

Федеральное государственное учреждение  
«Федеральный исследовательский центр  
Институт прикладной математики им. М.В. Келдыша  
Российской академии наук»

*На правах рукописи*

---

*подпись соискателя*

**Чечина Антонина Александровна**

**Математическое моделирование транспортных потоков на основе  
теории клеточных автоматов**

Специальность 05.13.18 - математическое моделирование, численные методы  
и комплексы программ

Диссертация на соискание ученой степени  
кандидата физико-математических наук

Научный руководитель  
кандидат физико-математических наук  
Чурбанова Наталья Геннадьевна

Москва

2021

## ОГЛАВЛЕНИЕ

Введение.....	6
Актуальность .....	6
Современное состояние исследований .....	7
Цель и задачи диссертационной работы.....	8
Научная новизна.....	9
Теоретическая и практическая значимость работы.....	10
Методика исследования .....	10
Достоверность полученных результатов .....	12
Апробация работы, внедрение.....	12
Основные положения, выносимые на защиту .....	16
Объем и структура диссертации.....	17
Глава 1. Обзор основных методов и подходов к транспортному моделированию .....	18
1.1 Различные подходы к моделированию трафика.....	18
1.2 Макроскопические модели. ....	20
1.3 Мезоскопические (кинетические) модели.....	24
1.4 Микроскопические модели. ....	25
1.5 История развития клеточных автоматов. ....	28
1.6 Теория клеточных автоматов применительно к транспортному моделированию. ....	32
1.7 Обзор моделей на основе теории клеточных автоматов.....	33
1.7.1 Правило 184 Вольфрама.....	33
1.7.2 Модель Нагеля-Шрекенберга .....	35
1.7.3 Модель Кернера-Кленова.....	37

1.7.4	Подход Буслаева – моделирование на разветвленных структурах.	38
1.7.5	Другие модели .....	39
1.8	Современные исследования транспортных потоков .....	39
	Выводы по Главе 1 .....	43
Глава 2. Математическая модель транспортных потоков на основе теории клеточных автоматов для моделирования трафика на городской улично-дорожной сети. Основные алгоритмы движения. Различные элементы УДС, примеры тестовых расчетов .....		
2.1	Построение двумерной модели .....	44
2.2	Общий алгоритм движения.....	47
2.2.1	Алгоритм а1- общий алгоритм смены полосы.....	48
2.2.2	Алгоритм а2 - общий алгоритм движения вперед.....	52
2.3	Достижение цели.....	52
2.4	Движение на элементах улично-дорожной сети.....	53
2.4.1	Алгоритм а3 перестроения с достижением цели (движение по участку дороги с выездом) .....	54
2.4.2	Алгоритм а4 прохождения четырехстороннего перекрестка со светофором.....	56
2.4.3	Алгоритм а5 - обеспечение безопасности на перекрестке.....	58
2.4.4	Алгоритм а6 прохождения Т-образного перекрестка со светофором .....	59
2.4.5	Алгоритм а7 прохождения участка с сужением/расширением.....	60
2.4.6	Алгоритм а8 въезда с второстепенной дороги на главную .....	61
2.4.7	Алгоритм а9 проезда участка с разворотом .....	62
2.4.8	Алгоритм а10 проезда участка с пешеходным светофором .....	63
2.4.9	Алгоритм а11 объезда широкого неподвижного препятствия.....	63

2.5	Различные стратегии поведения водителей .....	66
2.5.1	Алгоритм a12 «вежливый водитель» для смены полосы.....	68
2.5.2	Алгоритм a13 «вежливый водитель» для въезда с второстепенной дороги на главную.....	73
2.5.3	Алгоритм a14 снижения скорости при наличии затора .....	77
2.6	Модель транспортных потоков с «медленным стартом» .....	78
2.6.1	Алгоритм a15 «медленный старт».....	78
2.7	Граничные условия в модели.....	79
2.7.1	Алгоритм a16 задания граничных условий с возможностью образования очереди на въезд в расчетную область .....	79
	Выводы по Главе 2.....	81
Глава 3. Численная реализация. Программный комплекс САМ-2D. Пользовательский интерфейс и визуализация. Параллельная реализация. Расчеты на сети.....		
		82
3.1	Структура последовательной программы .....	82
3.2	Основные элементы и структуры.....	82
3.3	Параллельная реализация.....	89
3.3.1	Алгоритмы a17 и a18 выезда и въезда автомобиля на расчетный элемент .....	90
3.4	Интеграция с модулями Визуализации и Пользовательского интерфейса .....	93
3.4.1	Модуль пользовательского интерфейса и визуализации данных ...	93
3.4.2	Интеграция расчетного модуля САМ-2D с модулем пользовательского интерфейса и визуализации .....	97
3.5	Расчеты на сети .....	101
	Выводы по Главе 3.....	102

Глава 4. Верификация модели: Сравнение с Aimsun TSS, воспроизведение экспериментально наблюдаемых пространственно-временных структур....	103
4.1 Сравнение с Aimsun TSS .....	103
4.1.1 Тестовая задача 1 о пропускной способности перекрестка.....	103
4.2 Сравнение с экспериментальными данными .....	105
4.2.1 Различные способы осреднения данных .....	107
4.2.2 Тестовая задача 2 о въезде на автомагистраль с постоянными входящими потоками. Сравнение методов осреднения.....	109
4.2.3 Тестовая задача 3 о въезде на магистраль с переменным потоком $Q_1$ на главной дороге. Сравнение результатов с данными, полученными при помощи модели Кернера-Конхойзера.....	111
4.2.4 Тестовая задача 4 о движении на участке магистрали с выездом и въездом .....	113
Выводы по Главе 4.....	115
Заключение .....	117
Список литературы .....	119
Приложение 1 Лицензионный договор и акт приема-передачи по проекту «Разработка архитектуры, принципов построения и программного обеспечения суперкомпьютерной информационно-аналитической платформы прикладного моделирования, прогнозирования и экспертиз транспортных процессов и систем в виде сетевой компьютерной лаборатории» .....	129
Приложение 2 Свидетельство о государственной регистрации программы САМ-2D.....	136

## **ВВЕДЕНИЕ**

### **Актуальность**

Высокая загруженность автомобильных дорог транспортом – проблема, с которой сталкивается каждый житель большого города и его пригородов. Маятниковая миграция населения, вызванная сосредоточением рабочих мест в крупных городах, вместе с высокой стоимостью жилья, переносит проблемы заторов далеко за пределы мегаполиса. Растущее число автомобилей в расчете на душу населения только усугубляет эту проблему. Для улучшения транспортной ситуации требуется инструмент, позволяющий как управлять существующей транспортной инфраструктурой, так и планировать строительство новой, а также прогнозировать эффективность ограничительных мер, которые обычно бывают непопулярными (например, платная парковка на улицах города, ограничение въезда и др.)

Во многих мегаполисах мира успешно работает так называемая интеллектуальная транспортная система (ИТС), позволяющая в реальном времени отслеживать и вносить изменения в управление дорожным движением, выбирая самые эффективные решения.

В связи с появлением более мощных вычислительных ресурсов появилась возможность строить более сложные и приближенные к жизни математические модели транспортных потоков, учитывающие взаимодействие водителей с дорожными знаками, разметкой, светофорами и так далее, а также между собой. Моделирование на высокопроизводительных системах может позволить охватывать дорожные сети в масштабах целого города.

Появление в последние годы множества новых данных о потоках автотранспорта, собираемых датчиками и камерами видеонаблюдения, во-первых, открывает новые возможности для изучения закономерностей

динамики потоков, а во-вторых, позволяет проводить точную калибровку моделей для использования в конкретном месте, что может улучшить качество прогнозов, полученных в рамках вычислительных экспериментов.

Это особенно важно для управления потоками на «микроуровне» – то есть в масштабе нескольких соседних перекрестков или кварталов – например, вблизи мест скопления людей, таких как торговые центры, концертные залы и пр.

Все вышеперечисленное позволяет сделать вывод об актуальности и перспективности выбранного автором диссертации направления исследований.

### **Современное состояние исследований**

Несмотря на более чем 70-летнюю историю развития моделирования движения автотранспорта, данная тема по-прежнему остается малоизученной. Первые модели, созданные в рамках макроскопического подхода, не обеспечивают необходимую точность и не воспроизводят многие свойства реальных транспортных потоков, которые наблюдаются экспериментально. Связано это, в первую очередь, с вынужденной экономией на вычислениях. По мере развития этой области моделирования, развивались и усложнялись применяемые методы и подходы одновременно с ростом производительности компьютеров. С этим же связан и переход от макроскопических моделей, описывающих транспортный поток как сплошную среду, к микроскопическим моделям, которые отображают более тонкие взаимодействия, описывая движение каждого отдельного автомобиля, и дальнейшее усложнение этих моделей. Этот процесс идет до сих пор.

В целом, из-за большого разнообразия методов и подходов, применяемых в транспортном моделировании, а также большого разнообразия задач, которые необходимо решить для реального моделирования трафика на

городской дорожной сети, помимо непосредственно описания движения автомобилей, область по-прежнему остается недостаточно исследованной.

Выбранный для диссертационной работы подход на основе теории клеточных автоматов берет свое начало со статьи Нагеля и Шрекенберга [1] 1992 года. За прошедшие годы выделилось два основных направления развития данного подхода: моделирование на городских сетях при помощи сложных структур на основе ячеек клеточных автоматов [2, 3] и исследования в рамках теории трех фаз Б. Кернера [4], дополняющие правила перемещения вдоль дороги из модели Нагеля-Шрекенберга [5-10].

Подход на основе теории клеточных автоматов очень удобен для описания различных поведенческих стратегий водителей, включающих ситуации дискретного выбора, поскольку основан не на уравнениях в частных производных, а на логических алгоритмах и условиях «если - то». Моделирование водительских стратегий и принятия решений во время движения в потоке – область в настоящее время мало исследованная и потому представляющая интерес.

Также до сих пор при решении задач транспортного моделирования редко встречается выполнение расчетов на высокопроизводительных системах, что, безусловно, является важным для масштабных расчетов на городских сетях.

### **Цель и задачи диссертационной работы**

Целью работы является создание новой математической модели транспортных потоков, учитывающей реальные особенности поведения водителей и их взаимодействия с дорожно-транспортной инфраструктурой, и комплекса программ, адаптированного для высокопроизводительных вычислительных систем, реализующего эту модель.

Для этого автором диссертации решены следующие задачи:

Построена двумерная (многополосная) модель, включающая алгоритмы для движения автотранспорта на различных элементах улично-дорожной сети: регулируемых и нерегулируемых перекрестках, участках дороги с расширением и сужением, прямых участках дороги и т.д. В модели учтены разные стили вождения и стратегии смены полос участников движения, рассмотрен объезд препятствий (например, автомобилей, попавших в ДТП).

Разработан модульный программный комплекс, позволяющий делать расчеты на указанных элементах УДС по отдельности (в последовательном режиме), а также собирать из данных элементов сети различной конфигурации и делать расчеты на них (в параллельном режиме).

Проведено тестирование алгоритмов и верификация модели и комплекса на тестовых задачах. Проведено сравнение полученных результатов с экспериментальными данными, взятыми из литературы, а также с расчетами, выполненными при помощи коммерческого пакета для моделирования движения автотранспорта Aimsun TSS [11].

### **Научная новизна**

Работа сочетает в себе два свойства, редко встречающихся в работах других исследователей (даже по отдельности): во-первых, учет при моделировании стратегий вождения, а во-вторых – адаптацию модели для расчетов на суперкомпьютерах. В этом основное отличие представленных модели, алгоритмов и программного комплекса от других.

Созданная модель на основе теории клеточных автоматов является новой и оригинальной. Входящие в модель алгоритмы проезда различных элементов УДС, а также алгоритмы взаимодействия водителей с дорожной инфраструктурой и с другими водителями, алгоритмы, реализующие различные стратегии поведения на дороге, и программный комплекс для

высокопроизводительных вычислительных систем, реализующий модель, также являются новыми и оригинальными.

### **Теоретическая и практическая значимость работы**

В теоретической части, созданные новые алгоритмы, учитывающие разные стратегии поведения в транспортном потоке, а также реализация этих алгоритмов в виде комплекса программ для высокопроизводительных систем, представляет интерес как основа для создания новых подходов к моделированию автотранспорта.

В практической части, созданная модель и программный комплекс могут служить базой для интеллектуальных транспортных систем городов, позволяя управлять транспортными потоками в режиме реального времени, а также решать прогнозные задачи при планировании новой и модификации существующей дорожно-транспортной инфраструктуры.

### **Методика исследования**

Подход, основанный на теории клеточных автоматов, максимально хорошо подходит для описания сложных разнородных систем, какими являются транспортные потоки. Одно из преимуществ подхода – его дискретная логическая основа, которая позволяет легко описывать принятие решений водителями при движении и взаимодействии с другими участниками дорожного движения.

Созданная модель основана на одномерной модели Нагеля-Шрекенберга. Модель обобщена на многополосный случай путем добавления набора правил для смены полосы в различных условиях. Вместо одномерной расчетной области имеем двумерную; автомобили могут перестраиваться и обгонять друг друга. Правила обновления состояния ячеек из модели Нагеля-Шрекенберга, состоящие из четырех шагов, описывающих движение вдоль дороги, дополнены разветвленными логическими алгоритмами для

перестроений. Для моделирования движения в городе в модель добавлены элементы управляющего воздействия, такие, как дорожные знаки, сигналы светофора и разметка.

Поведение водителей на дороге может существенно отличаться в зависимости от их личностных качеств, стажа вождения и других параметров. Учет этих стратегий облегчает калибровку модели при практическом моделировании на участках дорожных сетей реальных городов (например, в больших городах стиль вождения, как правило, носит более агрессивный характер, а в городах с неинтенсивным движением – более спокойный). Также, как известно, частая смена полосы отдельными водителями может существенно влиять на весь транспортный поток, порождая волны уплотнений при стесненном движении, даже при отсутствии физических препятствий на дороге.

Для учета разной манеры вождения в модель включены «агрессивные» и «осторожные» водители, а для реалистичного моделирования проезда участка с сужением, при котором обычно водители пропускают автомобили из ряда с препятствием по одному (“zipper merge” или «елочка») – алгоритм «вежливый водитель».

Для верификации созданной модели и алгоритмов решен ряд тестовых задач, выполнено сравнение с экспериментальными данными, данными, полученными с использованием других моделей (взятыми из литературы) и результатами моделирования при помощи коммерческого пакета для транспортного моделирования. Выполнены расчеты на небольших дорожных сетях в параллельном режиме на суперкомпьютере К-100 (ИПМ им. Келдыша РАН) [12].

## **Достоверность полученных результатов**

Тестовые расчеты согласуются с экспериментальными данными. Сравнение с расчетами, полученными другими исследователями и с помощью коммерческого пакета Aimsun TSS, демонстрируют сходство основных закономерностей, однако, обеспечивают более детализированные данные и большее сходство с экспериментом.

## **Апробация работы, внедрение**

Работа была апробирована на ряде научных конференций и семинаров:

1. “Different Approaches to the Multilane Traffic Flows Simulation”. 17th International Conference Mathematical Modelling and Analysis, June 6 - 9, 2012, Tallinn, Estonia.
2. «Макро- и микроскопические модели транспортных потоков на многополосных магистралях». Всероссийская конференция «Теоретические основы и конструирование численных алгоритмов решения задач математической физики», посвященной памяти К.И. Бабенко, 10-16 сентября 2012 г., Россия, п. Абрау-Дюрсо.
3. “Different Approaches to the Multilane Traffic Simulation”. 15th International Conference "Traffic and Granular Flow -13" (TGF-13), 25-27 September, 2013, Juelich Germany.
4. “Mathematical simulation of multilane traffic flows using macro- and microscopic approaches” 6th Int. Conf. on Computational Methods in Applied Mathematics (CMAM-6), Sept. 28 - Oct. 4, 2014, St. Wolfgang, Austria.
5. “Two approaches to the vehicular traffic flows simulation using high-performance computer systems”. International Interdisciplinary Seminar “Mathematical Models and Modeling in Laser-Plasma Processes and Advanced Science Technologies”, May 30 – June 6, 2015, Petrovac, Montenegro.
6. “Simulation of multilane vehicular traffic on the basis of cellular automata theory”, International Interdisciplinary Seminar “Mathematical Models and

- Modeling in Laser-Plasma Processes and Advanced Science Technologies”  
July 4-9, 2016, Moscow, Russia.
7. “Multilane Traffic Flow Modeling Based on Cellular Automata Theory Using High-performance Computer Systems”. 5th European Seminar on Computing (ESCO 2016), June 5-10, 2016, Pilsen, Czech Republic.
  8. “Multilane traffic flow modeling using cellular automata theory”. International Conference on Mathematical Modeling and Computational Physics (MMCP’2017), July 3-7, 2017, Dubna, Russia.
  9. “Traffic flow modelling on road networks using cellular automata theory”, International Conference on Communication, Management and Information Technology (ICCMIT2018), April 02-04, 2018, Madrid, Spain.
  10. “Cellular automata in application to traffic flow simulation”, Актуальные проблемы прикладной математики и информационных технологий – Аль-Хорезми 2018, 13–15 сентября 2018 г., Ташкент, Узбекистан.
  11. «Моделирование и визуализация транспортных потоков на элементах дорожной сети с использованием теории клеточных автоматов», XII мультиконференция по проблемам управления (МКПУ-2019), 23-28 сентября 2019, Дивноморское, Геленджик, Россия.
  12. “Simulation and visualization of vehicular traffic on road networks using high performance computing systems”. 19th International Conference of Computational and Mathematical Methods in Science and Engineering, June 30-July 6, 2019, Rota, Cadiz, Spain.
  13. “Reproduction of experimental spatio-temporal structures in traffic flows using mathematical model based on cellular automata theory”, International Conference on Communication, Management and Information Technology (ICCMIT2019), March 26-28, 2019, Vienna, Austria.
  14. “Supercomputer Technology for Traffic Simulation in a Metropolis”. 2020 International Conference on Engineering Management of Communication and Technology (EMCTECH), Oct. 20-22, 2020, Vienna, Austria.

15. «Модель транспортных потоков на основе теории клеточных автоматов для решения задач управления движением на городских дорожных сетях»  
13 Конференция по проблемам управления МКПУ-2020, Конференция «Математическая теория управления и ее приложения» (МТУиП-2020). 6-8 октября 2020г., Санкт-Петербург, Россия.
16. «Алгоритмы поведения водителей в математической модели транспортных потоков на основе теории клеточных автоматов» V Международная конференция Моделирование нелинейных процессов и систем (MNPS-2020), 16-20 ноября 2020 г., Москва, Россия.
17. «Алгоритмы поведения водителей в математической модели транспортных потоков на основе теории клеточных автоматов», 79-я международная научно-методическая и научно-исследовательская конференция МАДИ, 25-30 января 2021г., Москва, Россия.

Гранты РФФИ, включающие результаты работы:

**13-01-00781-а (2013-2015)**

Математическое моделирование потоков автотранспорта с использованием высокопроизводительных вычислительных систем на основе методов математической физики и нейросетевой технологии, руководитель – Чурбанова Н. Г.

**13-01-12008-офи\_м (2013-2015)**

Различные подходы к математическому моделированию многополосных транспортных потоков на высокопроизводительных многопроцессорных вычислительных системах, руководитель – Чурбанова Н. Г.

**13-01-12046-офи\_м (2013-2015)**

Разработка фундаментальных основ, методов и средств прикладного моделирования транспортных процессов и систем на вычислительных комплексах гибридной архитектуры, руководитель – Колесниченко А. В.

**14-01-90009-Бел\_а (2014-2015)**

Разработка и исследование сеточных алгоритмов для эффективных расчётов некоторых эволюционных задач математической физики, гидродинамики и динамики транспортных потоков, руководитель – Четверушкин Б. Н.

**15-01-03445-а (2015-2017)**

Разработка эффективных вычислительных технологий для решения крупномасштабных задач математической физики на многопроцессорных системах сверхвысокой производительности, руководитель – Кулешов А. А.

**16-01-00347-а (2016-2018)**

Моделирование потоков автотранспорта на больших транспортных сетях с использованием микроскопического и мультиагентного подходов, руководитель – Чурбанова Н. Г.

**16-31-00087-мол-а (2016-2017)**

Математическое моделирование многополосных транспортных потоков на основе теории клеточных автоматов с использованием высокопроизводительных вычислительных систем, руководитель – Чечина А. А.

**18-01-00405-а (2018-2020)**

Развитие современных вычислительных технологий для моделирования крупномасштабных физических процессов с использованием высокопроизводительных вычислительных систем, руководитель - Кулешов А. А.

**18-51-41001-Узб\_г (2018-2019)**

Высокоточные вычислительные алгоритмы для решения задач газовой динамики, фильтрации и динамики транспортных потоков с использованием современных суперкомпьютеров, руководитель - Якобовский М. В.

**20-07-00528-а (2020-2022)**

Разработка вычислительных технологий для исследования и визуальной интерпретации динамики транспортных потоков на улично-дорожной сети мегаполиса на базе систем сверхвысокой производительности, руководитель - Чурбанова Н. Г.

Созданный программный комплекс САМ-2D был внедрен при создании платформы поддержки принятия решений для проекта «Разработка архитектуры, принципов построения и программного обеспечения суперкомпьютерной информационно-аналитической платформы прикладного моделирования, прогнозирования и экспертиз транспортных процессов и систем в виде сетевой компьютерной лаборатории» в рамках ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2014-2020 годы», Соглашение о предоставлении субсидии от 30.06.2014 г. № 14.604.21.0052. (см. Приложение 1).

На программный комплекс САМ-2D (упрощенная версия) было получено свидетельство на программу для электронных вычислительных машин № 2016662572 от 15.11.2016 (см. Приложение 2).

**Основные положения, выносимые на защиту**

1. Двумерная многополосная микроскопическая модель транспортных потоков на основе теории клеточных автоматов

2. Алгоритмы движения на различных элементах улично-дорожной сети для созданной модели
3. Различные поведенческие алгоритмы водителей для созданной модели
4. Комплекс программ САМ-2D для моделирования движения автотранспорта на городской улично-дорожной сети, адаптированный для высокопроизводительных вычислительных систем, интегрированный с модулем пользовательского интерфейса и визуализации результатов
5. Результаты вычислительных экспериментов, полученные с помощью комплекса программ.

### **Объем и структура диссертации**

Объем диссертационной работы – 136 страниц машинописного текста. Диссертация состоит из введения, четырех глав, заключения, библиографического списка из 90 наименований и двух приложений. Работа содержит 50 рисунков. Глава 1 содержит обзор основных существующих методов и подходов к транспортному моделированию, главы 2-4 содержат собственные исследования автора диссертации.

# ГЛАВА 1. ОБЗОР ОСНОВНЫХ МЕТОДОВ И ПОДХОДОВ К ТРАНСПОРТНОМУ МОДЕЛИРОВАНИЮ

## 1.1 Различные подходы к моделированию трафика.

История моделирования транспортных потоков насчитывает уже около 70 лет. За это время сформировалось несколько подходов к построению моделей. Первый подход – *макроскопический*, основанный на аналогии движения потока автомобилей и потока сжимаемой жидкости или газа. В основе моделей – адаптированные уравнения гидрогазодинамики, чаще всего одномерные. В уравнения добавляют дополнительные члены, которые отвечают за «человеческий фактор». При этом работа идет с усредненными величинами, характеризующими поток, такими как плотность и скорость потока. Отдельные автомобили в данном подходе не рассматриваются. Также к данному классу относят статические модели, в которых вообще нет речи о движении, рассматривается лишь мгновенное распределение потоков по сети в зависимости от матрицы корреспонденций (матрицы, содержащей информацию о том, сколько автомобилей перемещается в сети от узла дорожной сети  $i$  к узлу дорожной сети  $j$  в данное время суток) и пропускной способности дорог в сети.

