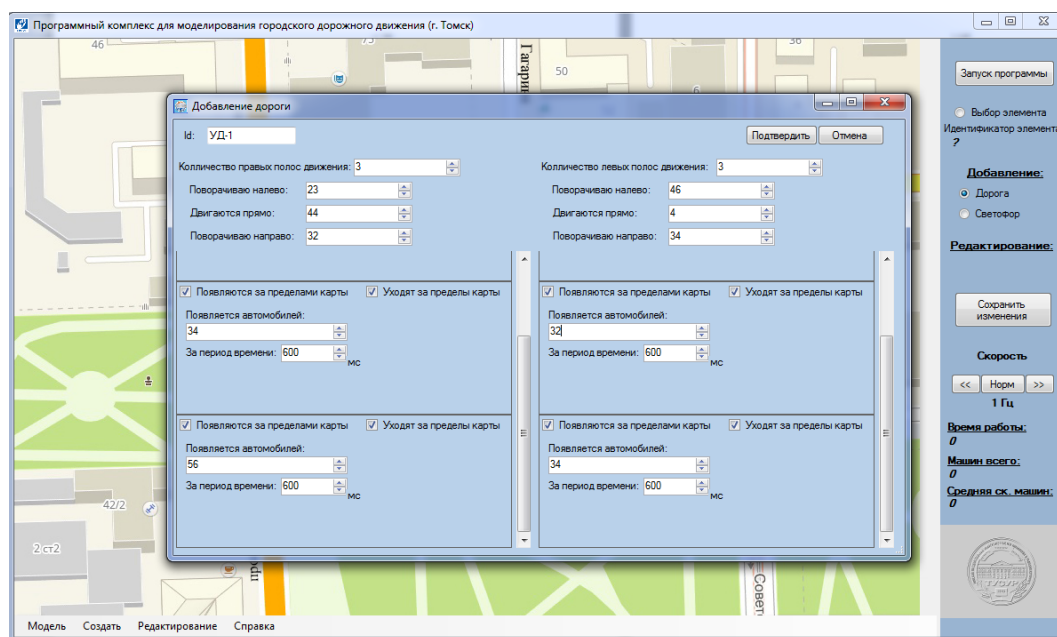


Рисунок 6.20 – Выбор элемента для редактирования.

Далее, нажмите по элементу «Уходят за пределы карты». После чего, перед Вами откроется форма привязки (Рисунок 6.21).

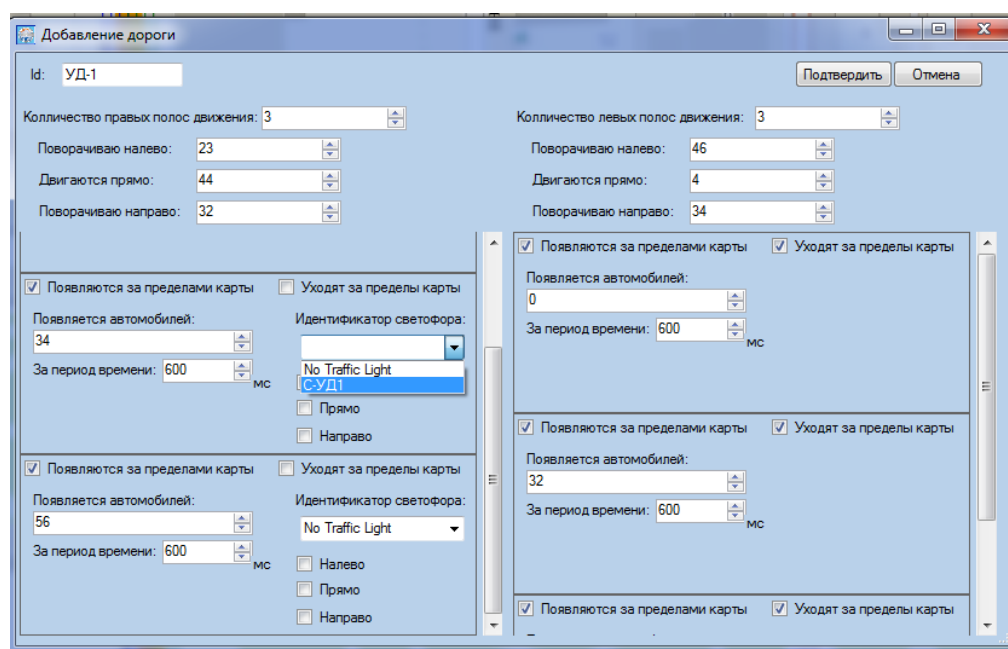


Рисунок 6.21 – Форма привязки участка дороги к светофору.