Главный плюс макромоделей – в экономии вычислительных ресурсов, поскольку время расчета не растет с ростом числа автомобилей в системе (то есть, плотности). Однако, приближение сплошной среды, к которому относятся макромодели, накладывает свои ограничения: для того, чтобы считать поток сплошной средой, расстояние между автомобилями должно быть порядка размеров самих автомобилей. Это соответствует синхронизированному, то есть достаточно плотному, потоку, где автомобили должны подстраивать свою скорость под скорость потока и скорость движения меньше свободных скоростей для данной дороги. Другое ограничение, вытекающее из приближения сплошной среды – это то, что

возможно рассмотрение только достаточно протяженных участков дорог, ведь в масштабах десятков и сотен метров физическими размерами автомобилей нельзя пренебречь.

Помимо перечисленных ограничений, есть еще ограничения, связанные со сложностью описания реакции водителей на светофоры и разметку, что важно при моделировании небольших участков с целью решений локальных задач управления дорожным движением.

Наконец, стратегии водителей в макромоделях должны быть одинаковыми.

Строго говоря, с помощью макромоделей можно описывать многофазные (то есть состоящие из автомобилей, отличающиеся по цели или по техническим характеристикам) транспортные потоки, однако это описание затруднительно и требует отдельного решения систем уравнений для каждого отдельного типа автомобилей.

Второй возможный подход - *мезоскопический*, он основан на уравнениях, выводящихся из кинетических соображений, аналогично уравнениям физической кинетики. Основная переменная в этом подходе – фазовая плотность потока, представляющая собой распределение автомобилей по координате и скорости. Основные недостатки модели – те же, что и для макроскопических моделей.

И, наконец, третий существующий подход – *микроскопический*. В нем мы имеем дело с каждым автомобилем в отдельности. Существенная часть микроскопических моделей относятся к моделям следования за лидером, в центре внимания в них взаимодействие ведомого и головного автомобиля. Как правило, модели этого класса используют уравнения движения автомобилей, но есть и исключение – модели на основе теории клеточных автоматов.

Работа с каждым автомобилем отдельно позволяет сделать модель более гибкой для описания процессов на «микроуровне» - то есть в масштабах отдельных перекрестков. Микроскопический подход не имеет ограничений на плотность потока, можно описывать как очень разреженное, так и плотное движение. Запись уравнений движения в явном виде позволяет точнее описывать паттерны торможения и ускорения автомобилей в потоке в зависимости от скоростей и расстояний между ними, что важно для воспроизведения экспериментальных свойств трафика. С развитием компьютерных технологий проблема нехватки вычислительных ресурсов для расчетов с большим числом автомобилей на масштабных сетях уже не представляет проблемы.

## 1.2 Макроскопические модели.

Одна из первых математических моделей транспортного потока – модель Лайтхилла-Уизема-Ричардса (LWR) – была создана в 1955 году [13].

В ее основе два соотношения: уравнение неразрывности (1) и связь скорости  $V(x,t)$  и плотности  $\rho(x,t)$  (2):

$$\frac{\partial \rho}{\partial t} + \left( V_e + \rho \frac{dV_e}{d\rho} \right) \frac{\partial \rho}{\partial x} = 0 \quad (1)$$

$$V(x,t) = V_e(\rho(x,t)) \quad (2)$$

Здесь  $x$  – координата,  $t$  – время. Соотношение предполагает, что транспортный поток  $Q(x,t) = \rho(x,t)V(x,t)$  или скорость  $V(x,t)$  всегда находятся в локальном равновесии относительно действительной плотности: транспортный поток и локальная скорость мгновенно следуют плотности, не только при стационарном трафике, но и во всех ситуациях. Точная форма зависимости скорость-плотность для  $V_e(\rho)$  определяется путем калибровки по эмпирическим данным скорость-плотность или поток-плотность.

Так как уравнения не определяют функциональную форму зависимости  $Q_e(\rho)$  (фундаментальной диаграммы), LWR определяет целый класс моделей. Все модели этого класса включают только одно динамическое уравнение – уравнение неразрывности. Поэтому они называются также моделями первого порядка.

Такие модели могут описывать обрушения трафика в узких местах благодаря недостаточной пропускной способности и распространение получающихся плотных областей. С микроскопической точки зрения, соответствующая мгновенная адаптация скорости реализует неограниченные ускорения, которые, конечно, нереалистичны. Более того, конечные времена адаптации скорости и времена реакции являются главными факторами, ведущими к росту волн трафика и скачку пропускной способности или нестабильностям транспортного потока в общем случае. Соответственно, модели LWR не могут описывать эти явления.

Модели второго порядка предполагают, что локальная скорость является независимой динамической величиной, которая, соответственно, моделируется дополнительным динамическим уравнением – уравнением для ускорения.

В общей форме уравнение можно записать так:

$$\frac{dV(x,t)}{dt} = \left( \frac{\partial}{\partial t} + V(x,t) \frac{\partial}{\partial x} \right) V(x,t) = A[\rho(x,t), V(x,t)] \quad (3)$$

В качестве примера модели второго порядка приведем модель Кернера-Конхойзера [14, 15]. Первое уравнение модели – уравнение неразрывности (1), а уравнение для ускорения выглядит так:

$$\frac{\partial V}{\partial t} + V \frac{\partial V}{\partial x} = \frac{V_e(\rho) - V}{\tau} - \frac{c_0^2}{\rho} \frac{\partial \rho}{\partial x} + \frac{\eta}{\rho} \frac{\partial^2 V}{\partial x^2} \quad (4)$$

Эта модель чисто феноменологическая, то есть она не основана на микроскопической модели. Вместо этого, ее уравнения сформулированы по аналогии с уравнениями одномерного сжимаемого газа со звуковыми скоростями  $\pm c_0$  и переменным коэффициентом диффузии скорости  $D_v = \eta / \rho \cdot \tau$  здесь - время релаксации скорости.

Член диффузии введен для того, чтобы предупредить нереалистичные острые переходы, практически ударные волны. Однако диффузия также снижает численную эффективность моделирования, так как она благоприятствует численной неустойчивости. Если такая неустойчивость существует, динамические величины сильно осциллируют и неограниченно растут. Следовательно, этого надо избегать любой ценой, в отличие от физической неустойчивости, которая желательна в определенном диапазоне плотности плотного трафика – в конце концов, физические неустойчивости являются причиной наблюдаемых волн трафика.

В качестве еще одного примера макромодели приведем квазигазодинамическую (КГД) модель, разработанную в ИПМ им. М. В. Келдыша РАН под руководством академика Б. Н. Четверушкина [16]. Особенность модели – ее двумерность, что выделяет ее среди большинства макромоделей. Система уравнений содержит составляющие скорости поперек дороги.

При построении модели используются уравнения, аналогичные уравнениям КГД системы, переменными в которых являются плотность  $\rho$ , поступательная (вдоль трассы) скорость  $u$ , поток  $q = \rho u$  как функция плотности и скорости. Отличие данной двумерной модели от одномерных аналогов состоит во введении боковой скорости  $v$  как скорости перестроения автомобилей из полосы в полосу. Благодаря этому можно описывать транспортные потоки с учетом реальной геометрии магистрали. Кроме того, уравнения содержат дополнительные члены – отвечающие, например, за

ускорение и торможение автомобилей как реакции на окружающую обстановку, время релаксации (реакции водителя). Ниже приведена система уравнений для описания динамики транспортных потоков, состоящая из уравнения неразрывности и двух уравнений движения для двух компонент скорости:

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) = \\ = \frac{\partial}{\partial x} \frac{\tau_x}{2} \left( \frac{\partial}{\partial x}(\rho u^2 + p_x) - f_x + \frac{\partial}{\partial y}(\rho uv) \right) + \frac{\partial}{\partial y} \frac{\tau_y}{2} \left( \frac{\partial}{\partial y}(\rho v^2 + p_y) - f_y + \frac{\partial}{\partial x}(\rho uv) \right); \end{aligned} \quad (5)$$

$$\begin{aligned} \frac{\partial \rho u}{\partial t} + \frac{\partial}{\partial x}(\rho u^2 + p_x) - f_x + \frac{\partial}{\partial y}(\rho uv) = \\ = \frac{\partial}{\partial x} \frac{\tau_x}{2} \left( \frac{\partial}{\partial x}(\rho u^3 + 3p_x u) - 3f_x u \right) + \frac{\partial}{\partial y} \frac{\tau_y}{2} \left( \frac{\partial}{\partial y}(\rho uv^2 + p_y u) - f_y u \right) + \\ + \frac{\partial}{\partial x} \frac{\tau_x}{2} \left( \frac{\partial}{\partial y}(\rho u^2 v + p_y v) - f_y v \right) + \frac{\partial}{\partial y} \frac{\tau_y}{2} \left( \frac{\partial}{\partial x}(\rho u^2 v + p_x v) - f_x v \right); \end{aligned} \quad (6)$$

$$\begin{aligned} \frac{\partial \rho v}{\partial t} + \frac{\partial}{\partial y}(\rho v^2 + p_y) - f_y + \frac{\partial}{\partial x}(\rho uv) = \\ = \frac{\partial}{\partial x} \frac{\tau_x}{2} \left( \frac{\partial}{\partial x}(\rho u^2 v + p_x v) - f_x v \right) + \frac{\partial}{\partial y} \frac{\tau_y}{2} \left( \frac{\partial}{\partial y}(\rho v^3 + 3p_y v) - 3f_y v \right) + \\ + \frac{\partial}{\partial x} \frac{\tau_x}{2} \left( \frac{\partial}{\partial y}(\rho v^2 u + p_y u) - f_y u \right) + \frac{\partial}{\partial y} \frac{\tau_y}{2} \left( \frac{\partial}{\partial x}(\rho v^2 u + p_x u) - f_x u \right) \end{aligned} \quad (7)$$

Здесь  $f_x = a_x \rho$ ,  $f_y = a_y \rho$  – ускоряющая и тормозящая силы,  $a_x = (u_{eq} - u)/T_x$ ,  $a_y = (v_{eq} - v)/T_y$  – компоненты ускорения,  $u_{eq} = u_f (1 - \rho / \rho_{jam})$  – равновесная

скорость, зависящая от  $u_f$  – скорости свободного движения, и  $\rho_{jam}$  – плотности автомобильной «пробки» (при которой движение отсутствует),  
 $T_x = t_{0x} (1 + r\rho / (\rho_{jam} - r\rho))$ ,  $T_y = t_{0y} (1 + r\rho / (\rho_{jam} - r\rho))$  – время релаксации,  
 $p_x = \lambda_x \frac{\rho^{\beta_x}}{\beta_x}$ ,  $p_y = \lambda_y \frac{\rho^{\beta_y}}{\beta_y}$  – аналог давления как реакция водителя на ситуацию впереди автомобиля. Остальные величины являются параметрами.

Отдельно следует отметить, что на практике более удобным оказалось использование вместо уравнения (7) уравнение для введенной авторами равновесной боковой скорости как суммы трех слагаемых:

$$v_{eq} = k_u \rho \frac{\partial u}{\partial y} - k_\rho u \frac{\partial \rho}{\partial y} + k_{des} \frac{u^2}{(x_{des} - x)^2} (y_{des} - y), \quad (8)$$

где первое слагаемое отражает стремление водителя перестроиться в полосу с большей скоростью, второе – перестроиться в полосу с меньшей плотностью, третье – достичь определенной цели с координатами  $(x_{des}, y_{des})$ .

### 1.3 Мезоскопические (кинетические) модели.

По аналогии со статистической физикой, макроскопические уравнения для движения транспорта можно вывести из взаимодействия автомобилей на микроуровне, используя кинетическое уравнение. Впервые этот подход был предложен И. Пригожиным [17] в 1960 году. В отличие от макроскопических моделей, оперирующих плотностью  $\rho$  и средней скоростью  $v$  потока, кинетические модели работают с фазовой плотностью потока  $f(x, v, t)$  – распределением автомобилей по координате и скорости.

Кинетическое уравнение для транспортного потока выглядит следующим образом:

$$\frac{\partial f}{\partial t} + \frac{\partial(\rho f)}{\partial x} = \left( \frac{\partial f}{\partial t} \right)_{\text{int}} + \left( \frac{\partial f}{\partial t} \right)_{\text{rel}} \quad (9)$$

Данное уравнение является уравнением неразрывности в фазовом пространстве, левая часть описывает изменение фазовой плотности за счет кинематического переноса, а члены в правой части отвечают за изменение скоростей автомобилей за счет процессов взаимодействия и релаксации. Взаимодействие здесь – это либо обгон, либо замедление до скорости впереди идущего автомобиля, а релаксация определяет стремление водителя к желаемой скорости.

Члены в правой части могут принимать различный вид, за счет чего на основе кинетического уравнения (9) были созданы более сложные модели. Одна из таких моделей – модель Павери-Фонтана [18], содержащая, помимо фактической, желаемую скорость автомобилей.

Модель Хельбинга [19, 20] также содержит желаемую скорость. Кинетическое уравнение осредняется по скоростям и желаемым скоростям автомобилей, а также выписываются моментные уравнения.

#### **1.4 Микроскопические модели.**

Наиболее важными представителями микроскопических моделей транспортных потоков являются модели следования за лидером. Они описывают динамику трафика с точки зрения индивидуальных единиц водитель-машина. В строгом смысле модели следования за лидером описывают поведение водителя только при наличии взаимодействий с другими автомобилями, а свободный транспортный поток описывается при помощи отдельной модели. В более общем смысле модели следования за лидером включают в себя все дорожные ситуации, такие как ситуации следования за лидером, свободный трафик и также стационарный трафик. В этом случае мы говорим, что микроскопическая модель полная.

Как понятно из названия, основная черта моделей – это «подстройка» движения автомобиля под действия лидера. Основным параметром, с которым работают модели – это интервал (временной или пространственный) до впереди идущего автомобиля.

Одна из простейших моделей класса следования за лидером – это модель оптимальной скорости [21, 22]:

$$\dot{x}_\alpha(t) = \frac{dx_\alpha(t)}{dt} = v_\alpha(t) \quad (10)$$

$$\dot{v} = \frac{v_{\text{opt}}(s) - v}{\tau} \quad (11)$$

Уравнение (10), очевидно, является общим для всех непрерывных микромоделей.

Уравнение для ускорения (11) содержит оптимальную скорость  $v_{\text{opt}}(s)$ , зависящую от расстояния до лидера  $s$ .  $\tau$  – время адаптации. Это уравнение описывает адаптацию действительной скорости  $v = v_\alpha$  к оптимальной скорости  $v_{\text{opt}}(s)$  за время адаптации.

В моделях следования за лидером общий вид ускорения можно записать:

$$\dot{v}_\alpha(t) = \frac{dv_\alpha(t)}{dt} = a_{\text{mic}}(s_\alpha, v_\alpha, v_l) = \tilde{a}_{\text{mic}}(s_\alpha, v_\alpha, \Delta v_\alpha) \quad (12)$$

Определение переменных понятно из рисунка 1 (взято из [23])

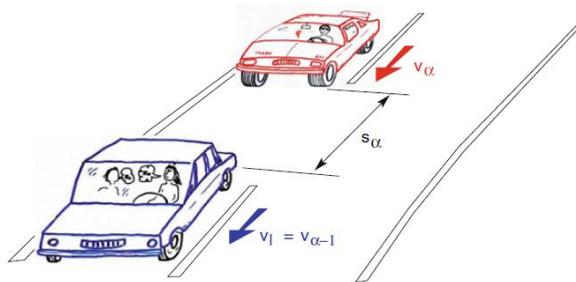


Рисунок 1 – определение переменных в моделях следования за лидером  
Ускорение зависит от скорости автомобиля, разницы скоростей с лидером и расстояния до лидера

В качестве еще одного примера приведем модель разумного водителя Трайбера [24].

$$\dot{v} = a \left[ 1 - \left( \frac{v}{v_0} \right)^\delta - \left( \frac{s^*(v, \Delta v)}{s} \right)^2 \right] \quad (13)$$

Ускорение в модели интеллектуального водителя дается в форме  $\tilde{a}_{mic}(s, v, \Delta v)$  и состоит из двух частей, одна сравнивает текущую скорость  $v$  с желаемой скоростью  $v_0$ , а другая сравнивает текущее расстояние  $s$  с желаемым расстоянием  $s^*$ .  $a$  – типичное ускорение. Желаемое расстояние:

$$s^*(v, \Delta v) = s_0 + \max \left( 0, vT + \frac{v\Delta v}{2\sqrt{ab}} \right) \quad (14)$$

имеет равновесный член  $s_0 + vT$  и динамический член  $v\Delta v / (2\sqrt{ab})$ , который реализует "интеллектуальную" тормозную стратегию.

- При нормальных условиях тормозной маневр "мягкий", то есть, замедление возрастает постепенно до комфортного значения  $b$  и плавно уменьшается до нуля непосредственно перед достижением стационарной ситуации следования за лидером или приходит к полной остановке.

- В критической ситуации замедление выходит за пределы комфортного значения до тех пор, пока не будет предотвращена опасность. Оставшийся тормозной маневр (если это нужно) будет продолжен с регулярным комфортным замедлением  $b$ .

### **1.5 История развития клеточных автоматов.**

Концепция клеточных автоматов (cellular automata, CA) была впервые представлена Джоном фон Нейманом в 1948 году в качестве доклада на симпозиуме по механизмам мозга и поведения в Калифорнийском технологическом институте [25]. За докладом последовала серия работ, целью которой было построение систематической теории, имеющей логико-математическую форму и позволяющей понять как естественные биологические системы, так и аналоговые и цифровые вычислительные машины. Особый интерес для фон Неймана представляли сложные системы, такие, как нервная система человека или большие вычислительные машины, появление которых он предвидел. Далее основной целью его работы стало создание модели системы, способной воспроизводить саму себя [26]. Позже, совместно со Станиславом Уламом, он разработал принцип, описывающий физическую структуру клеточного автомата: сеть из ячеек, которые могут быть либо «включены», либо «выключены».

Вклад фон Неймана в развитие вычислительной техники нельзя переоценить. В частности, он участвовал в создании первого лампового компьютера ЭНИАК и более поздних и совершенных машин. Так называемая архитектура фон Неймана - созданный им широко известный принцип хранения данных и команд в памяти компьютера.

В 70х годах 20 века на основе клеточных автоматов было создано множество так называемых игр-симуляторов, самой известной из которых, без сомнения, является игра «Жизнь» Джона Конвея [27]. Игра ведется на бесконечном или конечном поле, размеченном на клетки. Каждая клетка может находиться в

одном из двух состояний: быть живой или мертвой. На поле имеется некоторое начальное распределение клеток, определяющее исход игры. У каждой клетки есть 8 ближайших соседей, расположенных сверху, снизу, с боков и по диагонали. Если у живой клетки меньше двух или больше трёх живых соседей, она умирает – от «одинокости» или от «перенаселенности». В противном случае клетка остается живой. Мертвая клетка, у которой есть 3 живых соседа, оживает. Игра заканчивается в трех случаях: если все клетки умирают, если на поле периодически возникает одна и та же конфигурация и если с каждым новым шагом на поле ничего не меняется (возникает стабильная конфигурация). В игре появляются разнообразные фигуры, некоторые из которых сохраняют форму, у некоторых форма может повторяться через определенное число поколений, при этом сама фигура может смещаться по полю. Пример конфигураций, возникающих в игре, представлен на рисунке 2.

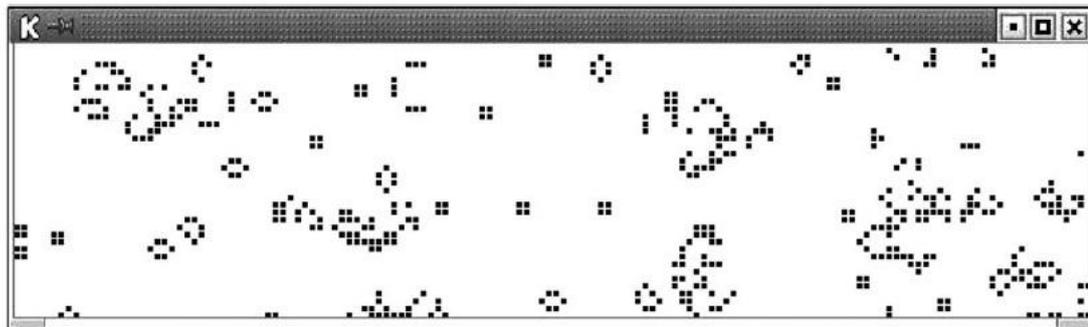


Рисунок 2. Фигуры, возникающие в игре «Жизнь». Черным представлены живые клетки, белым – мертвые

Существуют так называемые «сады Эдема» - фигуры, у которых не может быть предыдущих состояний.

«Жизнь» и подобные игры представляют интерес с точки зрения чистой математики, но и применительно к другим областям знаний.

В 80е годы теория клеточных автоматов завоевала еще большую популярность благодаря работе Стивена Вольфрама «Новый вид науки» [28].

Его основной идеей стало создание нового способа описания природы событий – вместо привычных для классической математики уравнений использовать набор правил (компьютерных программ). Он показал, что даже программы, имеющие в основе самые простые правила, могут иметь сложное, неочевидное поведение. То есть, построение сложного совсем не обязательно должно быть, в свою очередь, тоже сложным. В этом он видел сходство с природой: множество простых механизмов, взаимодействуя, в итоге рождает самые разнообразные комплексные системы. Вот эту эволюцию от простого к сложному, по мнению Вольфрама, лучше всего воспроизводят клеточные автоматы. Вольфрам сформулировал *«принцип вычислительной эквивалентности»*: когда наблюдатель видит поведение системы, не являющееся очевидным и простым, это поведение можно ассоциировать с вычислением аналогичной сложности. Иными словами, функционирование системы настолько же сложно, насколько сложно воспроизведение этого функционирования посредством вычислений. С одной стороны, это означает некоторую конечность возможностей теоретической науки. С другой стороны, даже для сложных систем, с течением времени можно достичь решения.

Принцип вычислительной эквивалентности имеет беспрецедентные следствия для науки и научного мышления. Любые системы, начиная от простых программ и заканчивая нашим мозгом и всей вселенной, в своей основе имеют один и тот же базис, некую эквивалентность происхождения фундаментальных явлений, что позволяет применить к ним единый научный подход.

Теория клеточных автоматов стала самостоятельной областью математики, представляя собой новый интересный объект исследования, который, благодаря специальным эффективным методам, можно изучить глубже, чем когда-либо.

Дискретный подход с простыми правилами вместо уравнений позволяет решать новые задачи физики, которые раньше казались неразрешимыми. Вольфрам также надеется, что клеточные автоматы помогут подойти ближе к созданию так называемой «Теории всего» - теории, охватывающей все фундаментальные взаимодействия.

Большой потенциал у клеточных автоматов в социальных науках – там, где есть большие разнородные системы со многими взаимодействующими элементами. Правила лучше уравнений позволяют отражать такие процессы, ведь для многих социальных явлений уравнения или вовсе не придуманы, или сильно упрощают действительность.

О применении в биологии уже было упомянуто выше – сам фон Нейман считал биологию одной из основных областей применения своей теории. Сложность биологических систем всегда являлась препятствием для создания классической математической теории, позволяющей эту сложность воспроизвести. Зная, что простые программы могут производить высокий уровень сложности, можно с уверенностью сказать, что клеточные автоматы способны воспроизвести многие свойства биологических организмов и систем.

Вольфрам также говорит о возможности применения теории в таких областях, как искусство, философия, технология, теория эволюции, экспериментальная математика, теория хаоса, теория катастроф, искусственный интеллект и так далее.

Конечно, неоценимый вклад был внесен в развитие компьютерной техники, позволяя описывать функции сложных вычислительных машин.

На основе теории клеточных автоматов уже были созданы тысячи математических моделей самых разных процессов, как биологических, так и технических, социальных и экономических.

## 1.6 Теория клеточных автоматов применительно к транспортному моделированию.

В 1992 году Кай Нагель и Михаэль Шрекенберг [1] предложили применить теорию клеточных автоматов к транспортному моделированию. Предложенная ими модель была одномерной, то есть могла описывать лишь однополосное движение. Дорога представляла собой ряд ячеек, каждая из которых могла быть либо пустой, либо занятой одним автомобилем. Каждый автомобиль характеризовался скоростью - целым числом от 0 до некоторого максимального значения  $V_{max}$ . На каждом шаге по времени состояние ячеек обновлялось по определенным правилам – автомобили двигались вдоль ряда ячеек, ускоряясь и замедляясь с соблюдением правил безопасности, а также имея возможность дополнительно снизить скорость с определенной вероятностью без видимых причин. Несмотря на свою простоту, модель хорошо воспроизводила имеющиеся на тот момент эмпирические данные. Работа Нагеля и Шрекенберга положила начало широкому использованию клеточных автоматов в транспортном моделировании. Учеными со всего мира было создано множество вариантов моделей транспортных потоков на ее основе.

Ранее данный подход представлялся наиболее перспективным для детального описания локальных дорожных ситуаций на небольших расстояниях, так как модели являются достаточно гибкими благодаря возможности реализовать любую стратегию водителя без существенных алгоритмических затрат. Однако в связи с возможностями существующей в настоящее время вычислительной техники сверхвысокой производительности модели данного типа могут с успехом применяться и для моделирования движения на больших дорожных сетях.

Благодаря развитию компьютерной техники появилась возможность учитывать в создаваемых моделях все больше нюансов поведения

участников движения, а благодаря развитию самой теории транспортных потоков – новые представления о закономерностях, наблюдаемых в данных измерений. Также в последнее время собирается и хранится большое количество данных с датчиков и камер наблюдения, а значит, и возрастают требования к создаваемым математическим моделям.

В мегаполисах по всему миру для управления дорожным движением в реальном времени внедряется так называемая Интеллектуальная Транспортная Система (ИТС). Моделирование различных ситуаций и мер, принимаемых в ответ на возникающие проблемы – аварии, заторы и т.п. – является необходимой частью управления. Поэтому создание новых моделей, позволяющих быстро, но точно делать такого рода прогнозы, является актуальной задачей на сегодняшний день.

## **1.7 Обзор моделей на основе теории клеточных автоматов.**

Модели можно разделить на 2 группы: детерминистические и стохастические. Рассмотрим сначала детерминистические.

### *1.7.1 Правило 184 Вольфрама*

Данная модель относится к классу элементарных клеточных автоматов. Это группа из 256 ( $2^{2^3}$ , согласно формуле (3)) одномерных моделей с числом соседей 3. Стивен Вольфрам классифицировал все эти модели в 1983 году, а сейчас их можно найти в Атласе Вольфрама на сайте [29]

Расчетная область в модели одномерная и представляет собой цепочку клеток. У каждой клетки имеется два соседа – справа и слева. Как всегда, состояние соседей определяет состояние клетки на следующем шаге по времени. Занятая клетка обозначается 1, а свободная 0. Всего у трех клеток может быть  $2^3=8$  конфигураций – (111), (110), (101), (100), (011), (010), (001), (000). Если упорядочить их по убыванию, а затем выписать результат, то есть состояние на следующем шаге для каждого из вариантов, получим {1, 0, 1, 1,

1, 0, 0, 0}, а это - 184 в двоичной системе, отсюда и название модели. Наглядное графическое представление можно увидеть на рисунке 3 (взято из Атласа Вольфрама).

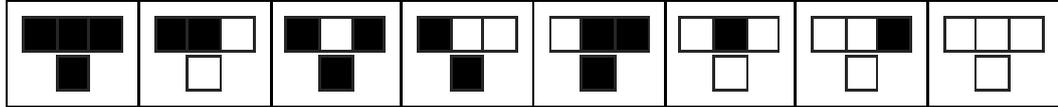


Рисунок 3. Графическое представление правила 184. В верхней строке – клетка на текущем шаге по времени в окружении двух соседей, в нижней строке – состояние клетки на следующем временном шаге, в зависимости от состояния соседей

Если рассмотреть состояния 1 как автомобили, а состояния 0 – как пустые клетки, получается, что на каждом шаге автомобили перемещаются на 1 ячейку направо при наличии свободной ячейки по ходу движения и остаются на месте, если ячейка занята.

Для того, чтобы выписать правила обновления ячеек, перейдем к рассмотрению автомобилей (непустых клеток, 1).

Правила движения для машины с номером  $i$  следующие:

$$v_i(t) = \min \{g_{si}(t-1), 1\} \quad (15)$$

$$x_i(t) = x_i(t-1) + v_i(t) \quad (16)$$

Здесь  $g_{si}(t-1)$  - расстояние до впереди идущего автомобиля на предыдущем шаге по времени,  $v_i(t)$  – текущая скорость  $i$ -го автомобиля,  $x_i(t)$  и  $x_i(t-1)$  - координата  $i$ -го автомобиля на шагах по времени  $t$  и  $t-1$ . Правило (15) обозначает, что автомобили стремятся двигаться со скоростью 1, если это отвечает правилам безопасности. Правило (16) определяет конечное положение  $i$ -го автомобиля в системе на текущем шаге.

### 1.7.2 Модель Нагеля-Шрекенберга

Трасса в модели Нагеля-Шрекенберга [1] представляется в виде одномерной решетки, каждая ячейка которой может быть либо пустой, либо содержать частицу, обозначающую транспортное средство. Частицы перемещаются из одной ячейки в другую (свободную) в одном направлении. В случае однополосного движения они не могут обгонять друг друга. Вся система – пространство, время, скорость – дискретна. Предметом рассмотрения является состояние всей системы в отдельные последовательные моменты времени. Скорость показывает, на сколько ячеек автомобиль перемещается за один шаг по времени. Ускорение происходит мгновенно между шагами. На каждом слое по времени происходит обновление состояния системы по определённым правилам с определёнными вероятностями. Обновление в модели Нагеля-Шрекенберга состоит из четырех шагов:

- 1) Ускорение. Скорость автомобиля  $i$  увеличивается на единицу, если максимальная разрешенная скорость не достигнута:

$$V_i \rightarrow \min(V_i + 1, V_{max}).$$

- 2) Торможение. Скорость автомобиля уменьшается на единицу, если есть угроза столкновения с впереди идущим автомобилем:

$$V_i \rightarrow \min(V_i, D_i - 1), \quad \text{где } D_i \text{ - расстояние до впереди идущего автомобиля.}$$

- 3) Случайные возмущения. Если скорость автомобиля положительна, то она может быть уменьшена на единицу с некоторой вероятностью:

$$V_i \rightarrow \max(V_i - 1, 0) \text{ с вероятностью } p.$$

- 4) Движение. Каждый автомобиль продвигается вперед на количество ячеек, соответствующее его новой скорости после выполнения предыдущих шагов:

$$X_i \rightarrow X_i + V_i.$$

Для упрощения записи считаем, что скорость и расстояние измеряются в ячейках, а время безразмерно. По этой причине величины можно складывать, вычитать и сравнивать друг с другом.

Схема перехода от одного временного слоя к другому в одномерной модели на основе СА представлена на рисунке 4 (из [1]).

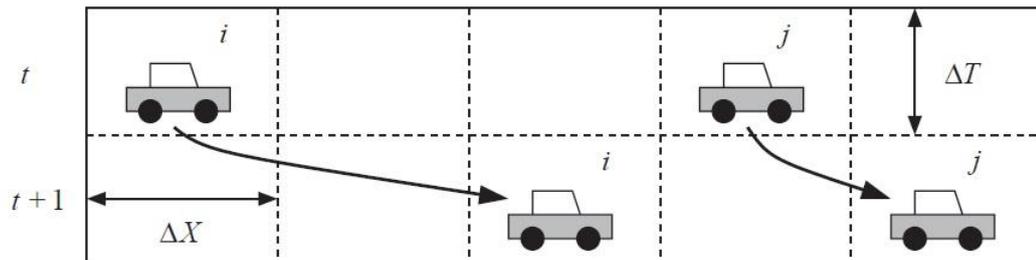


Рисунок 4. Переход к следующему шагу по времени в одномерной КА модели

Здесь  $t$  – номер шага по времени,  $i$  и  $j$  – номера автомобилей,  $\Delta T$  и  $\Delta X$  – величина шага по времени и размер ячейки соответственно.

Каждый шаг по времени состояние ячеек обновляется, автомобили передвигаются вдоль решетки на целое число ячеек.

В статье приводится ряд тестовых расчетов с различными начальными и граничными условиями для демонстрации адекватности модели, проводится сравнение с экспериментальными данными.

В статье сделана оценка значений размера ячейки и шага по времени. Длина ячейки равна 7.5 м, что соответствует средней длине участка дороги, занимаемым автомобилем, стоящим в пробке (исходя из экспериментальных значений). Максимальная скорость  $V_{max} = 5 \frac{\text{ячеек}}{\text{шаг по времени}}$ . Исходя из проведенных вычислительных экспериментов, средняя скорость, отвечающая свободному движению, оценивается как  $4.5 \frac{\text{ячейки}}{\text{шаг по времени}}$ ; этому значению ставится в соответствие экспериментальное значение скорости свободного

движения – 120 км/ч (33,3 м/с). Простой расчет позволяет, исходя из этих данных, оценить шаг по времени, который получается примерно равным 1

$$\text{секунде: } \frac{7.5 \frac{\text{метров}}{\text{ячейку}} \times 4.5 \frac{\text{ячеек}}{\text{шаг по времени}}}{33.3 \text{ м/с}} \approx 1 \text{ с.}$$

Отсюда следует, что «шаг по скорости»  $\Delta V = \frac{\Delta X}{\Delta T} = 27 \text{ км/ч}$ .

Эти значения используются в большинстве созданных впоследствии моделей.

Модель автотранспортных потоков на основе теории клеточных автоматов Нагеля-Шрекенберга стала основой для многих последующих, более сложных, моделей, созданных исследователями из разных стран.

### *1.7.3 Модель Кернера-Кленова*

В качестве примера более сложной и детальной модели приведем пример модели Кернера-Кленова [5,6]. Для соответствия теории трех фаз Б. Кернера модель содержит переускорение. Обновление состояния ячеек происходит по следующим правилам:

- (a) переускорение за счет перестроения в более быстрый ряд
- (b) сравнение расстояния до машины впереди  $g$  с расстоянием синхронизации  $G$  (см. рисунок 5)
- (c) адаптация скорости к скорости машины впереди в пределах расстояния синхронизации  $G$
- (d) случайное переускорение в пределах расстояния синхронизации без смены полосы
- (e) шаги 1-4 из модели Нагеля-Шрекенберга.

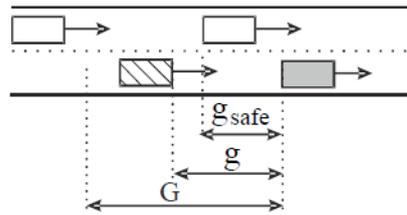


Рисунок 5 – сравнение расстояния до впереди идущего автомобиля  $g$  с расстоянием синхронизации  $G$ .  $g_{\text{safe}}$  – безопасное расстояние

Благодаря математическому описанию стохастического переускорения с задержкой и эффекта адаптации внутри синхронизованного потока, в разработанной модели переход от свободного к плотному потоку - это  $F \rightarrow S$  переход (согласно теории трех фаз Кернера) в метастабильном свободном потоке, как наблюдается во всех эмпирических данных.

#### 1.7.4 Подход Буслаева – моделирование на разветвленных структурах

При помощи клеточных автоматов можно моделировать большие сети дорог. В качестве примера приведем модель, созданную А. П. Буслаевым [30] и коллегами в МАДИ.

В основе его подхода – кольцевые структуры из клеточных автоматов с общими ячейками, за которые происходит конкуренция. Линейная структура колец называется ожерельем, а двумерная – кольчугой (рисунок 3). Кольчуга может иметь различную топологию, например, тор.

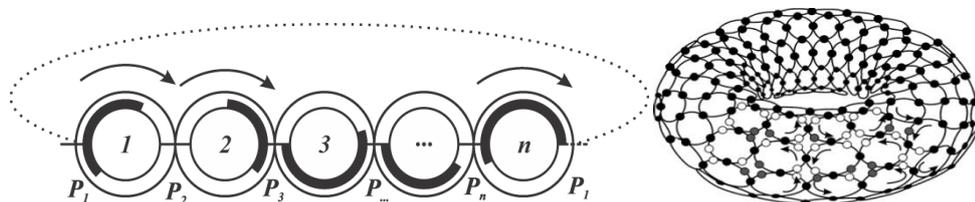


Рисунок 3 – ожерелье и кольчуга на торе.

Движение по кольчуге соответствует движению по сети с перекрестками. Изучение динамики таких систем - сложная задача. В системе может возникать состояние синергии, т.е. в состоянии, в котором конфликты невозможны в текущий момент и во все моменты в будущем.

### *1.7.5 Другие модели*

Один из самых содержательных обзоров по транспортному моделированию на основе теории клеточных автоматов – [31] 2005 года. Он содержит описание многочисленных моделей, созданных в период до 2005 года, в исторической последовательности.

Подход СА применяется также для решения таких задач, как моделирование движения с велосипедистами и автобусами, безопасного вождения, исследование влияния поворотов и т.д. [32-36].

## **1.8 Современные исследования транспортных потоков**

Моделирование автотранспортных потоков активно развивается как в нашей стране, так и за рубежом, и в течение последних лет появилось большое число работ, посвященных этой тематике.

Появляются публикации, посвященные теории динамики транспортных потоков и различным способам моделирования автотранспортной сети. Например, в работе [37] исследуются основные теоретические аспекты динамики транспортного потока. Отмечается, что перегруженность в трафике можно интерпретировать в терминах неустойчивости уравнения динамического движения. Проводится аналогия с теорией Ландау, исследуется связь с неустойчивостью уравнения движения и переходом между различными состояниями движения.

Различные модели пассажирских потоков анализируются в статье [38]. В статье обоснована необходимость и важность применения имитационного моделирования, а также правильность его применения для исследования пассажирских потоков в транспортных системах.

Вопросам прогнозирования посвящена статья [39]. Здесь представлена комплексная методика прогноза автотранспортных и пассажирских потоков в

транспортной системе крупного города на основе моделей динамического распределения потоков по сети.

В статье [40] рассматривается методика виртуального моделирования в дорожно-транспортной области. Приводятся способы реализации в виде программного обеспечения, плюсы и минусы такого подхода, а также возможные проблемы.

Имитационное моделирование представлено в статье [41]. Здесь рассматриваются вопросы моделирования городских транспортных потоков в среде имитационного моделирования AnyLogic.

Большое количество работ посвящено вопросам управления автотранспортом. Можно привести, например, работы [42, 43]. В последней статье в качестве инструмента выбран модуль VISUM программного комплекса имитационного моделирования PTV Vision.

Современные исследования динамики транспортных потоков идут, в основном, по пути усложнения уже существующих моделей. Примером могут служить следующие публикации, посвященные макроскопическим моделям гидродинамического типа.

Авторы статьи [44] описывают предложенную ими новую lattice-модель, с помощью которой исследуют, как объем памяти водителя влияет на динамику автотранспортного потока. Анализ линейной устойчивости показывает, что временная длина памяти водителя оказывает важное влияние на устойчивость транспортного потока.

Те же авторы в статье [45] для исследования динамики потока при движении по криволинейной дороге предлагают расширенную одномерную гидродинамическую lattice-модель. Условие устойчивости они получают на основе анализа линейной устойчивости.

В статье [46] исследуется влияние поведения водителя на движение по криволинейной дороге с использованием гидродинамического lattice-подхода.

Дальнейшее усовершенствование получают и модели микроскопического типа. В статье [47] рассматривается модель следования за лидером, которая учитывает характеристики автомобилей, следующих перед рассматриваемым автомобилем на соседней полосе. Обнаружено, что максимальная длина очереди и непостоянство ускорения оказывают большое влияние на смешанный поток трафика на перекрестке с различным временем циклов сигнала светофора и значительно зависят от интенсивности движения.

Модель следования за лидером рассматривается и в статье [48]. Авторы рассматривают микроскопическую модель автомобильного трафика с запаздывающим аргументом. Важным результатом работы является то, что данная модель способна воспроизводить такой феномен, как stop-and-go волны.

В статье [49] предлагается расширенная модель следования за лидером для моделирования движения транспортного потока с учетом эффекта автомобильного сигнала. Условие устойчивости этой модели получается на основе анализа линейной устойчивости. Фазовая диаграмма показывает, что эффект автомобильного сигнала играет важную роль в улучшении стабилизации системы движения.

Авторы статьи [50] описывают расширенную модель следования за лидером, которая может использоваться для описания динамических характеристик смешанного движения с пешеходами, идущими по соседней полосе.

Большое количество публикаций посвящено тем или иным модификациям моделей на основе теории клеточных автоматов.

В статье [51] приведено интересное обобщение теории клеточных автоматов на случай морских перевозок в применении к морскому транспорту. В этом случае правила дискретизации по пространству дополняются правилами картографирования.

Авторы статьи [52] исследуют пропускную способность автомагистрали с двумя въездами и одним промежуточным съездом между ними также с использованием модели клеточных автоматов. Целью исследований является максимизация пропускной способности системы, путем установления оптимального потока для двух въездов.

В статье [53] представлена недавно модифицированная модель ККВ (Kerner-Klenov-Wolf), включающая различные типы транспортных средств. Вводится переменная чувствительность водителя к колебаниям скорости. Делаются выводы о влиянии изменения скорости одного или нескольких транспортных средств на общую скорость потока при различной интенсивности начального потока.

В работе [54] представлена надежная при численной реализации модель клеточных автоматов, ориентированная на то, чтобы точно воспроизводить замедление и ускорение в соответствии с реалистичными реакциями водителей, когда рассматриваются транспортные средства с различными возможностями замедления.

Различные типы транспортных средств рассматриваются в статьях [55, 56].

Исследованию «эффекта упреждения» посвящена статья [57]. К наиболее интересным статьям можно отнести также публикации [58-60]. Из последних русскоязычных публикаций можно отметить статью, посвященную исследованию влияния на объем трафика разделения транспортного потока на подпотоки [61].

Активно ведутся исследования транспортных потоков в МФТИ. Помимо множества важнейших исследовательских работ ([62-65]), учеными Физтеха было выпущено учебное пособие по транспортному моделированию [66] под редакцией А. В. Гасникова. В создании принимали участие ведущие специалисты в своих областях. Так, глава 1 пособия написана проф. Е. А. Нурминским и доц. Н. Б. Шамрай (ИАПУ ДВО РАН) и посвящена применению теории бескоалиционного равновесия для расчета транспортной сети при условии стационарности потоков и моделям построения матрицы корреспонденций. В написании главы 2, посвященной математическим моделям транспортных потоков, приняли также участие доц. С. Л. Кленов (кафедра общей физики МФТИ) и доц. Я. А. Холодов (кафедра вычислительной математики МФТИ). Глава 3, посвященная теории трех фаз Кернера транспортного потока, всецело написана С. Л. Кленовым.

Следует отметить работы А. Б. Киселева (МехМат МГУ) с соавторами [67, 68], посвященные вопросам организации дорожного движения и исследованию неустойчивости потоков при большом числе перестроений.

### **Выводы по Главе 1**

Глава 1 содержит обзор основных методов и подходов к моделированию трафика: макроскопический, мезоскопический и микроскопический подходы. Представлены плюсы и минусы разных подходов. Представлен обзор клеточных автоматов как самостоятельной математической теории и применительно к теории транспортного моделирования.

Как показано в Главе 1, теория клеточных автоматов широко применяется в транспортном моделировании, однако, этот подход далеко не исчерпал себя. Целью данной работы является создание новой модели с упором на поведение водителей при взаимодействии с элементами дорожной инфраструктуры и с другими водителями.

## ГЛАВА 2. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ТРАНСПОРТНЫХ ПОТОКОВ НА ОСНОВЕ ТЕОРИИ КЛЕТОЧНЫХ АВТОМАТОВ ДЛЯ МОДЕЛИРОВАНИЯ ТРАФИКА НА ГОРОДСКОЙ УЛИЧНО-ДОРОЖНОЙ СЕТИ. ОСНОВНЫЕ АЛГОРИТМЫ ДВИЖЕНИЯ. РАЗЛИЧНЫЕ ЭЛЕМЕНТЫ УДС, ПРИМЕРЫ ТЕСТОВЫХ РАСЧЕТОВ

### 2.1 Построение двумерной модели

В основу многополосной модели, созданной автором диссертации, положена одномерная модель Нагеля-Шрекенберга [1], использующая идеологию клеточных автоматов.

Ранее этот подход представлялся наиболее перспективным для детального описания локальных дорожных ситуаций на небольших расстояниях, так как модели являются достаточно гибкими благодаря возможности реализовать взаимодействие водителей с дорожно-транспортной инфраструктурой и между собой без существенных алгоритмических затрат. Однако в связи с существующей в настоящее время вычислительной техникой сверхвысокой производительности модели данного типа могут с успехом применяться и для моделирования движения на больших дорожных сетях.

Трасса в этом случае представляет собой двумерную решетку, где количество ячеек в поперечном направлении соответствует числу полос трассы, как показано на рисунке 6:

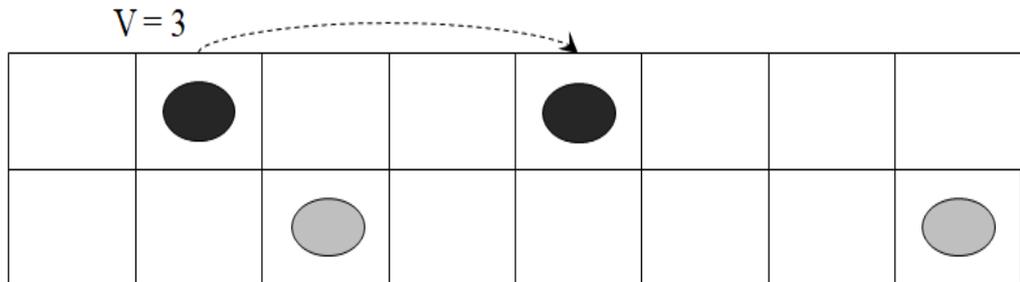


Рисунок 6 - Расчетная область в обобщенной двумерной микроскопической модели

В многополосной модели разрешены перестроения машин из полосы в полосу и обгоны. Процесс обновления состояний ячеек делится на два шага:

- смена полосы при необходимости и наличии возможности;
- движение вперед по правилам однополосного движения.