При выборе существующих светофоров из списка, участок дороги будет реагировать на изменения параметров светофора (красный, зелены, дополнительная секция).

6.1.9 Перестроение автомобилей на другой участок дороги

Для привязки светофора к участку дороги, выберите компонент «Выбор элемента». Выберите необходимый Участок дороги. После чего, нажмите правой кнопкой мыши по карте и выберите «Редактировать». Перед вами откроется окно задания параметров (Рисунок 6.22).

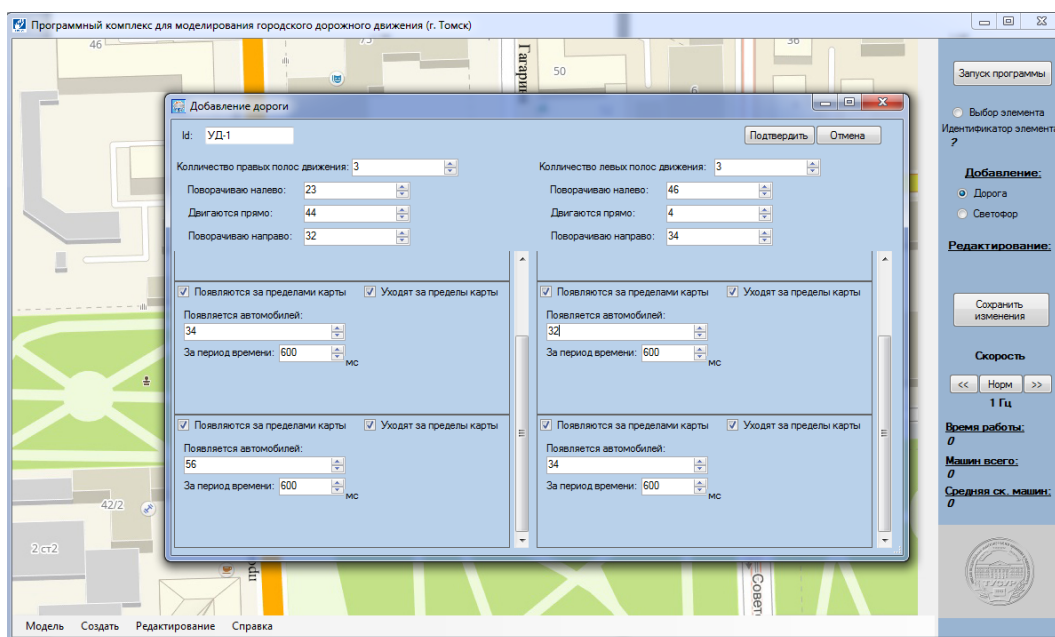


Рисунок 6.22 – Выбор элемента для редактирования.

Далее, нажмите по элементу «Уходят за пределы карты». После чего, перед Вами откроется форма привязки (Рисунок 6.23).