Размер ячейки расчетной области равен 7,5 метрам в длину и 3,5 метрам в ширину. Эти параметры соответствуют средней площади, занимаемой легковым автомобилем, стоящим на дороге, с учетом минимального необходимого расстояния до соседних автотранспортных средств - АТС. Ширина ячейки соответствует стандартной ширине полосы автомобильной дороги обычного типа (нескоростная дорога).

В двумерную микроскопическую модель в качестве параметров входят две вероятностные характеристики: вероятность спонтанного снижения скорости из модели Нагеля-Шрекенберга и вероятность остаться в своей полосе в случае, когда перестроение возможно и выгодно водителю. Первый параметр соответствует ситуациям, когда водитель, например, посчитал состояние дорожного полотна потенциально опасным или вынужден отвлечься на что-то во время движения и по этой причине принимает решение временно снизить скорость. Второй параметр характеризует наличие «спокойных» водителей, не склонных лишней раз менять полосу движения, даже если это позволит ехать быстрее. Другой причиной отказа от перестроения может быть нерешительность водителя или отсутствие опыта.

Каждому автомобилю приписывается уникальный номер (ID), по которому можно отследить его положение в системе. Как и в случае одномерного движения, АТС характеризуется текущей и максимальной скоростью. Скорость принимает целые значения от 0 до 4, а при высоких скоростных режимах, и более. Максимальная скорость равна  $V_{max} = 4 \frac{\text{ячейки}}{\text{шаг по времени}}$ . Если шаг по времени равен одной секунде, то дискретному значению скорости 4

соответствует скорость 108 километров в час, то есть скорость свободного движения автомобиля по автомагистрали. Так как правилами дорожного движения в России скорость ограничена 110 км/ч, нам нет смысла рассматривать  $V_{max} = 5 \frac{\text{ячеек}}{\text{шаг по времени}} = 135 \text{ км/ч}$ , как в модели Нагеля-Шрекенберга.

Ускорение в модели – это число единиц, на которое автомобиль может поменять скорость за один временной шаг. Важно сопоставлять это значение с реальным возможным ускорением автомобиля при движении. С этой точки зрения для более адекватного описания физического движения автомобиля при заданных параметрах ячеек и шага по времени не имеет смысла менять скорость автомобиля более чем на одну единицу за один шаг по времени.

Модель обеспечивает возможность достижения автомобилями цели. Целями автомобилей при многополосном движении могут быть, например, съезд с дороги или поворот в определенную сторону на светофоре. В обоих случаях машины, начиная с определенного момента времени, стремятся перестроиться в целевую полосу, игнорируя значения плотности и скорости на ней. Таким образом, в описываемой модели для каждой машины необходимо хранить параметр цели. Некоторые автомобили не достигают цели, если не смогли вовремя перестроиться в полосу, из которой разрешен поворот или съезд. Это происходит из-за того, что нужная полоса занята другими транспортными средствами. Модель предусматривает гибкий подход к достижению автомобилем его цели: число автомобилей, которые не достигли цели, может уменьшаться за счет разрешения поворотов из большего числа полос, усложнения алгоритма перестроения или за счет реализации алгоритма «вежливый водитель». Алгоритм заставляет водителей в целевой полосе пропускать машины, которые хотят перестроиться.

Каждый автомобиль в модели имеет набор параметров, таких как:

- уникальный номер (ID),

- текущая скорость,
- максимальная скорость,
- конечная цель.

Имеются также вспомогательные параметры, упрощающие выполнение логических операций. О них будет сказано далее.

В модели, однако, мы оперируем именно с ячейками, а не с автомобилями. Наличие автомобиля и набор его характеристик – это свойство данной ячейки на данном шаге по времени. Все параметры, кроме скорости, в пустых ячейках равны значениям по умолчанию, мы их не используем. Если скорость в ячейке с автомобилем принимает значение от 0 до 4 и соответствует скорости автомобиля, то параметр скорости пустой ячейки полагается равным -1, ячейки с неподвижным препятствием -10. Эти значения используются для построения алгоритмов движения.

Программный комплекс, созданный автором в рамках данной диссертационной работы и основанный на представленной модели, зарегистрирован под названием САМ-2D (“Cellular Automata 2-Dimensional”). Для простоты изложения разработанную модель будем также называть САМ-2D.

Далее приведем основные алгоритмы, разработанные автором включенные в модель и позволяющие описывать движение в различных обстоятельствах. Алгоритмы будут приведены в виде блок-схем и/или псевдокода и дополнены словесным описанием.

## **2.2 Общий алгоритм движения**

Рассмотрим сначала самый простой и самый общий случай – движение по участку дороги без перекрестков, светофоров, сужений/расширений и т.д.

Итак, на каждом шаге по времени обновление состояния ячеек в модели происходит в два этапа:

- А) Для каждого АТС выясняется возможность и необходимость смены полосы. Производится смена полосы.
- Б) Производится движение вперед по выбранной полосе по правилам однополосного движения.

В самом общем виде алгоритм смены полосы А) можно записать так:

### 2.2.1 Алгоритм a1- общий алгоритм смены полосы

- 1) Проверка четности номера шага по времени. На четных шагах происходит смена полосы направо, на нечетных – налево: если  $step\%2=0$   $wmr=true$ ,  $wml=false$ ; иначе  $wmr=false$ ,  $wml=true$ .

Здесь  $wml(r)=true/false$  – переменная, обозначающая, состоится или не состоится смена полосы налево/направо соответственно. Изначально ее значение – true (перестроение состоится). Дальнейшая последовательная проверка условий переопределяет значение переменной (или может оставлять без изменений).

- 2) Выгодно ли перестроение: если  $d_{m,k\pm 1} \leq d_{m,k}$  или  $v_{m,k\pm 1} < v_{m,k}$  то  $wml(r)=false$  (здесь  $d_{m,k}$  – расстояние до а/м в данной полосе,  $d_{m,k\pm 1}$  – расстояние до а/м в целевой полосе,  $v_{m,k}$  и  $v_{m,k\pm 1}$  – скорости этих автомобилей,  $m$  – номер ячейки,  $k$  – номер полосы).
- 3) Есть ли препятствие на этой полосе: если существует  $m: i < m \leq i+10$ ,  $v_{m,k}=-10$  то  $wml(r)=true$  ( $m$  – номер ячейки впереди,  $v_{m,k}=-10$  соответствует неподвижному препятствию).
- 4) Свободна ли целевая ячейка: если  $v_{i,k\pm 1} \neq -1$  то  $wml(r)=false$  ( $v_{i,k\pm 1} = -1$  соответствует пустой ячейке)

- 5) Условие безопасности: если  $d^{prev}_{m,k\pm 1} < V$  то  $wml(r)=false$  ( $d^{prev}_{m,k\pm 1}$  - расстояние до предыдущего автомобиля в целевой полосе,  $v^{prev}_{i,k\pm 1}$  - его скорость,  $V$  – величина, выбираемая исходя из стиля вождения данного водителя.  $V=v_{max}$  или  $V=v^{prev}_{i,k\pm 1}$  для «осторожных» или «агрессивных» водителей соответственно, более подробно об этом будет рассказано ниже).
- б) Стохастический элемент: если  $p < rand()$  то  $wml(r)=false$  ( $p$  – заданная вероятность,  $rand()$  – случайная величина).

Как и в оригинальной модели Нагеля-Шрекенберга считаем, что скорость и расстояние измеряются в ячейках, а время безразмерно и эти величины можно складывать, вычитать и сравнивать друг с другом.

Шаг 1 отвечает за проверку четности номера шага. Перестроение направо на четных шагах, налево на нечетных шагах выполняется с целью избежания конфликтов при числе полос больше двух, когда автомобили, расположенные «через ряд» друг от друга, могут претендовать на одну и ту же ячейку в ряду посередине. Согласно правилам дорожного движения (ПДД), автомобили должны руководствоваться правилом «помеха справа», то есть автомобиль из ряда левее должен пропустить автомобиль из ряда правее при таком конфликте. Однако это требует усложнения взаимодействия между автомобилями из разных рядов, то есть увеличения числа соседей для конкретной ячейки в терминах клеточных автоматов. Такое усложнение возможно в будущих версиях модели.

Шаг 2 определяет желание водителей двигаться в более быстром и/или более свободном ряду.

Шаг 3 проверяет, есть ли препятствие впереди на текущей полосе (если оно есть, водитель принимает решение в пользу перестроения, даже если целевой ряд более медленный). Также происходит проверка наличия препятствия в

целевой полосе. При наличии препятствия запускается алгоритм объезда широкого препятствия, более подробно о котором будет рассказано далее.

Шаг 4 определяет, свободна ли целевая ячейка.

Шаг 5 проверяет условие безопасности. Для безопасного перестроения расстояние между целевой ячейкой и автомобилем, движущимся по целевой полосе и подъезжающим к этой ячейке, должно быть достаточным. Достаточность расстояния определяется скоростью автомобиля в целевой полосе и стилем вождения управляющего перестраивающимся транспортным средством. Соответственно, расстояние, которое водитель считает безопасным, может отличаться.

На шаге 6 перестроение может с некоторой вероятностью отмениться, даже если выполнены все условия для смены полосы.

Для наглядности общий алгоритм смены полосы представлен на рисунке 7 в виде блок-схемы.

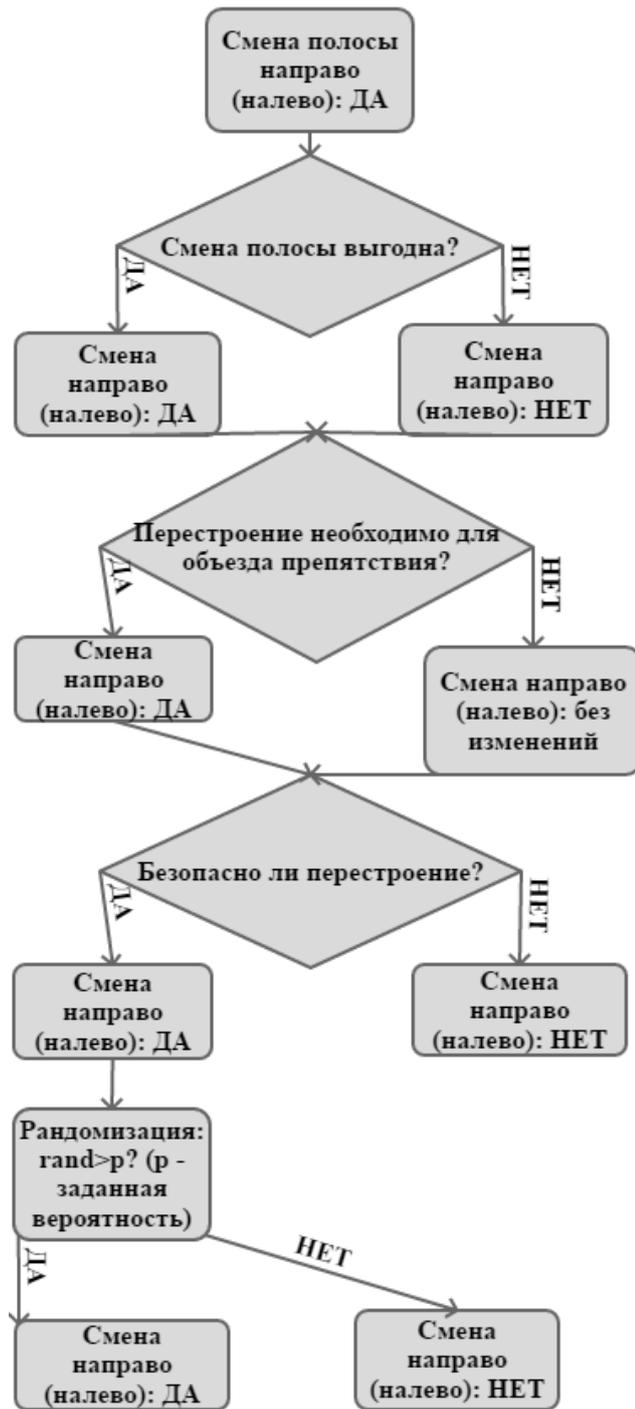


Рисунок 7. Общий алгоритм смены полосы

Подшаг Б) (алгоритм *a2*) выглядит так же, как в модели Нагеля-Шрекенберга:

### 2.2.2 Алгоритм a2 - общий алгоритм движения вперед

- 1) Ускорение. Скорость автомобиля увеличивается на единицу, если максимальная разрешенная скорость не достигнута:

$$v_{i,k} \rightarrow \min (v_{i,k} + 1, V_{max}).$$

Это отображает желание водителя двигаться с максимально разрешенной скоростью.

- 2) Торможение. Скорость автомобиля уменьшается на единицу, если есть угроза столкновения с впереди идущим автомобилем:

$$v_{i,k} \rightarrow \min (v_{i,k}, d_{m,k} - 1), \text{ где } d_{m,k} - \text{расстояние до впереди идущего автомобиля. Данное условие – условие безопасности.}$$

- 3) Случайные возмущения. Если скорость автомобиля положительна, то она может быть уменьшена на единицу с некоторой вероятностью:

$$v_{i,k} \rightarrow \max (v_{i,k} - 1, 0) \text{ с вероятностью } p.$$

- 4) Движение. Выполнив пп. 1-3, мы определили скорость автомобиля на текущем шаге по времени и можем теперь переставить его по системе ячеек на расстояние, соответствующее данной скорости. Каждый автомобиль продвигается вперед на данное количество ячеек:

$$x_{i,k} \rightarrow x_{i,k} + v_{i,k}.$$

### 2.3 Достижение цели

Рассмотрим теперь более сложные случаи. Если элемент улично-дорожной сети - не просто прямой участок дороги, а, например, перекресток или разворот, у каждого автомобиля появляется несколько вариантов выезда с такого элемента.

Модель предусматривает достижение автомобилем цели. Цель определяется в рамках одного элемента улично-дорожной сети и соответствует номеру дороги, по которой машина должна выехать из расчетной области. Для обеспечения движения по сложным элементам УДС каждый такой элемент в модели в общем случае состоит из четырех дорог (направлений движения),

моделирование на каждом из них происходит независимо от других. Взаимодействие между дорогами описывается отдельными алгоритмами, отличными для каждого конкретного элемента.

#### 2.4 Движение на элементах улично-дорожной сети

Для созданной модели разработаны алгоритмы движения на элементах улично-дорожной сети с управляющей инфраструктурой. На рисунке 5 представлены примеры моделирования движения на таких элементах.



Рисунок 8 - элементы УДС

Мелкие квадраты представляют автомобили, разные оттенки отвечают разным целям. Более крупные квадраты – это светофоры. Светлый оттенок обозначает зеленый сигнал, темный оттенок – красный. Представленные на рисунке 8 изображения – это стоп-кадры визуализации расчетов, выполненных при помощи созданной модели САМ-2D.

Рассмотрим алгоритмы движения на участках УДС более подробно.

### 2.4.1 Алгоритм а3 перестроения с достижением цели (движение по участку дороги с выездом)

Рассмотрим теперь простейший случай, включающий достижение цели: участок дороги с выездом. Наличие выезда обозначает, что у автомобилей теперь есть две возможные цели: проехать участок прямо или воспользоваться съездом. В данном случае подшаг (А) общего алгоритма движения модифицируется.

1. Проверка четности номера шага по времени. На четных шагах происходит смена полосы направо, на нечетных – налево: если  $step \% 2 = 0$   $wmr = true$ ,  $wml = false$ ; иначе  $wmr = false$ ,  $wml = true$ .
- 7) Выгодно ли перестроение: если  $d_{m,k\pm 1} \leq d_{m,k}$  или  $v_{m,k\pm 1} < v_{m,k}$  то  $wml(r) = false$  (здесь  $d_{m,k}$  – расстояние до а/м в данной полосе,  $d_{m,k\pm 1}$  – расстояние до а/м в целевой полосе,  $v_{m,k}$  и  $v_{m,k\pm 1}$  – скорости этих автомобилей,  $m$  – номер ячейки,  $k$  – номер полосы).
2. Есть ли препятствие на этой полосе: если существует  $m: i < m \leq i+10$ ,  $v_{m,k} = -10$  то  $wml(r) = true$  ( $m$  – номер ячейки впереди,  $v_{m,k} = -10$  соответствует неподвижному препятствию).
3. Перестроение в нужный ряд вблизи цели: если  $d_{i,k} < D$  (где  $d_{i,k}$  – расстояние до цели, а  $D$  – фиксированное расстояние - параметр моделирования) и целевая полоса находится слева (справа), то  $wml(r) = true$ , если  $d_{i,k} \geq D$   $wml(r) = wml(r)$ .
4. Свободна ли целевая ячейка: если  $v_{i,k\pm 1} \neq -1$  то  $wml(r) = false$  ( $v_{i,k\pm 1} = -1$  соответствует пустой ячейке)
5. Условие безопасности: если  $d^{prev}_{m,k\pm 1} < V$  то  $wml(r) = false$  ( $d^{prev}_{m,k\pm 1}$  – расстояние до предыдущего автомобиля в целевой полосе,  $v^{prev}_{i,k\pm 1}$  – его скорость,  $V$  – величина, выбираемая исходя из стиля вождения данного водителя.  $V = v_{max}$  или  $V = v^{prev}_{i,k\pm 1}$  для «осторожных» или «агрессивных»

водителей соответственно, более подробно об этом будет рассказано ниже).

- б. Стохастический элемент: если  $p < rand()$  то  $wml(r) = false$  ( $p$  – заданная вероятность).

Отличие от общего алгоритма перестроения – в пункте 4. Вблизи от цели, при расстоянии до цели  $d_{i,k} < D$ , где  $D$  – фиксированное расстояние (параметр моделирования, зависящее от типа дороги и режима движения: чем больше скорость автомобилей, тем больше  $D$ ), в алгоритм добавляется ветвь, переопределяющая значение переменных  $wmr/wml$  в зависимости от того, нужно ли перестроение для достижения цели.

На рисунке 9 представлена блок-схема алгоритма перестроения автомобилей при наличии цели.

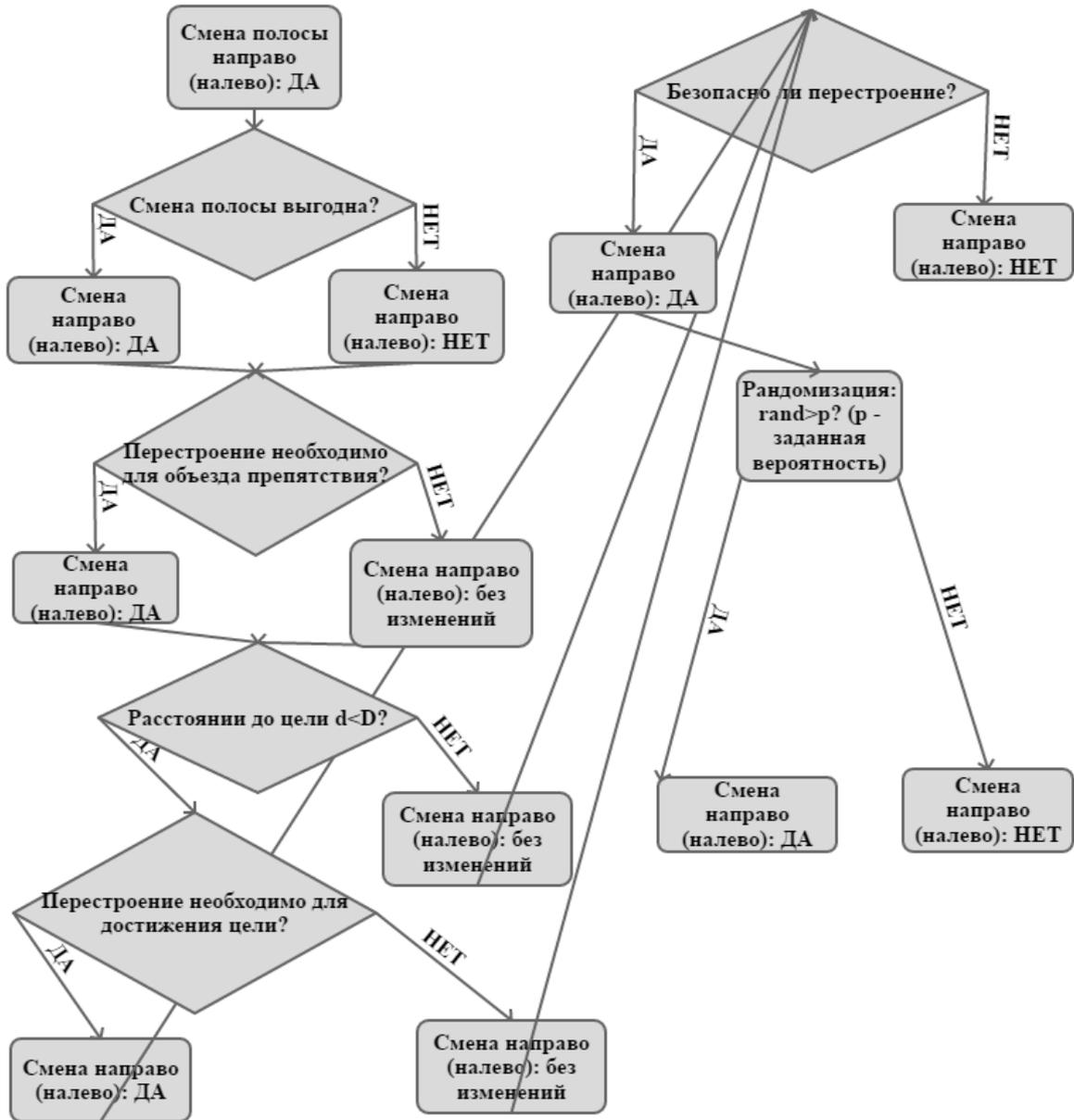


Рисунок 9 - Блок-схема алгоритма перестроения автомобилей при наличии цели

Подшаг (Б) – движение вдоль дороги - реализуется так же, как в общем алгоритме перестроения (алгоритм *a2*).

#### 2.4.2 Алгоритм *a4* прохождения четырехстороннего перекрестка со светофором

При прохождении регулируемого четырехстороннего перекрестка смена полосы происходит по алгоритму *a3* (перестроение с достижением цели). Его шаги 1-7 полностью повторяются.

Движение вперед происходит согласно алгоритму  $a2$ , однако при подъезде к перекрестку в него добавляются новые условия. Следующие правила применяются для того, чтобы автомобиль мог замедляться перед перекрестком и останавливаться на светофоре, а также поворачивать:

- 1) Замедление за  $D_2$  ячеек перед поворотом: если  $d_{i,k,j} < D_2$  то  $v_{i,k,j} = \min(v_{i,k,j} - 1, 1)$  ( $d_{i,k,j}$  – расстояние до перекрестка,  $D_2$  – фиксированное расстояние,  $v_{i,k,j}$  – скорость автомобиля,  $i$  – номер ячейки,  $k$  – номер полосы,  $j$  – номер дороги).
- 2) Остановка в точке поворота: если  $d_{i,k,j} = 0$  то  $v_{i,k,j} = 0$ .
- 3) Замедление и остановка на красный: если  $reg = R$  то при  $d_{i,k,j} > 0$   $v_{i,k,j} = \min(v_{i,k,j} - 1, 0)$ , при  $d_{i,k,j} = 0$  то  $v_{i,k,j} = 0$  ( $reg$  – режим светофора,  $R$  соответствует красному сигналу для данного направления)
- 4) Поворот/смена дороги  $j$ : если  $v_{i,k,j} = 0$  и  $reg \neq R$  и  $k = k^{goal}$  и  $v_{i^*,k^*,j^*} = -1$  (целевая ячейка свободна) то  $v_{i,k,j} \rightarrow v_{i^*,k^*,j^*}$  (здесь  $k^{goal}$  – целевая полоса, из которой разрешен поворот)
- 5) Движение прямо, если текущая полоса не предполагает поворот и целевая ячейка свободна: если  $v_{i,k,j} = 0$  и  $reg \neq R$  и  $k \neq k^{goal}$  и  $v_{i+1,k,j} = -1$  то  $v_{i,k,j} \rightarrow v_{i+1,k,j}$

Если цель машины поворот на перекрестке, и она находится в пределах  $D_2$  клеток от поворота, то ее скорость уменьшается, пока не достигнет 1. При достижении точки поворота машина останавливается.

Если горит красный сигнал светофора, автомобиль, подъезжающий к перекрестку, замедляется, в ячейке перед перекрестком он останавливается. Когда загорается зеленый, автомобиль снова начинает движение.

Если цель автомобиля – поворот на перекрестке (номер дороги, соответствующий цели, предполагает поворот) и он находится в полосе, из

которой данный поворот разрешен, он поворачивает. Это означает, что он попадает в соответствующую ячейку другой дороги, проверив, что она не занята.

### 2.4.3 Алгоритм а5 - обеспечение безопасности на перекрестке

Рассмотрим реализацию поворотов более подробно. Как уже упоминалось ранее, расчетная область всех элементов УДС состоит из четырех независимых участков. Для четырехстороннего перекрестка эти участки – направления движения (обозначены цифрами 0-3 на рисунке 7). Обозначения направлений следующие: движение снизу вверх –  $j=0$ , сверху вниз –  $j=1$ , справа налево –  $j=2$ , слева направо –  $j=3$ .

Каждое направление, или дорога, может иметь любое количество полос.

Подчеркнем, что термин «дорога» здесь соответствует именно одному направлению движения, в отличие от обычного употребления этого слова, когда дорогой называют оба направления (если движение не одностороннее). В обычном понимании, дорога – это совокупность направлений 0 и 1 или 2 и 3 на рисунке 10. Мы, однако, при упоминании номера дороги будем всегда иметь в виду номер направления.

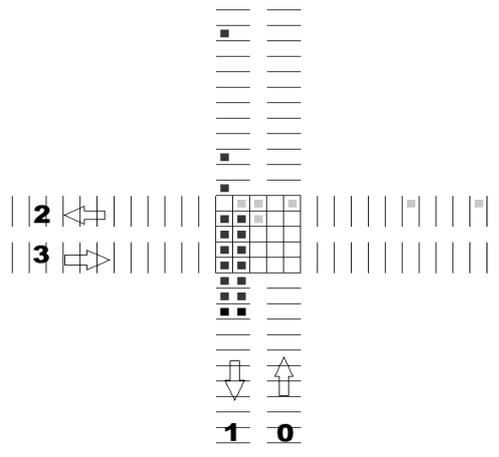


Рисунок 10 - схема движения на четырехстороннем перекрестке

Если моделируется, скажем, прямой участок УДС, то мы «выключаем» лишние направления (просто взяв входящие потоки по данным направлениям равными нулю) и взаимодействие этих направлений с актуальными.

При описании пересечений дорог различных типов необходимо, разумеется, описать взаимодействие пересекающихся направлений. В данном случае при расчете движения на четырехстороннем перекрестке у направлений 0-3, 0-2, 1-3, 1-2 имеются общие области. Они вместе с «разделительными полосами» в одну клетку, имеющимися между направлениями 0-1 и 2-3, образуют область перекрестка.

Для учета нахождения автомобилей на перекрестке создается матрица  $cross[x][y]=v_{j1} + v_{j2}$ , где  $v_{j1}$  и  $v_{j2}$  – значения скоростей в соответствующих ячейках по направлениям  $j1$  и  $j2$ ,  $x$  и  $y$  – переменные, нумерующие элементы матрицы перекрестка. Если ячейка расчетной области пустая, значение скорости в ней равно -1. Таким образом, элемент матрицы равен -2 тогда и только тогда, когда ячейка перекрестка пустая.

При подъезде автомобиля к границе перекрестка происходит обращение к соответствующему элементу матрицы  $cross[x][y]$  для проверки наличия автомобилей. Если целевая ячейка занята, автомобиль останавливается и ждет, когда она освободится. На следующем шаге он может попробовать перестроиться в другой ряд и повторить попытку проехать.

#### *2.4.4 Алгоритм аб прохождения T-образного перекрестка со светофором*

Этот алгоритм реализуется как частный случай алгоритма *a4*. Для этого используется входящая в общий алгоритм перестроения возможность объезда неподвижного препятствия. Ячейки расчетной области, занятые препятствием, обозначены при помощи параметра скорости, равного -10. «Лишние» ячейки расчетной области заполняются неподвижными препятствиями.

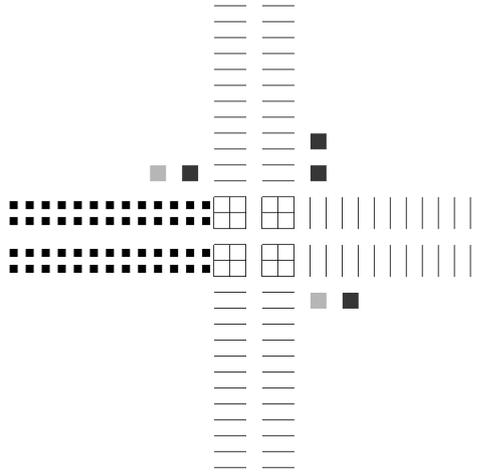


Рисунок 11 – Т-образный перекресток как вариант четырехстороннего

На рисунке 11 показано, как с помощью препятствий, обозначенных черными квадратами, происходит перекрытие незадействованных областей перекрестка. Теперь автомобили не могут попасть в запрещенные ячейки.

Для Т-образного перекрестка также адаптирован алгоритм достижения цели. Так как по направлению  $j=2$  (справа налево) нельзя проехать в прямом направлении, возможна ситуация, при которой автомобили, которым не удалось перестроиться в целевой ряд для поворота, а также те, для кого задана несуществующая цель  $j=2$ , могут «застрять» на перекрестке. Для того чтобы этого не происходило, все автомобили должны повернуть направо или налево, даже если не оказались в целевом ряду. Чтобы не нарушать ПДД, они поворачивают туда, куда разрешен поворот из ряда, в котором они оказались.

#### 2.4.5 Алгоритм а7 прохождения участка с сужением/расширением

Рассмотрим участок дороги с сужением и расширением, как показано на рисунке 12.



Рисунок 12 – участок УДС с сужением/расширением

При движении слева направо на дороге имеется расширение, а справа налево – сужение. На практике мы в данном случае имеем дело с участком, имеющем по две полосы в каждом направлении, но расчетная область в левой части рисунка, там, где дороги однополосные, ограничена при помощи заполнения ячеек неподвижными препятствиями, аналогично Т-образному перекрестку.

Проезд участка с сужением обеспечивается общим алгоритмом перестроения *a1*, который предполагает обнаружение препятствия в текущей полосе при приближении к нему (за 10 ячеек) и перестроение при первой возможности (*a1* п.3).

Проезд участка с расширением также не требует дополнительной адаптации алгоритмов. Перестроение в появившуюся полосу происходит за счет желания водителя двигаться по более свободному ряду (*a1* п.2).

#### 2.4.6 Алгоритм *a8* въезда с второстепенной дороги на главную

С точки зрения расчетной области, въезд на главную дорогу – вариант Т-образного перекрестка, лишние ячейки, на которые не должны заезжать автомобили, обозначаются при помощи параметра скорости  $v=-10$ .

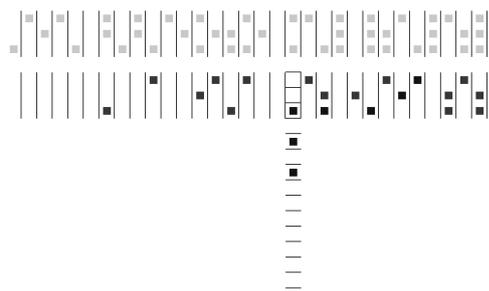


Рисунок 13 - участок УДС с въездом

На рисунке 13 представлен участок УДС, состоящий из трехполосной дороги с двусторонним движением и однополосного въезда. Широкая дорога является главной.

Алгоритм въезда, обеспечивающий приоритет автомобилям, движущимся по главной дороге, выглядит следующим образом.

- 1) Замедление за  $D_2$  ячеек перед поворотом: если  $d_{i,k,j} < D_2$  то  $v_{i,k,j} = \min(v_{i,k,j}-1, 1)$  ( $d_{i,k,j}$  – расстояние до перекрестка,  $D_2$  – фиксированное расстояние,  $v_{i,k,j}$  – скорость автомобиля,  $i$  – номер ячейки,  $k$  – номер полосы,  $j$  – номер дороги).
- 2) Задается переменная, обозначающая, состоится ли въезд  $we$  (want to enter), по умолчанию  $we=false$  (въезд невозможен).
- 3) Если расстояние до машины, подъезжающей к точке въезда по главной дороге  $d_{i,k,j}^{prev} > V^*$  то  $we=true$ .  $V^*$  может принимать разные значения в зависимости от стиля вождения,  $v_{i,k,j}^{prev} < V^* \leq V_{max}$ , где  $v_{i,k,j}^{prev}$  – скорость автомобиля на главной дороге,  $V_{max}$  – максимальная скорость.
- б) Поворот/смена дороги  $j$ : если  $we=true$  и целевая ячейка свободна ( $v_{i^*,k^*,j^*}=-1$ ) то  $v_{i,k,j} \rightarrow v_{i^*,k^*,j^*}$ .

#### 2.4.7 Алгоритм а9 проезда участка с разворотом

Участок с разворотом (рисунок 14) – это комбинация участка с выездом и участка с въездом.

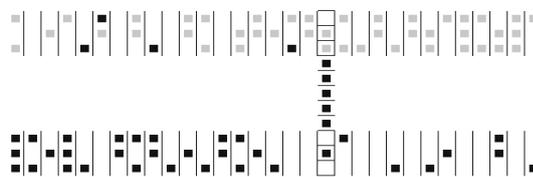


Рисунок 14 – участок с разворотом

Расчетная область, как всегда, реализована при помощи запрета движения на неиспользуемых участках четырехстороннего перекрестка. На рисунке 11 движение по нижней дороге происходит слева направо ( $j=3$ ), автомобили, въехавшие по данному направлению, обозначены темными квадратами. Часть автомобилей проезжает участок прямо, а часть едет на разворот.

Движение по верхней дороге происходит, соответственно, справа налево ( $j=2$ ) и автомобили, захватившие на этот участок справа, более светлые. На верхней дороге в левой части видно несколько темных автомобилей, которые попали на это направления с полосы для разворота.

Данный пример расчета выполнен для широкой разделительной полосы между направлениями 2 и 3 (для этого потребовалась небольшая дополнительная модификация расчетной области, помимо заполнения недоступных ячеек). Длина полосы для разворота может быть любой, в том числе и равной нулю.

С точки зрения алгоритмов, здесь используется комбинация  $a3$  и  $a8$  (выезда и въезда с второстепенной дороги на главную).

#### 2.4.8 Алгоритм $a10$ проезда участка с пешеходным светофором

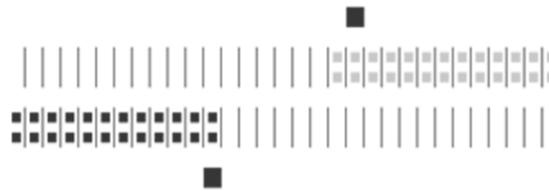


Рисунок 15 – участок с пешеходным светофором

Элемент с пешеходным светофором представлен на рисунке 15. Алгоритм проезда светофора – упрощенный вариант алгоритма  $a4$  без поворотов. Реализуются только части, отвечающие за светофорное регулирование.

#### 2.4.9 Алгоритм $a11$ объезда широкого неподвижного препятствия

Следующий алгоритм – усложненный вариант алгоритма перестроения  $a1$  для случая, когда на многополосной дороге имеется препятствие, занимающее более одной полосы (рисунок 16). Препятствие обозначено прямоугольниками с крестиками. Светлые и темный квадраты обозначают автомобили. Темный автомобиль должен принять решение, с какой стороны

объезжать препятствие – в данном случае, более выгодным является перестроение направо (в нижние полосы на данном рисунке).

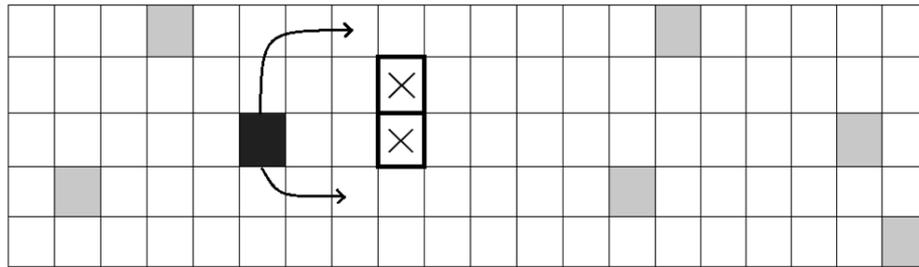


Рисунок 16 – участок с широким препятствием

Этот случай реализуется, например, при ремонте дорожного покрытия или при крупном ДТП. Водитель, подъезжающий к такому препятствию, должен принять решение, с какой стороны его объехать, выбрав более короткий путь. Для этого он проверяет, сколько ячеек поперек дороги занимает препятствие слева и справа от него. Если для объезда препятствия перестраиваться в данную сторону невыгодно и проще объехать его с другой стороны, смена полосы отменяется.

Формальная запись алгоритма перестроения для случая с широким препятствием выглядит следующим образом:

- 1) Проверка четности номера шага по времени: если  $step \% 2 = 0$   $wmr = true$ ,  $wml = false$ ; иначе  $wmr = false$ ,  $wml = true$ .
- 2) Выгодно ли перестроение: если  $d_{m,k+1} \leq d_{m,k}$  или  $v_{m,k+1} < v_{m,k}$  то  $wml(r) = false$
- 3) Есть ли препятствие на этой полосе:
  - если есть  $m$ :  $i < m \leq i+10$ ,  $v_{m,k} = -10$  то есть ли препятствие на целевой полосе? ( $v_{m,k+1} = -10$ ?)
    - если нет,  $wml(r) = true$
    - если да, находим значения  $NLl$  и  $NLr$  – число полос, занятых препятствием справа и слева от текущего

положения автомобиля. При  $NLl < NLr$   $wml = true$ ,  $wmr = wmr$ ;  
иначе  $wml = wml$ ,  $wmr = true$

○ если препятствия на этой полосе нет,  $wml(r) = wml(r)$ .

4) Свободна ли целевая ячейка: если  $v_{i,k\pm l} \neq -1$  то  $wml(r) = false$ .

5) Условие безопасности: если  $d_{m,k\pm l}^{prev} < V$  то  $wml(r) = false$ .

6) Стохастический элемент: если  $p < rand()$  то  $wml(r) = false$ .

Шаги 1-2 и 4-6 такие же, как в *al*, обозначения те же. Шаг 3 отличается.

Для верификации алгоритма была решена тестовая задача. На рисунке 17 показан некоторый момент времени выполнения численного эксперимента. Черные квадраты представляют широкое неподвижное препятствие, более светлые квадраты - автомобили. Движение происходит слева направо.

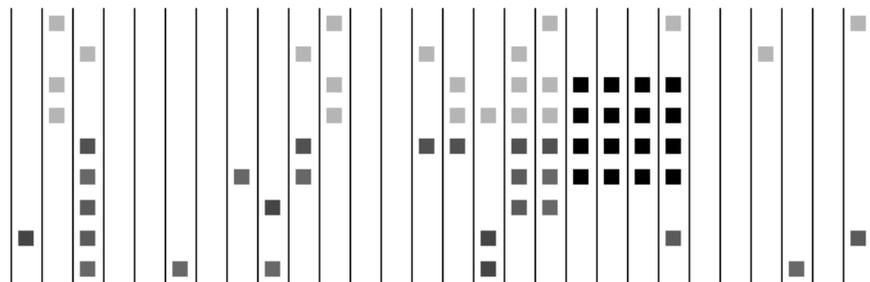


Рисунок 17 – участок с широким препятствием (результат моделирования)

Цель эксперимента – проверить, что автомобили выбирают нужное направление объезда. Для наглядности автомобили, которые заезжают в расчетную область по правым рядам и должны объезжать препятствие справа, имеют более темный цвет, а те, что заезжают по левым рядам и должны объезжать слева – более светлые. Выполнен ряд расчетов с разным расположением и шириной препятствия. Расчеты показывают, что водители верно выбирают направление объезда; это видно на рисунке 17. Можно сделать вывод, что алгоритм работает верно.

## 2.5 Различные стратегии поведения водителей

В модели реализованы две стратегии перестроения водителей: «агрессивные» и «осторожные». Эти качества проявляются в манере перестроения и при въезде с второстепенной дороги на главную.

- «Осторожные» водители для смены полосы или въезда выбирают большой интервал до следующего автотранспортного средства в целевой полосе/дороге. Расстояние до следующего автомобиля должно быть больше максимально разрешенной скорости в системе.

- «Агрессивные» водители оценивают текущую скорость следующего автомобиля, движущегося по целевой полосе/дороге, и перестраиваются вплотную, сохраняя лишь минимальный зазор, необходимый для обеспечения безопасности.

Реализация этих стратегий входит во все описанные выше алгоритмы перестроения и въезда:

- ✓ Условие безопасности: если  $d_{m,k\pm 1}^{prev} < V$  то  $wml(r)=false$  ( $d_{m,k\pm 1}^{prev}$  - расстояние до предыдущего автомобиля в целевой полосе,  $v_{i,k\pm 1}^{prev}$  - его скорость,  $V$  – величина, выбираемая исходя из стиля вождения данного водителя).

Если водитель «осторожный», то  $V=v_{max}$  или больше. Перестроение произойдет, только если число ячеек, отделяющее его от автомобиля в целевой полосе, больше разрешенной максимальной скорости в системе.

Если водитель «агрессивный», то  $V=v_{i,k\pm 1}^{prev}$ . Водитель сравнивает расстояние до автомобиля в целевой полосе с его скоростью, может перестроиться «вплотную» к подъезжающему по целевой полосе автомобилю.

Количество тех и других водителей в системе, а также условия на скорость можно варьировать.

Разумеется, при реализации модели «агрессивные» водители быстрее объезжают препятствия, и в результате вероятность образования затора меньше. Для верификации алгоритма были выполнены тестовые расчеты по моделированию объезда ДТП. На рисунках 18а, 18б представлены результаты этих расчетов.

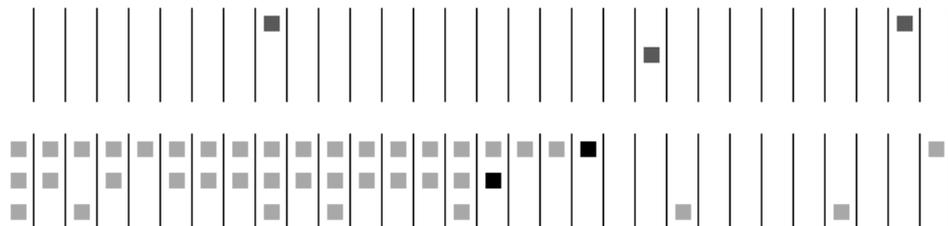


Рисунок 18(а) – объезд ДТП, «осторожные» водители

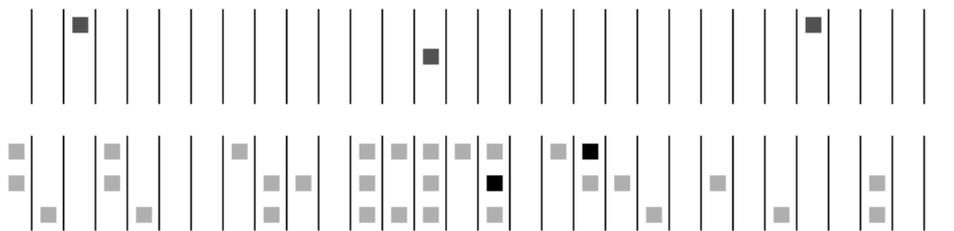


Рисунок 18(б) – объезд ДТП, «агрессивные» водители

На обоих рисунках изображен участок трехполосной дороги, в верхней части направление движения справа налево, в нижней части – слева направо. Квадратами обозначены автомобили. Черным цветом обозначены стоящие автомобили – участники ДТП. В расчете они играют роль неподвижного препятствия. Более светлые – движущиеся автомобили.

Расчет, представленный на рисунке 18(а), выполнен для ситуации, когда все водители являются «осторожными», то есть для перестроения ждут образования зазора на целевой полосе, равного, как минимум, максимально разрешенной скорости в системе. На рисунке 18(б) водители перестраиваются более смело, сравнивая имеющийся промежуток на целевой полосе с текущей скоростью следующего автомобиля. Как видно из рисунков, при «агрессивной» стратегии перестроения водителям удается

преодолеть возникшее препятствие более эффективно: плотность автомобилей перед препятствием во втором случае меньше, итоговая фактическая пропускная способность участка больше.

Часть водителей в системе могут, при определенных условиях, стать «вежливыми», то есть пропустить автомобиль из соседнего ряда, который хочет перестроиться, но не имеет такой возможности по соображениям безопасности. «Вежливым» может быть как «агрессивный», так и «осторожный» водитель.

Целью данного алгоритма является описание кооперации водителей для прохождения сужений или неожиданных препятствий на дорогах. Как известно, по правилам дорожного движения автомобили из полосы, где находится препятствие, могут перестроиться в соседнюю полосу для объезда только в том случае, если они не мешают водителям из свободной полосы. Однако, на практике, в ситуации плотного потока или затора это бы привело к тому, что такие автомобили были бы вынуждены ждать освобождения целевой полосы очень долго. На этот случай существует негласное правило проезда «елочкой»: по очереди, один автомобиль из полосы с препятствием – один автомобиль из полосы без препятствия. Конечно, не все водители поголовно применяют это правило. Недостаток водителей, готовых пропускать других добровольно, однако, восполняется наличием более агрессивных, которые «вклиниваются» в полосу, не оставляя выбора, кроме как остановиться и пропустить их. Оба варианта описываются созданным алгоритмом.

### *2.5.1 Алгоритм $a12$ «вежливый водитель» для смены полосы*

Алгоритм «вежливый водитель» (“cooperative driver”) состоит из двух частей. Первая часть – модифицированный алгоритм смены полосы при наличии в системе вежливых водителей ( $a12.1$ , рисунок 19). Вторая часть ( $a12.2$ ) – присвоение водителю статуса «вежливый».

Каждая часть, относящаяся к алгоритму  $a12$ , обозначена флагом  $cdonoff$  (cooperative driver on/off). Если  $cdonoff$  принимает значение true или 1, модули “cooperative driver” включаются в расчет и меняют общую схему принятия решения при смене полосы.