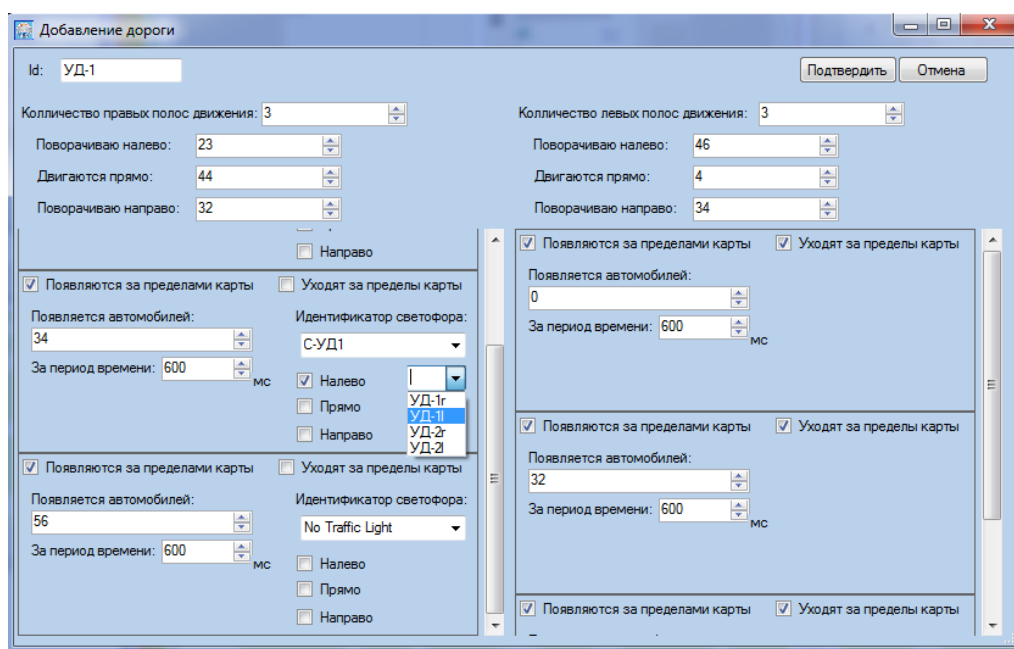


Рисунок 6.23 – Форма привязки участка дороги.

При выборе существующих полос движения из списка, участок дороги будет выполнять перестроение на него в случае, если перед автомобилем стоит цель поворота.

6.1.10 Сохранение пользовательских изменений

Для сохранения результатов работы программы выберите пункт «Сохранить изменения». Изменения будут сохранены в текущий файл `map.map` и отобразятся при следующем запуске программы.

При выходе из программы, в случае, если вы не сохранили изменения, программа автоматически выдаст запрос на сохранение сделанных изменений (Рисунок 6.24).

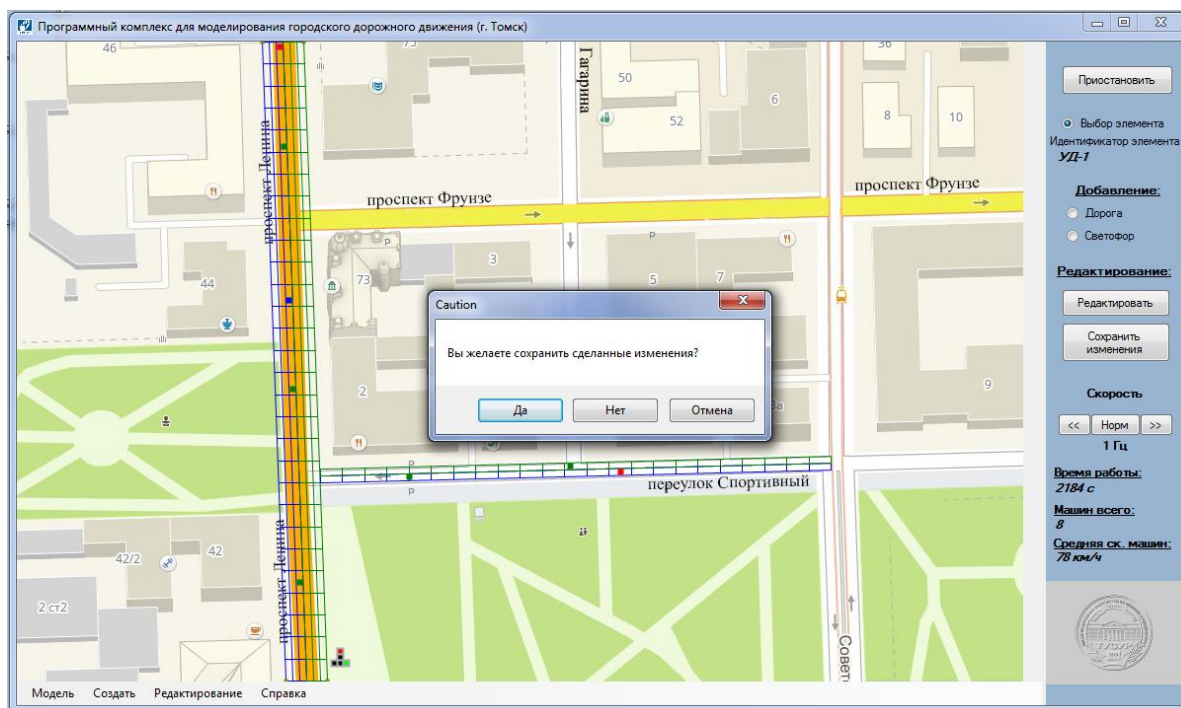


Рисунок 6.24 – Запрос на сохранение.

6.2 Завершение работы с программой

Для завершения работы с программой, закройте все диалоговые окна. Для этого, нажмите на кнопку закрытия приложения (красный квадрат с крестиком).