Вежливые водители в системе отмечены при помощи дополнительного параметра, описывающего состояние каждой ячейки -  $cd$ , который принимает значения да/нет (1 или 0). Напомним, что каждая ячейка имеет набор параметров, позволяющий сохранить всю необходимую информацию об автомобиле: его скорость, максимальная скорость, идентификационный номер ID, точка назначения (номер выезда), время въезда в расчетную область, а также ряд других вспомогательных параметров. Среди этих параметров есть индикаторы желания перестроиться направо/налево ( $lchr(l)$ , lane change right/left), которые не учитывают условия безопасности, то есть, они определяют не окончательное решение о перестроении, а лишь намерение поменять полосу. Эти параметры в модели выполняют ту же роль, что и «поворотники» в настоящем автомобиле. Другие водители могут их видеть и, при определенных обстоятельствах, принимать во внимание и замедляться или пропускать автомобиль, который хочет перестроиться, а могут и игнорировать.

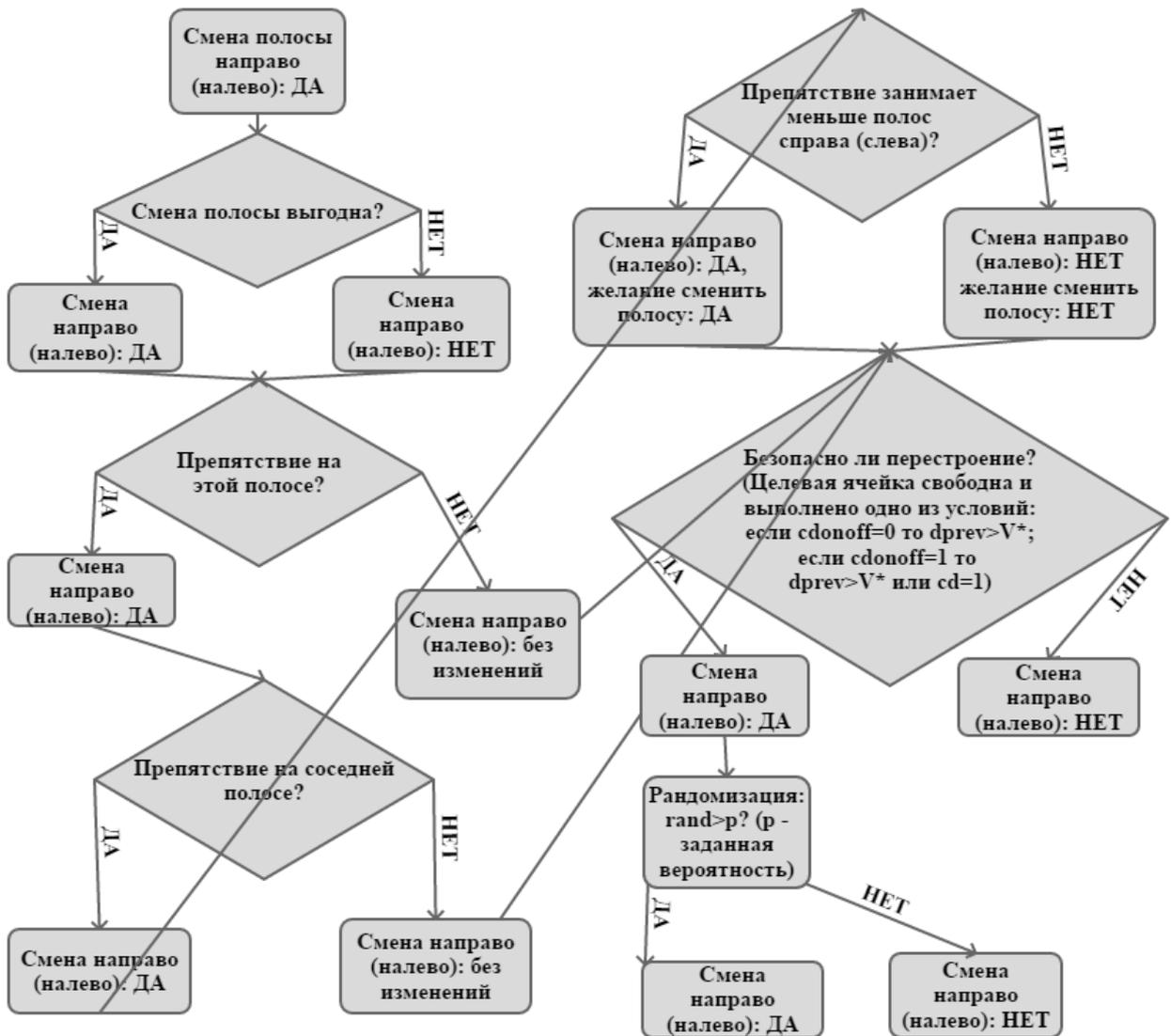


Рисунок 19 – Алгоритм *a12.1* перестроения при наличии «вежливых» водителей

Первая часть алгоритма, представленного на рисунке 16, уже была описана ранее (*a11*, пп.1-3). Отличие начинается после проверки, занимает ли препятствие меньше полос с той стороны, куда предполагается перестроение. В данном месте параметр  $wmr/wml$  сохраняется в переменную, обозначающую желание перестроиться. Если в результате указанных действий параметр имеет значение true, автомобиль «включает поворотник». Далее происходит проверка безопасности перестроения. Она выглядит по-разному для расчета с «вежливыми» водителями ( $cdonoff=1$ ) и без них ( $cdonoff=0$ ). Если в системе нет «вежливых» водителей, то проверяется условие  $d^{prev} > V^*$ , где  $d^{prev}$  – расстояние до предыдущего автомобиля в целевой

полосе, а  $V^*$  - скорость для сравнения, зависящая от особенностей стратегий водителей («агрессивных» и «осторожных»). При наличии в системе «вежливых» водителей ( $cdonoff=1$ ) условие безопасности  $d^{prev} > V^*$  уже может быть избыточно жестким, так как вежливые водители специально тормозят и останавливаются, чтобы пропустить других, и водитель с включенным «поворотником» может безопасно перестроиться вплотную перед стоящим автомобилем с «вежливым» водителем. Таким образом, совместно с условием безопасности проверяется, является ли водитель предыдущего автомобиля в целевой полосе «вежливым» ( $cd=1$ ) на данный момент. О том, как водители в системе становятся «вежливыми», будет изложено далее.

Соответственно, если условие безопасности не выполнено, смена полосы отменяется.

Следующим шагом в алгоритме является рандомизация. Водитель, желающий поменять полосу и имеющий такую возможность, тем не менее, может передумать перестраиваться с некоторой заданной вероятностью  $p$ . Затем, если водитель не передумал, происходит перестроение.

Далее рассмотрим, в каких условиях водителю присваивается статус «вежливый» (алгоритм *a12.2*). Это переменная величина, которая переопределяется каждый шаг по времени, однако можно задать определенное правило, по которому только часть водителей может стать вежливыми в процессе расчета. В нашем случае в качестве определяющего параметра служит идентификационный номер (ID), присваиваемый при добавлении автомобиля в систему. Половина водителей (с четными ID) могут стать «вежливыми», а другая половина – нет.

Присвоение параметра  $cd=1$  происходит непосредственно перед сменой полосы, после выполнения всех шагов рис. 1. Таким образом, использоваться этот параметр в алгоритме будет только на следующем шаге по времени. Для случая перестроения условия для того, чтобы водитель стал «вежливым»,

следующие:

- $step\%2=0$  для  $wmr$  ( $step\%2=1$  для  $wml$ )
- «Поворотник» автомобиля в полосе с препятствием включен:  $lchr(l)=true$ .
- Скорость автомобиля в полосе с препятствием  $v_{i,k\pm l}^{prev}=0$ . В полосе с препятствием уже образовался затор.
- Скорость данного автомобиля  $-1 < v_{i,k} \leq 1$ . Он либо уже остановился, либо его скорость достаточно мала для полной остановки.
- Номер данного автомобиля в системе  $ID\%2=0$  (или другое условие, определяющее соотношение «вежливых» водителей и всех остальных),
- $cd \neq -1$  ( $cd=-1$  после того, как водитель пропустил автомобиль из соседнего ряда и тот перестроился).

Последнее условие служит для того, чтобы один и тот же водитель, став вежливым, не пропускал сразу много машин из соседнего ряда, а сам начинал движение, пропустив только одну. Таким образом моделируется так называемое перестроение «елочкой» (zipper merge в англоязычной литературе), когда водители из свободного ряда пропускают по одному автомобилю из ряда с затрудненным движением. Практика показывает, что такой подход проезда участка с неподвижным препятствием наиболее эффективен.

Для верификации алгоритма рассмотрим задачу об объезде неподвижного препятствия. Для расчетов был выбран случай двухполосной дороги, одну полосу которой заблокировал неподвижный автомобиль (рисунок 20).

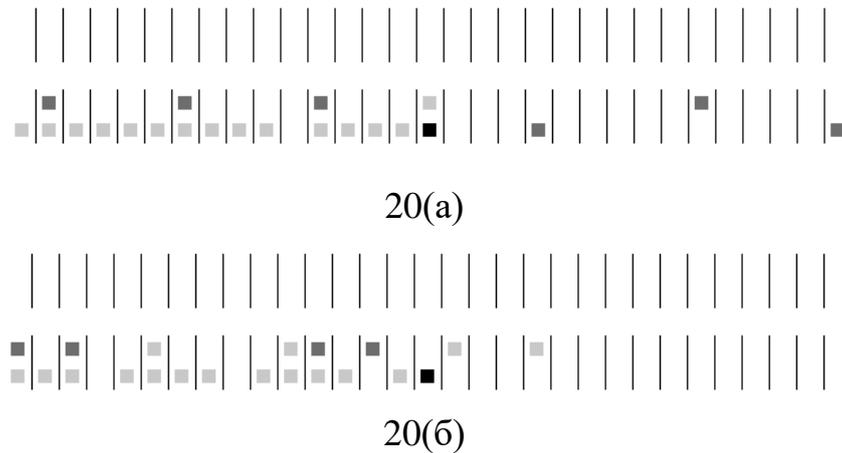


Рисунок 20 — Объезд препятствия – результат моделирования. Сверху (20(a)) – без «вежливых водителей», снизу (20(б)) – с «вежливыми водителями»

Здесь автомобили представлены квадратами, черный квадрат - неподвижное препятствие, светло-серые – автомобили, заехавшие в расчетную область по полосе с препятствием, темно-серые – те, кто въехал по свободной полосе. Движение происходит слева направо. При расчетах видно, что, в случае (а) - без «вежливых» водителей - движение в правой полосе, где находится препятствие, сильно затруднено, а в случае (б) - с «вежливыми» водителями - затруднено меньше. Однако, это не настолько очевидно, особенно из рисунков, поэтому была выполнена серия расчетов среднего времени нахождения автомобиля в системе при разных условиях. Для наглядности также отдельно анализировалось среднее время нахождения в системе автомобилей из полосы с препятствием. Расчеты показывают, что, в зависимости от выбранных параметров моделирования и значений потока, наличие «вежливых» водителей может снизить среднее время нахождения в системе автомобилей из полосы с препятствием в два раза и более.

### 2.5.2 Алгоритм *a13* «вежливый водитель» для въезда с второстепенной дороги на главную

На рисунке 21 представлена блок-схема алгоритма принятия решения о въезде с второстепенной дороги на главную (*a13.1*). Задается булева переменная *we* (want to enter), обозначающая, будет ли решение о въезде

положительным, изначально она имеет значение false. Далее следует проверка безопасности, которая, как и в блок-схеме *a12.1*, различается для случая с «вежливыми водителями» и без них.

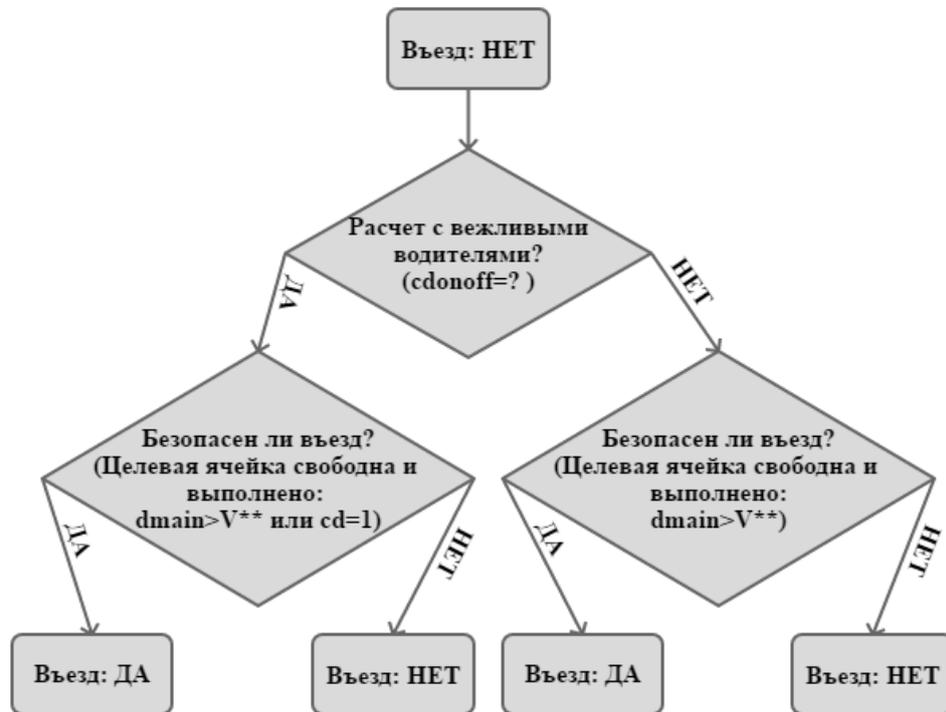


Рисунок 21 – Алгоритм *a13.1* въезда на главную дорогу с второстепенной при наличии «вежливых» водителей

В случае, когда  $cdonoff=1$ , должно выполняться одно из двух условий:  $dmain>V^{**}$  или  $cd=1$ . Здесь  $dmain$  – расстояние от ближайшего автомобиля, едущего по главной дороге, до точки въезда, а  $V^{**}$  – некоторая скорость, выбираемая в зависимости от желаемых особенностей водительских стратегий. Как и в случае с  $V^*$ , ее можно задать по-разному.  $cd=1$ , аналогично, обозначает, что водитель, движущийся по главной дороге, «вежливый» и готов остановиться, чтобы уступить дорогу въезжающему.

Соответственно, если  $cdonoff=0$ , то действует только условие  $dmain>V^{**}$ .

Рассмотрим условия задания  $cd=1$  для этого случая. Водители, движущиеся по главной дороге, могут пропускать тех, кто едет по второстепенной дороге, если там образовался затор. Водитель автомобиля А на главной дороге,

подъехавший к точке пересечения с второстепенной, становится «вежливым» ( $cd=1$ ) и пропускает автомобиль В, если выполнены все перечисленные условия (a13.2):

Для автомобиля В –

- $i^* = i^*_{entr}$ . Автомобиль В находится у въезда.
- $\langle v^*_{i,k} \rangle = 0$ . На второстепенной дороге, кроме В, есть еще несколько стоящих в очереди на въезд автомобилей.

Для автомобиля А –

- Скорость А  $-1 < v_{i,k} \leq 1$ . Он либо уже остановился, либо его скорость достаточно мала для полной остановки.
- $ID \% 2 = 0$  (или другое условие, определяющее соотношение «вежливых» водителей и всех остальных),
- $cd \neq -1$ . На этом шаге по времени водитель А еще не пропускал другие автомобили ( $cd = -1$  после того, как водитель пропустил автомобиль из соседнего ряда и тот перестроился).

Далее в расчетном модуле следует набор шагов из модели Нагеля-Шрекенберга (алгоритм a2) с дополнительными условиями для обеспечения работы алгоритмов «вежливый водитель». Так, на первом шаге, отвечающем за ускорение, при включенном режиме  $cdonoff=1$  проверяется условие: если водитель вежливый ( $cd=1$ ) и его скорость равна нулю, то данный автомобиль не ускоряется, его скорость остается равной нулю. На втором шаге, описывающем замедление, автомобиль с  $cd=1$  замедляется до остановки.

Как только «вежливый» водитель пропустил другого, ему присваивается  $cd=-1$ , что означает, что на данном шаге по времени он уже не может снова стать «вежливым».

Рассмотрим задачу о въезде на главную дорогу (рисунок 22) для верификации алгоритма a13. По главной дороге поток автомобилей равен  $Q1$ , по второстепенной дороге –  $Q2$ .

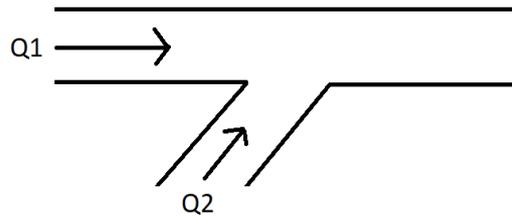


Рисунок 22 — Въезд на главную дорогу с второстепенной

В зависимости от выбора стратегии поведения водителей в модели, а именно, насколько большое расстояние до следующей машины на главной дороге им комфортно иметь при заезде, можно получить разные картины при расчетах. Если водители «агрессивные» и эффективно перестраиваются, при одних и тех же значениях потоков  $Q1$  и  $Q2$  получим снижение среднего времени нахождения автомобиля в расчетной области. Однако, при достаточно больших значениях потока  $Q1$  водителям с второстепенной дороги, независимо от их стратегии, заехать на главную тяжело. В результате на второстепенной дороге образуется затор, а среднее время нахождения автомобилей, попадающих в расчетную область с второстепенной дороги, в системе стремится к общему времени расчета. Однако в реальности, даже при интенсивном потоке по главной дороге водители находят способ заехать на нее и не стоят на въезде бесконечно.

При включении алгоритма «вежливый водитель» картина меняется. Результат моделирования представлен на рисунке 23. Квадратами обозначены автомобили; более светлые – те, что въехали по главной дороге, более темные – те, что въехали по второстепенной. Движение происходит слева направо.

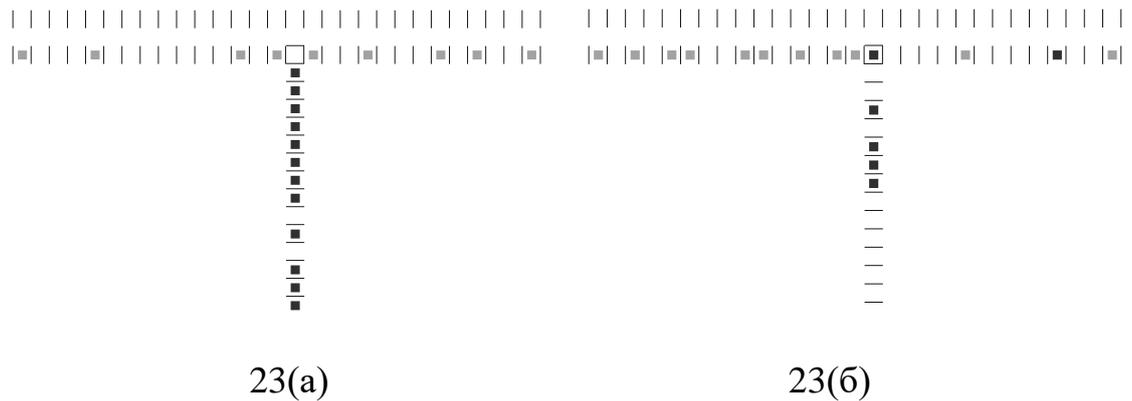


Рисунок 23 — Въезд на главную дорогу – результат моделирования.  
 Слева (23(a)) – без «вежливых водителей», справа (23(б)) – с «вежливыми водителями»

Как видно из рисунков, затор на въезде при включении алгоритма становится меньше. Также это подтверждает расчет среднего времени нахождения автомобилей с второстепенной дороги в системе.

Возникает вопрос о том, снижает ли наличие «вежливых» водителей общее среднее время нахождения автомобилей в системе (всех, а не только тех, кто должен уступать дорогу). Расчеты показывают, что не всегда – это зависит от выбора параметров моделирования, а конкретно, скоростей  $V^*$ ,  $V^{**}$ . Это означает, что данные параметры необходимо тщательно подбирать при калибровке модели под реальные условия при решении практических задач моделирования.

### 2.5.3 Алгоритм a14 снижения скорости при наличии затора

Для того чтобы объезд препятствий происходил эффективнее, при скоплении автомобилей в ряду с препятствием водители из других рядов реагируют на это, снижая скорость, упрощая тем самым перестроение. Такое поведение можно наблюдать на дороге: видя аварию или какой-либо неподвижный объект на дороге, водители замедляются из соображений безопасности.

Данный алгоритм встроен в алгоритм a2.

Задаются счетчики  $jcl$  и  $jcr$  (jam count left, jam count right), которые

изначально равны нулю. Для проверки необходимости снизить скорость выполняются следующие шаги.

- 1) Вычисление количества неподвижных автомобилей в соседних рядах справа и слева  $k \pm 1$ : для всех  $m$ :  $1+i \leq m < \min(i+10, size)$  если  $v_{m,k \pm 1,j} = 0$  то  $jcl(r) = jcl(r) + 1$  (здесь  $i$  – номер ячейки,  $k$  – номер ряда,  $j$  – номер дороги,  $size$  – длина расчетного участка в ячейках).
- 2) Вычисление расстояния до препятствия в соседних рядах (если оно есть): для всех  $m$ :  $1+i \leq m < \min(i+10+V_{max}, size)$  если  $v_{m,k \pm 1,j} = -10$  то расстояние до препятствия  $d_{obstl}(r) = m - i$ .
- 3) Если  $d_{obstl}(r) \leq 10$  и  $jcl(r) \geq 9$  то параметр, обозначающий наличие пробки перед препятствием  $jam = true$ .
- 4) На шаге 2  $a2$  дополнительное условие снижения скорости: если  $jam = true$ ,  $v_{i,k,j} = \min(v_{i,k,j} - 1, 0)$ .

## 2.6 Модель транспортных потоков с «медленным стартом»

Модели с «медленным стартом» (“slow-to-start”) – класс моделей, где автомобили, начинающие движение, стартуют только при условии, что перед ними имеется более одной свободной ячейки (см., например, [66]). На первый взгляд, это условие не следует очевидно и однозначно из наблюдения за поведением водителей. Оно было включено для того, чтобы автомобили не разъезжались слишком быстро от места затора, что позволяет, в свою очередь, воспроизвести эффект гистерезиса при переходе от свободного к синхронизированному потоку: при одной и той же величине потока движение может быть как свободным, так и стесненным, в зависимости от случайных процессов в транспортном потоке.

### 2.6.1 Алгоритм $a15$ «медленный старт»

Данный алгоритм также встроен в общий алгоритм движения вперед  $a2$ .

Шаг 1 в  $a2$  дополняется условиями на медленный старт и выглядит так:

- Если  $v_{ik} > 0$  то  $v_{ik} = \min(v_{ik} + 1, V_{max})$ ;
- Если  $v_{ik} = 0$  то ищем расстояние до ближайшего автомобиля впереди  $d$ .  
Если  $d > 1$ , то  $v_{ik} = v_{ik} + 1$ ;  
если  $d = 1$ , то  $v_{ik} = v_{ik}$

Остальные шаги *a16* повторяют шаги 2-4 *a2*.

Данный алгоритм также можно применять при калибровке модели для наилучшего совпадения данных тестовых расчетов с экспериментальными данными.

## **2.7 Граничные условия в модели**

Задание граничных условий при реализации модели требует отдельного внимания, поскольку, перед добавлением автомобилей в расчетную область следует убедиться в том, что начальные ячейки свободны и на участке нет затора.

### *2.7.1 Алгоритм *a16* задания граничных условий с возможностью образования очереди на въезд в расчетную область*

На рисунке 24 представлена блок-схема алгоритма задания граничных условий.



Рисунок 24 - Блок-схема алгоритма очереди на въезд в расчетную область

Поддержание необходимого значения входящего потока обеспечивается добавлением автомобилей каждый  $n$ -ый шаг по времени, где  $n$  вычисляется в зависимости от заданного потока (на данной дороге, с данной целью). При скоплении автомобилей в расчетной области может возникнуть ситуация, когда на нужном шаге по времени автомобиль не может быть добавлен в расчетную область. Для учета таких автомобилей служит набор переменных «очередь» ( $q_{j1j2}$ , где  $j1$  обозначает дорогу, по которой въезжает автомобиль, а  $j2$  - его целевую дорогу). Соответственно, если при въезде первая ячейка

оказалась занята, автомобиль попадает в очередь (соответствующее значение  $q_{j1j2}$  увеличивается на 1:  $q_{j1j2} = q_{j1j2} + 1$ ) На каждом шаге по времени проверяется наличие автомобилей в очереди ( $q_{j1j2} > 0$ ). Если  $q_{j1j2} > 0$ , автомобиль добавляется в расчетную область, а значение  $q_{j1j2}$  уменьшается на 1 ( $q_{j1j2} = q_{j1j2} - 1$ ).

## **Выводы по Главе 2**

Глава 2 посвящена построению математической модели транспортных потоков на основе теории клеточных автоматов CAM-2D и входящих в нее алгоритмов.

Выполнено обобщение модели Нагеля-Шрекенберга на двумерный случай. Созданы алгоритмы перестроения для различных случаев. Разработаны алгоритмы движения по различным участкам улично-дорожной сети: прямому участку дороги, четырехстороннему и Т-образному перекрестку, участку с сужением-расширением, участкам с въездом и выездом, участку с разворотом, участку с пешеходным светофором. Разработаны алгоритмы достижения цели, обеспечения безопасности на перекрестке, объезда широкого неподвижного препятствия, снижения скорости при наличии затора, задания граничных условий с возможностью образования очереди на въезд в расчетную область. Созданы алгоритмы поведения водителей: «агрессивный», «осторожный», «вежливый». Приведены тестовые задачи для верификации этих алгоритмов. Приведен алгоритм «медленный старт».

## ГЛАВА 3. ЧИСЛЕННАЯ РЕАЛИЗАЦИЯ. ПРОГРАММНЫЙ КОМПЛЕКС САМ-2D. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС И ВИЗУАЛИЗАЦИЯ. ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ. РАСЧЕТЫ НА СЕТИ

### 3.1 Структура последовательной программы

Для созданной модели и алгоритмов был разработан программный комплекс, реализующий их. Комплекс написан на языке C/C++ и использует библиотеку Glut для визуализации расчетов. Визуализация при помощи Glut происходит одновременно с расчетом и позволяет отлаживать программу и анализировать результаты. Пример такой визуализации представлен, например, на рисунке 5.

Программный комплекс имеет модульную структуру, каждый модуль отвечает за расчет одного элемента УДС, реализуя соответствующие алгоритмы, представленные в Главе 2:

- прямолинейный участок;
- перекресток (включая различные типы и настройки светофоров);
- участок с расширением/сужением;
- въезд/выезд с автомагистрали;
- участок с разворотом;
- участок с пешеходным светофором;
- и т. д.

### 3.2 Основные элементы и структуры

Ячейки, составляющие дорогу, в коде описываются структурой road traffic:

```
struct road {
    int v;
```

```

int d;
int vmax;
long int id;
int cd;

int color;
bool lchl;
bool lchr;
bool changeable;

};
road traffic[4][NL][size+1];

```

Структура представляет собой набор трехмерных массивов данных  $traffic[j][k][i]$ , измерения которого – номер дороги  $j$ , номер полосы  $k$  и номер ячейки  $i$ . Число дорог на элементе – 4 (или меньше), число полос – произвольное целое  $NL$ , максимальное число ячеек – произвольное целое  $size+1$ . Для универсальности число ячеек для элемента в сети равно 31, что соответствует 232,5 м. Для отдельных задач при необходимости  $size$  можно задать любым. Элементы структуры – параметры автомобиля: скорость  $v$ , цель (в рамках одного элемента УДС)  $d$ , максимальная скорость  $V_{max}$ , идентификационный номер  $ID$ , Булевы переменные, обозначающие желание перестроиться налево и направо  $lchl$  и  $lchr$ , параметр, определяющий, является ли водитель автомобиля «вежливым»  $cd$ .

Каждая «дорога» с номером  $j$ , где  $0 \leq j \leq 3$ , соответствует одному направлению движения на перекрестке (см. Рис. 25, направление указано стрелками и пронумеровано).

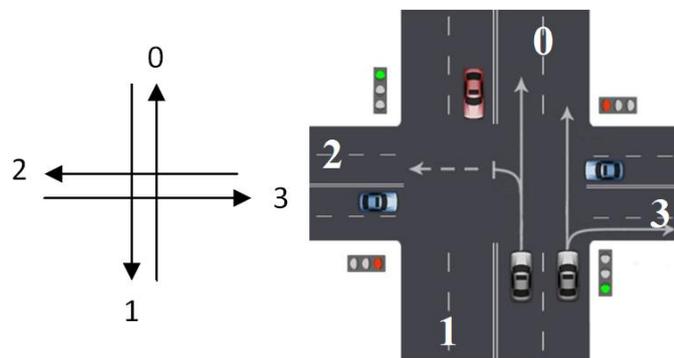


Рисунок 25 – Обозначения номеров дорог на примере четырехстороннего перекрестка.

Максимальное число направлений движения (дорог) в модели – 4. Дороги могут иметь любое число полос  $k$ . Нумерация ячеек  $i$  начинается от начала (въезда) на дорогу.

Все элементы разных типов в модели имеют указанную структуру. «Лишние» направления просто не используются, а те участки, на которые автомобиль физически не может заехать (например, сужение), отличаются значением параметра скорости ( $v=-10$ ). В процессе вычислений при перемещении автомобилей по сети происходит проверка параметра скорости, если он равен -10, автомобиль считает эту ячейку недоступной и объезжает ее или останавливается, если объезд невозможен.

Первый массив данных `traffic[j][k][i].v` содержит значения скоростей автомобилей во всех ячейках. В ячейках, занятых автомобилями, этот параметр принимает значения от 0 до 4. Если же ячейка пустая, скорость определяется равной -1; препятствие соответствует значению скорости -10.

Второй массив `traffic[j][k][i].d` содержит значения цели (точки назначения). Цель определяется в рамках одного элемента УДС и является, по сути, номером дороги, с которой должен произойти выезд автомобиля за рамки данного элемента. Данный параметр также принимает натуральные значения от 0 до 3. При переходе на соседний элемент цель автомобиля переопределяется. Данный параметр также отвечает за цвет автомобиля при визуализации; автомобили с разными целями имеют различный цвет при визуализации расчета.

Массив `traffic[j][k][i].vmax` содержит максимальные значения скоростей автомобилей, массив `traffic[j][k][i].id` – уникальные номера (ID) автомобилей, `traffic[j][k][i].cd` – параметр, определяющий, является ли данный автомобиль (водитель) «вежливым». Также в структуру входит ряд других вспомогательных параметров (*color*, *lchr*, *lchl*, *changeable*), служащих для моделирования конкретных ситуаций. Так, параметр *changeable* определяет

возможность перестроения в данной ячейке, то есть, по сути, определяет, является ли в данном месте разметка сплошной или прерывистой. Параметр `color` задает цвет автомобиля (отличный от четырех цветов, определяющих цель) и служит для отладки программы, обеспечивая более наглядное представление перестроений. Параметры `lchr` и `lchl` служат для описания логики взаимодействия «вежливых» водителей с другими участниками движения: эти параметры показывают желание водителей перестроиться (аналог включенного поворотника) в ситуациях, когда их полоса занята неподвижным препятствием.

Запуск последовательной версии программы для расчета на одном участке улично-дорожной сети осуществляется при помощи bat-файла, содержащего параметры моделирования:

```
REM Runner for cross simulation

REM 1st argument: debug mode (0 - off, other - on)

REM 2nd argument: cdonoff (0 - off, other - on)

REM arguments 3-6 Length of regime in seconds

REM arguments 7-9 Influx vehicles (veh/min) on road 0 with goals 0, 2, 3 (3)

REM arguments 10-12 Influx vehicles (veh/min) on road 1 with goals 1, 2, 3
(2)

REM arguments 13-15 Influx vehicles (veh/min) on road 2 with goals 0, 1, 2
(3)

REM arguments 16-18 Influx vehicles (veh/min) on road 3 with goals 0, 1, 3
(0)

REM arguments 19 p

REM arguments 20 vmax

REM arguments 21 maxstep

REM returns total number of cars passed through crossing

REM controls: x - stop run, space - next step, z - start autorun. esc - exit
```

```
Debug\Obstacle.exe 1 0 45 45 90 90 0 0 0 0 0 0 0 0 20 0 0 0 0.2 4 3600
```

pause.

Первый параметр позволяет производить расчет и визуализацию по шагам, переход к следующему шагу происходит при нажатии пробела. Второй параметр – наличие вежливых водителей *cdonoff* (эту функцию можно отключать и включать по необходимости). Затем задаются длины фаз светофоров (в секундах) на перекрестке (при их наличии) и входящие потоки по каждой из дорог и для каждого возможного значения параметра цели  $\text{flux}_{j_1-j_2}$ :  $\{\text{flux}_{0\_0} \dots \text{flux}_{3\_3}\}$  (итого 12 значений – 4 дороги на въезд и 3 на выезд, поскольку автомобиль не может выехать с того направления, с которого он заехал). Входящие потоки задаются в автомобилях в минуту. Далее задается вероятность  $p$  для ускорения и перестроения, максимальная скорость для данного расчета, общее расчетное время (в шагах/секундах). Все значения могут быть заданы вручную непосредственно в bat-файле либо в программе, или могут быть переданы в bat-файл из web-интерфейса взаимодействия программы и пользователя, созданного в сотрудничестве с коллегами из ИПМ им. М. В. Келдыша РАН (см. раздел «Интеграция с модулями Визуализации и Пользовательского интерфейса» данной главы).

Программа состоит из отдельных подпрограмм, обеспечивающих все необходимые функции. Сначала в отдельной подпрограмме при помощи библиотеки Glut создается изображение дороги и автомобилей для отслеживания их перемещений в процессе расчета. Затем задаются (или считываются) начальные значения элементов структуры – происходит заполнение ячеек расчетной области. Как правило, в начале расчета все ячейки пустые. Далее задаются граничные условия. Опять же, их можно задать напрямую в программе, можно считать из bat-файла, а при параллельных расчетах – получить от соседнего процессора в виде

сообщения МРІ. Тут же происходит проверка наличия автомобилей в очереди на въезд.

В bat-файле, как сказано выше, входящие потоки задаются в автомобилях в минуту для каждой дороги и для каждой конечной цели (оставшиеся 3 дороги) в отдельности. Считывая значения потоков, программа определяет, на каких шагах по времени  $n_{j_1_j_2}$  нужно добавлять те или иные автомобили (например, при  $Q=20$  авт/мин автомобиль добавляется каждый 3-й шаг по времени) – реализуется алгоритм *al6*.

Это происходит следующим образом. Значения  $flux_{j_1_j_2}$  в автомобилях в час переводятся в авт/сек:

```
fsj1_j20=fluxj1_j2/60;
```

Расчет значений шагов для добавления автомобилей  $n_{j_1_j_2}$  выглядит так:

```
if ((fsj1_j2>0)&&(fsj1_j2<=1)) nj1_j2=1/fsj1_j2;
if (fsj1_j2==0) nj1_j2=maxstep+1;
if ((fsj1_j2>1)&&(fsj1_j2<=NL)) {
    nj1_j2=1;
}
if (fsj1_j2>NL){
    nj1_j2=1;
    fsj1_j2=NL;}

```

Здесь  $NL$  – число полос,  $maxstep$  – время расчета. Переменные  $n_{j_1_j_2}$  принимают целые значения, соответственно значения округляются до целого. Если входящий поток по данному направлению равен нулю, шаг по времени, на котором должны добавляться такие автомобили, не реализуется в расчете (шаг равен  $maxstep+1$ ). Если значение потока больше, чем 1 (авт x полосу)/сек, то есть, больше абсолютной максимальной пропускной способности въезда за шаг по времени, то это значение приравнивается  $NL$ .

Далее, при наступлении шага, кратного данному, автомобиль добавляется в систему. Добавление происходит на нулевую ячейку ( $i=0$ ) и, как правило, на правую полосу ( $k=0$ ). Если нулевая ячейка на правой полосе занята, проверяется первая полоса. Если она свободна, автомобиль добавляется, если

нет – проверяется следующая и так далее до  $k=NL$ . Если свободных мест на въезде не обнаружено, автомобиль добавляется в очередь на въезд.

В программе очередь на въезд реализована в виде 12 счетчиков, соответствующих 12 возможным парам «номер дороги въезда + номер дороги цели», как для входящих потоков. При добавлении автомобиля в очередь соответствующий счетчик увеличивается на 1 (или больше, если за один шаг добавляется несколько автомобилей на соседние полосы). Остальные параметры являются стандартными (так, например, все автомобили въезжают с одинаковой скоростью), поэтому хранить эти параметры необходимости нет. На каждом шаге проверяется значение этих счетчиков, и, если какой-то из них отличен от нуля, происходит попытка добавить автомобиль с соответствующей целью на соответствующую дорогу. Если попытка оказывается успешной, значение счетчика уменьшается на 1.

При добавлении машины в систему ей присваивается уникальный номер.

Пример кода, добавляющего автомобиль на дорогу  $j_1$  с целью  $j_2$ :

```
if(step/nj1_j2>=1.0 && (step%nj1_j2)==0)
{
    if(fs_j1_j2>=1){
        f=fs_j1_j2;}
    else f=1;
    a=0;
    for(k=0; k<NL; k++){
if (traffic[1][k][0].v==-1){
    traffic[1][k][0].d=3;
    traffic[1][k][0].v=v0;
    traffic[1][k][0].vmax=vmax;
    traffic[1][k][0].id=1000+Ncars+1;
    Ncars=Ncars+1;
    a=a+1;
    if (a==f) k=NL;}
        else qj1_j2=qj1_j2+f-a;
    }};
```

Параметры  $a$  и  $f$  служат для случая, когда на одном шаге нужно добавить сразу несколько автомобилей на одну и ту же дорогу на разные полосы.  $qj_1_j_2$  – параметр очереди.

Далее следует функция `count` – основная функция, реализующая все модельные алгоритмы *a1 - a15*.

На каждом шаге по времени происходит запись в файл следующих параметров в определенном формате: номер текущего шага, параметры работы светофора (если есть) на данном шаге (подробнее о формате - ниже), затем – для всех автомобилей в системе - ID автомобиля, номер дороги, номер полосы, номер ячейки, цель.

```
out<<traffic[j][k][i].id<<"_"<<j<<"_"<<k<<"_"<<i<<"_"<<traffic[j][k][i].d <<"|";
```

Таким образом, в результате имеем всю информацию о системе на каждом шаге по времени.

Дополнительно могут рассчитываться усредненные характеристики движения, пропускная способность участка и т. д.

### 3.3 Параллельная реализация

Так как логические операции плохо поддаются распараллеливанию на графических ускорителях (GPU), был выбран вариант реализации параллельной версии на CPU. Параллельная версия программного комплекса SAM-2D написана на языке C/C++. Отладка и тестовые расчеты выполнены на суперкомпьютере К-100 ЦКП ИПМ им. М.В. Келдыша РАН.

Созданной модели транспортных потоков присущ как геометрический параллелизм, так и параллелизм данных. Для реализации был выбран первый вариант. Расчетная область – часть городской улично-дорожной сети – разделяется на отдельные элементы: перекрестки, прямые участки дороги, участки с въездами и съездами и т.д. Границы проходят по участкам без особенностей, обеспечивая простоту передачи данных.

Из модулей, как из конструктора, можно собирать любые сети, соответствующие реальной карте города.

Для того чтобы программный комплекс был универсальным, модули, отвечающие за моделирование на отдельных участках УДС, стандартизированы по размеру. Каждый модуль рассчитывается отдельно на своем процессоре, на границах происходит обмен данными при помощи библиотеки MPI. Автомобиль, выезжающий с одного элемента, появляется на другом, в месте, соответствующем его скорости на момент перехода.

В отличие от многих других задач, при моделировании с помощью клеточных автоматов недостаточно просто передать данные с одного процессора на другой. Перед передачей нужно убедиться, что автомобили, которые покидают один элемент, действительно могут попасть на следующий: их целевая ячейка, соответствующая текущей скорости, также как и все предшествующие ячейки, должна быть свободна. Рассмотрим, как происходит взаимодействие между процессами при въезде и выезде автомобилей с расчетного элемента.

### 3.3.1 Алгоритмы $a17$ и $a18$ выезда и въезда автомобиля на расчетный элемент

Алгоритм  $a17$  выезда автомобилей с расчетных элементов представлен на рисунке 26, а добавления – на рисунке 27.

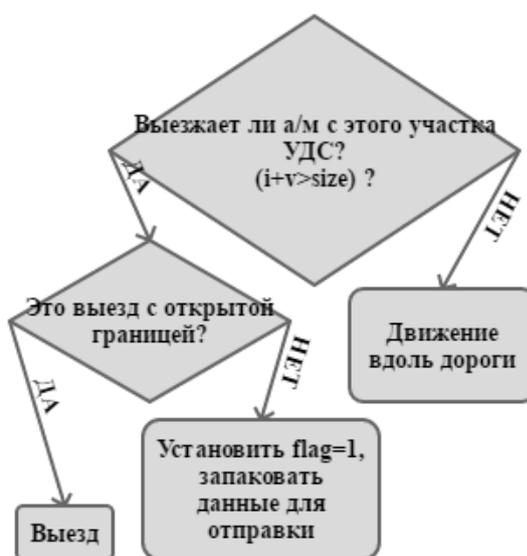
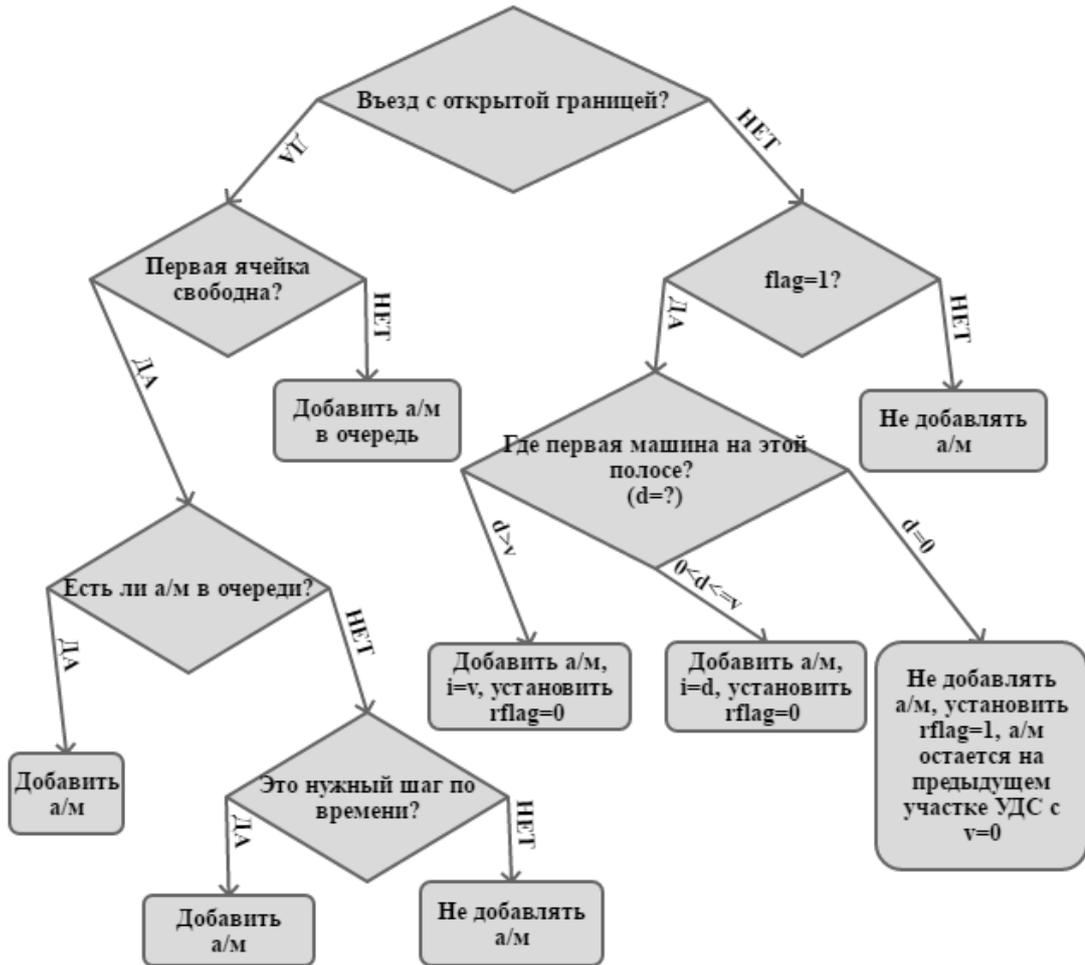


Рисунок 26 – Алгоритм *a17* выезда автомобиля с расчетного элементаРисунок 27 – Алгоритм *a18* добавления автомобиля на расчетный элемент

Рассмотрим сначала выезд автомобиля. Если значение номера ячейки после выполнения очередного шага превысило размер расчетной области, то проверяем, является ли этот выезд открытым или это граница со следующим расчетным элементом.

Если выезд открытый, то никаких дополнительных действий не требуется, автомобиль выезжает из расчетной области.

Если это внутренняя граница, устанавливаем значение переменной  $flag=1$  для процессора нужного ранга (то есть того, где рассчитывается следующий элемент). По умолчанию  $flag=0$ . Это означает, что на данном шаге есть

автомобиль (автомобили), готовый въехать на следующий элемент. Значение *flag* передается следующему процессору.

Дальше переходим к въезду. Сначала выполняется проверка, является ли данный элемент внешней границей расчетной области. Если да, то добавлять автомобили нужно исходя из заданных граничных значений входящих потоков, пользуясь алгоритмом *a15*. Если нет, проверяем значение *flag* для данного процесса.

При *flag=0* добавлять автомобили не нужно.

При *flag=1* определяем, свободна ли целевая ячейка (и ячейки до нее).

Если да, получаем данные об автомобиле от предыдущего элемента и автомобиль добавляется на текущий расчетный элемент на целевую ячейку. Если скорость автомобиля больше, чем расстояние до первого автомобиля, его скорость уменьшается, и он попадает на самую дальнюю от въезда свободную ячейку. Устанавливается значение переменной *rflag=0* и ее значение возвращается на предыдущий процессор. Это означает, что автомобиль был благополучно добавлен и его можно убрать с предыдущего расчетного элемента.

Если добавить автомобиль некуда, устанавливается *rflag=1* и возвращается предыдущему процессору. Автомобиль остается на предыдущем элементе в последней ячейке *i=size*. Он попадет на следующий расчетный элемент, как только там освободится ячейка.

### 3.4 Интеграция с модулями Визуализации и Пользовательского интерфейса

#### 3.4.1 Модуль пользовательского интерфейса и визуализации данных

Для программного комплекса САМ-2D совместно с коллегами из ИПИМ им. М.В. Келдыша РАН А.В. Ермаковым и М.С. Германом был разработан модуль пользовательского интерфейса и визуализации результатов [70].

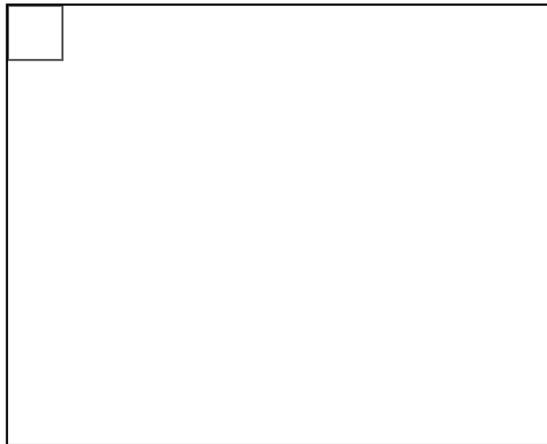
Вид веб-интерфейса представлен на рисунках 25-27.

Интерфейс позволяет пользователю задавать размеры расчетной области и входящие потоки на перекрестке, получать данные вычислений с сайта ws-dss.com – платформы поддержки принятия решений, куда интегрирован один из модулей САМ-2D, вычисляющий пропускную способность перекрестка (рисунок 25). Платформа ws-dss была создана в рамках работы **«Разработка архитектуры, принципов построения и программного обеспечения суперкомпьютерной информационно-аналитической платформы прикладного моделирования, прогнозирования и экспертиз транспортных процессов и систем в виде сетевой компьютерной лаборатории»** в рамках ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2014-2020 годы», Соглашение о предоставлении субсидии от 30.06.2014 г. № 14.604.21.0052.

### Схема перекрестка

Можно указать параметры вручную или получить из ws-dss.com

Получить из ws-dss



X: 500 Y: 400 клеток X: 10

- Обновить
- Загрузить:  Файл не выбран
- Сохранить

**Задание полос в каждом из направлений**

Из центра на север:

С севера в центр:

Из центра на запад:

С запада в центр:

С запада в центр:

Из центра на восток:

Из центра на юг:

С юга в центр:

Рисунок 28 – веб интерфейс. Задание потоков на перекрестке, интеграция с ws-dss.com

На рисунке 29 показано, как из отдельных элементов при помощи веб-интерфейса задается улично-дорожная сеть.

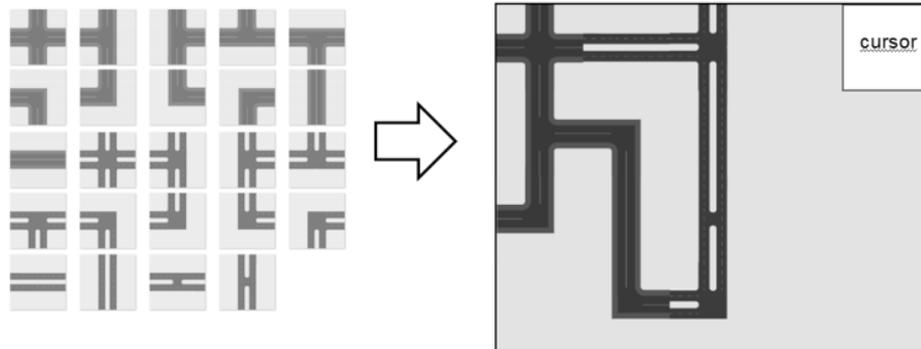


Рисунок 29 – веб интерфейс. Задание дорожной сети

## Вычисления

Рабочая папка:

Загрузить начальные данные для расчёта (светофоры и др.):  Файл не выбран 

<input type="button" value="Создать bat"/>	
<input type="button" value="Запуск bat"/>	
<input type="button" value="Есть ли результат?"/>	
<input type="button" value="Отправить ws-dss"/>	по параметрам ws-dss (см. выше)

## Отрисовка



Из файлов	Доп действия
Рабочая папка (*.txt): <input type="text" value="D:\temp\IMP"/>	
<input type="button" value="Загрузить в БД"/>	<input type="button" value="Очистить БД"/>
<input type="button" value="Загрузить из БД в RAM"/>	

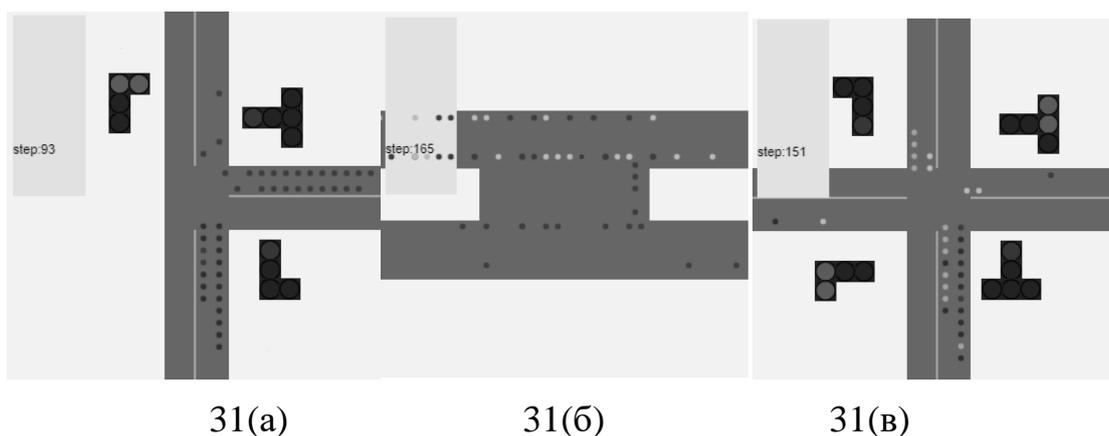
© 2021 - ИПМ РАН

Рисунок 30 – веб интерфейс. Создание bat-файла для запуска расчета, загрузка данных для визуализации полученных результатов

На рисунке 30 сверху представлена часть, создающая файл запуска для расчетного модуля САМ-2D, содержащий необходимые начальные данные и параметры, для отправки на сервер ws-dss или на расчетный модуль на персональном компьютере.

На рисунке 30 снизу представлено меню загрузки результатов расчета, выполненного при помощи САМ-2D, для визуализации полученных результатов.

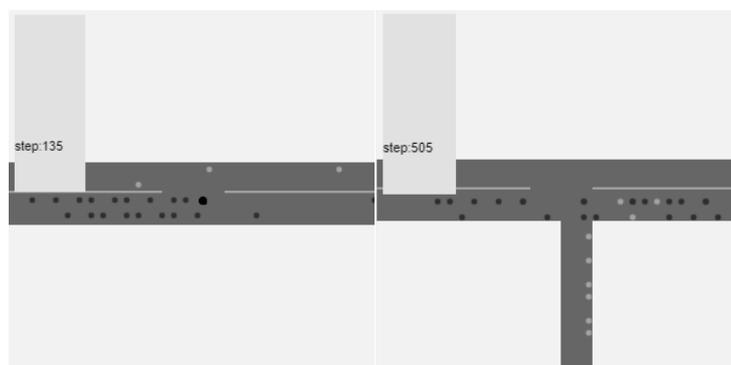
Примеры результатов этой визуализации расчетов на различных элементах УДС (31(а) - Т-образный перекресток, 31(б) - участок с разворотом, 31(в) - четырехсторонний перекресток, 31(г) - участок с неподвижным препятствием - ДТП, 31(д) - участок с въездом) представлены на рисунке 31.



31(a)

31(б)

31(в)



31(г)

31(д)

Рисунок 31 – Визуализация результатов расчетов на отдельных элементах УДС (модуль визуализации)

На рисунках автомобили представлены в виде точек. Различные оттенки серого отвечают за обозначение целей машин, как и в отладочной версии визуализации.

Видно, что модуль визуализации обеспечивает более качественную и реалистичную картинку по сравнению с отладочной визуализацией, содержащейся в расчетном модуле САМ-2D и написанной при помощи библиотеки Glut языка С (см. для сравнения рисунок 8). Помимо этого, в отличие от отладочной версии, модуль визуализации позволяет отображать результаты параллельных расчетов, выполненных на суперкомпьютере К100.

### 3.4.2 Интеграция расчетного модуля САМ-2D с модулем пользовательского интерфейса и визуализации

Для визуализации записывается отдельный текстовый файл. При параллельных вычислениях каждый модуль записывает свой файл. Файлы имеют определенный формат для обеспечения интеграции с модулем визуализации.

Модуль обращается к папке, расположение которой указывает пользователь (рисунок 27, меню «отрисовка»). Обработываются все файлы, лежащие в указанной папке. Файлы, формат которых не соответствует правилам, перечисленным ниже, отбрасывается.

В шапке каждого файла содержится информация, необходимая для распознавания и отрисовки элемента:

**0\_60\_90\_0|3\_2\_1\_0|31|108|0.2|3600|10\_0\_0\_6\_0\_0\_0\_0\_0\_0\_0\_0|0\_1|3**

Здесь:

1. «**0\_60\_90\_0**»: Длительность режима 0 светофора (сек)\_ Длительность режима 1 светофора (сек)\_ Длительность режима 2 светофора (сек)\_ Длительность режима 3 светофора (сек)
2. «**3\_2\_1\_0**»: тип светофора 0\_тип светофора 1\_тип светофора 2\_тип светофора 3
3. «**31**»: *size*, размер дороги (клетки)
4. «**108**»:  $V_{max}$ , максимальная скорость автомобилей (км/ч)
5. «**0.2**»:  $p$ , вероятность
6. «**3600**»: *maxstep*, длительность расчета (шаги/секунды)
7. «**10\_0\_0\_6\_0\_0\_0\_0\_0\_0\_0\_0**»: потоки по направлениям flux $j_1$ - $j_2$   
вх.поток0\_0\_вх.поток0\_2\_вх.поток0\_3\_вх.поток1\_1\_вх.поток1\_2\_вх.по

ток1\_3\_вх.поток2\_0\_вх.поток2\_1\_вх.поток2\_2\_вх.поток3\_0\_вх.поток3\_1\_вх.поток3\_3

- 8. «0\_1»: позиция этого перекрёстка относительно других в данном расчёте.  $\{x,y\}=\{0,1\}$  – в данном расчёте всего два перекрёстка, слева от данного есть еще один.
- 9. «3»: тип перекрёстка

Каждому типу перекрёстка и светофора присвоено обозначение. Задание типов перекрёстков представлено на рисунке 29. Номер рисунка соответствует типу. В нашем примере – Т-образный перекрёсток (3.jpg).

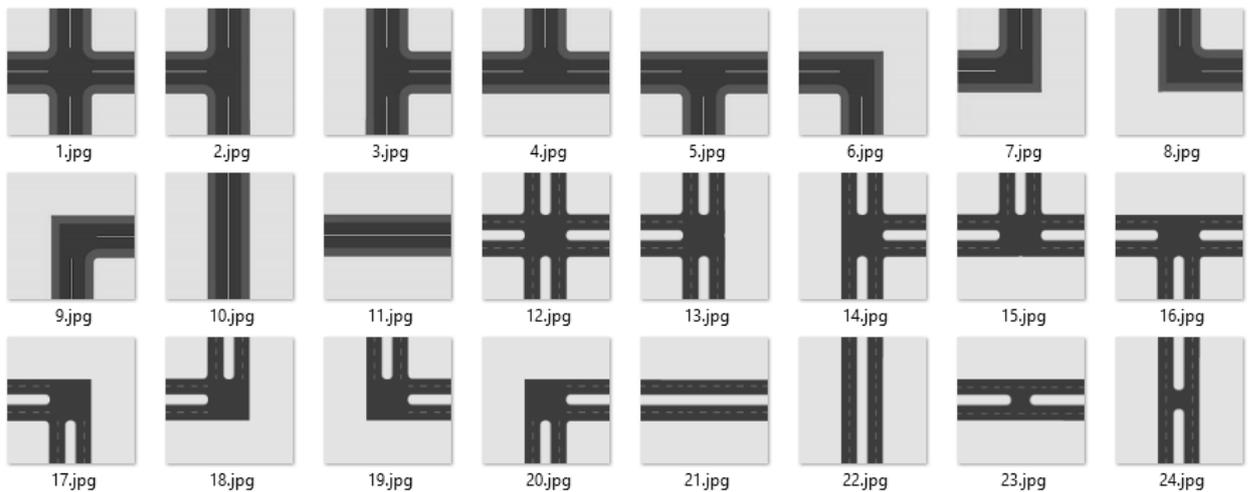


Рисунок 32 – задание типов перекрёстков

Задание типов светофоров выглядит следующим образом. Существует 5 типов светофоров (рисунок 30): с двумя стрелками направо и налево (обозначим его LSR), со стрелкой налево (LS), со стрелкой направо (SR), без стрелок (S).

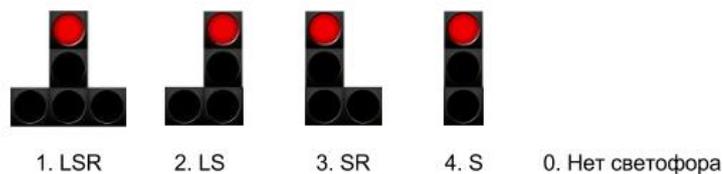


Рисунок 33 – типы светофоров

Каждый тип светофора имеет набор режимов работы, все режимы принимают значения от 1 до 16, нумерация сквозная. Поэтому режим работы однозначно соответствует определенному типу. Но нам требуется обозначить как типы, так и режимы, поскольку типы используются для первоначальной отрисовки расчетной области.

Нумерация режимов представлена на рисунках 31-34.

Тип светофора 0 (нет светофора) имеет режим работы 00.

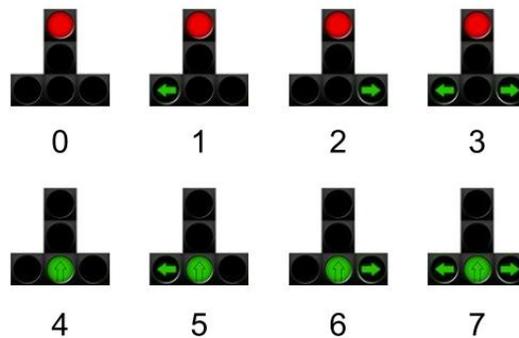


Рисунок 34 – режимы работы LSR

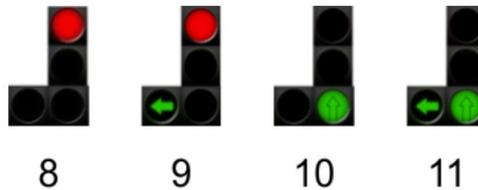


Рисунок 35 – режимы работы LS

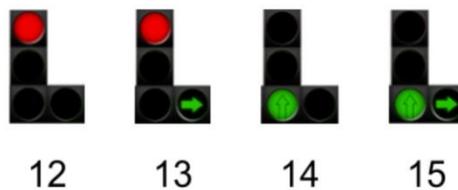


Рисунок 36 – режимы работы SR

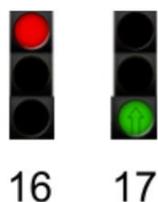


Рисунок 37 – режимы работы S

Прочитав шапку файла, модуль визуализации подготавливает соответствующий рисунок участка УДС с нужными светофорами или без них. После обработки шапки известен размер поля визуализации, общее количество перекрестков, их типы и другие характеристики.

Далее обрабатываются данные о движении транспорта и режимах светофоров, содержащиеся в строках от второй до последней. Обрабатываются по очереди по одной строке из каждого файла. Так как строки упорядочены по времени, можно делать промежуточную передачу данных для визуализации еще до достижения конца обработки файлов.

Каждая строка расчетных данных выглядит так:

**10\_14\_10\_0\_0:1001\_0\_0\_17\_0|1002\_1\_0\_2\_1**

Здесь «10\_14\_10\_0\_0»: Номер шага\_режим светофора 0\_режим светофора 1\_режим светофора 2\_режим светофора 3

«1001\_0\_0\_17\_0», «1002\_1\_0\_2\_1»: данные об автомобилях с идентификаторами (ID) 1001 и 1002 в формате

ID\_№дороги\_№полосы\_№ячейки\_цель\_маркер аварии(если нужно)(-10).  
Автомобилей в строке может быть любое количество.

Каждая строка содержит один шаг по времени. При считывании данных модуль визуализации рисует автомобили в соответствии с их положением и целями (цели определяются цветом/оттенками серого).

### 3.5 Расчеты на сети

Для проверки работы параллельной версии программы были выполнены тестовые расчеты на небольших сетях, состоящих из двух перекрестков. На рисунке 35 представлен расчет на сети, содержащей разворот и четырехсторонний перекресток. На рисунке 36 представлен расчет на сети, содержащей Т-образный и четырехсторонний перекрестки. Оба рисунка получены при помощи модуля визуализации.

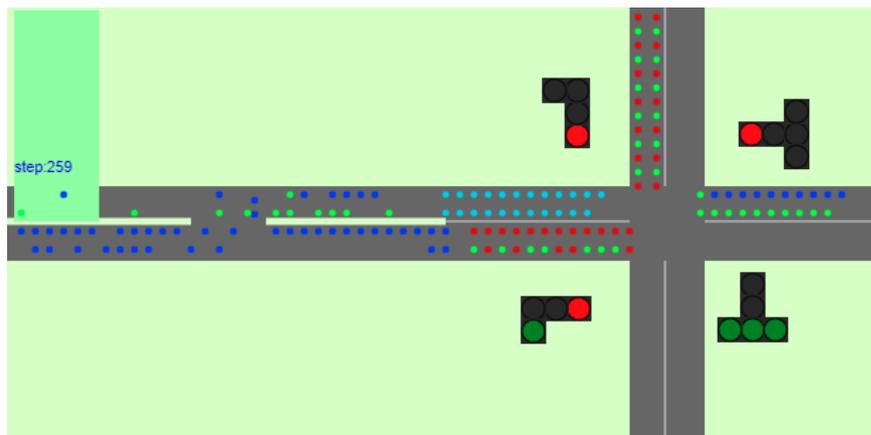


Рисунок 38 – результаты расчета. Разворот + четырехсторонний перекресток

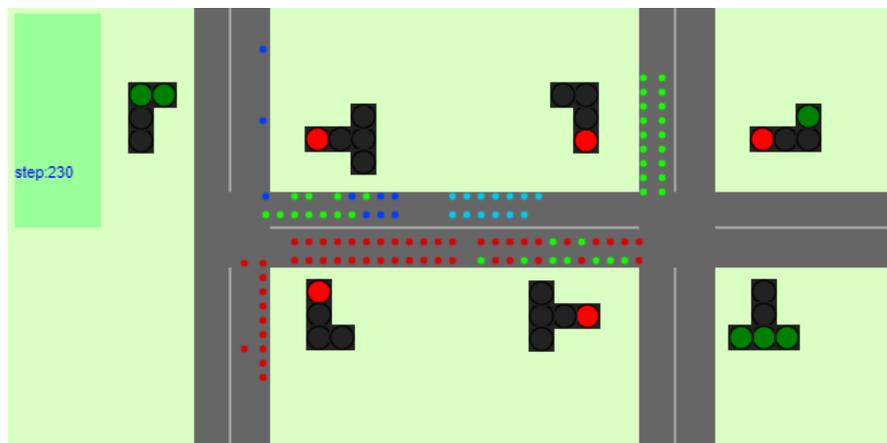


Рисунок 39 – результаты расчета. Т-образный + четырехсторонний перекрестки

Оттенки серого/цвета определяются целями автомобилей. Так как цель – это номер дороги  $j$ , который задается заново для каждого элемента УДС, при переходе от одного элемента к другому цели переопределяются.

Расчеты показывают, что расчетный модуль САМ-2D и модуль пользовательского интерфейса и визуализации интегрированы и работают верно. Помимо этого, параллельная версия САМ-2D также успешно прошла тестирование и работает верно.

### **Выводы по Главе 3**

Глава 3 посвящена численной реализации модели САМ-2D и созданию программного комплекса САМ-2D.

Представлена последовательная версия комплекса САМ-2D, входящие в него расчетные модули и основные элементы и структуры. Представлена параллельная версия программного комплекса, показаны особенности обмена граничными данными.

Представлен созданный в сотрудничестве с коллегами модуль пользовательского интерфейса и визуализации, позволяющий задавать входные параметры, начальные данные, а также обрабатывать и визуализировать результаты. Описана интеграция расчетной части САМ-2D с этим модулем.

Для апробации параллельной версии комплекса САМ-2D, а также проверки интеграции модулей, выполнены тестовые расчеты на небольших сетях с визуализацией результатов.

## ГЛАВА 4. ВЕРИФИКАЦИЯ МОДЕЛИ: СРАВНЕНИЕ С AIMSUN TSS, ВОСПРОИЗВЕДЕНИЕ ЭКСПЕРИМЕНТАЛЬНО НАБЛЮДАЕМЫХ ПРОСТРАНСТВЕННО-ВРЕМЕННЫХ СТРУКТУР

### 4.1 Сравнение с Aimsun TSS

Для верификации созданной модели было проведено сравнение с коммерческим пакетом для прикладного моделирования транспортных потоков Aimsun TSS, разработанным в Испании и успешно применяющимся для решения транспортных задач в Барселоне, других городах Испании и за ее пределами.

#### 4.1.1 Тестовая задача 1 о пропускной способности перекрестка

В качестве тестовой задачи 1 была выбрана задача о пропускной способности регулируемого четырехстороннего перекрёстка. Целью этой задачи является поиск оптимального светофорного режима, а именно определение длительности сигналов светофоров, обеспечивающей минимальное время прохождения перекрёстка всеми участниками движения.

Перекрёсток состоит из двух пересекающихся дорог, по которым автомобили движутся в двух направлениях и каждое направление имеет по две полосы. На рисунке 40 представлено состояние перекрёстка в некоторый момент времени. На полосах движения изображены автомобили, различающиеся по конечной цели. Каждому автомобилю предписывается своя цель – после прохождения перекрёстка оказаться на заданной дороге.

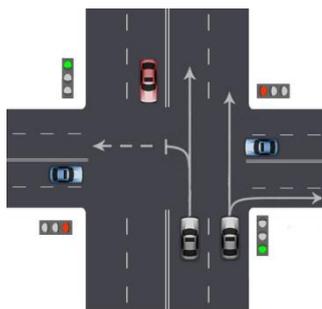


Рисунок 40 – Постановка задачи о движении на перекрёстке

На пересечении дорог стоят светофоры, которые пропускают машины в определенных направлениях в определенные моменты времени. Всего у светофоров существуют 4 фазы работы. Порядок их переключения и возможность пропуска в различных направлениях соответствуют реальному объекту г. Москвы.

По условиям задачи на двух направлениях – снизу и справа – поток машин значительно превышает поток машин на двух других направлениях

На рисунке 41 представлены результаты расчётов, проведенных при помощи модели CAM-2D и при помощи коммерческого программного обеспечения Aimsun TSS. При различных светофорных режимах (длительностях фаз светофоров) проводилось исследование пропускной способности перекрестка.

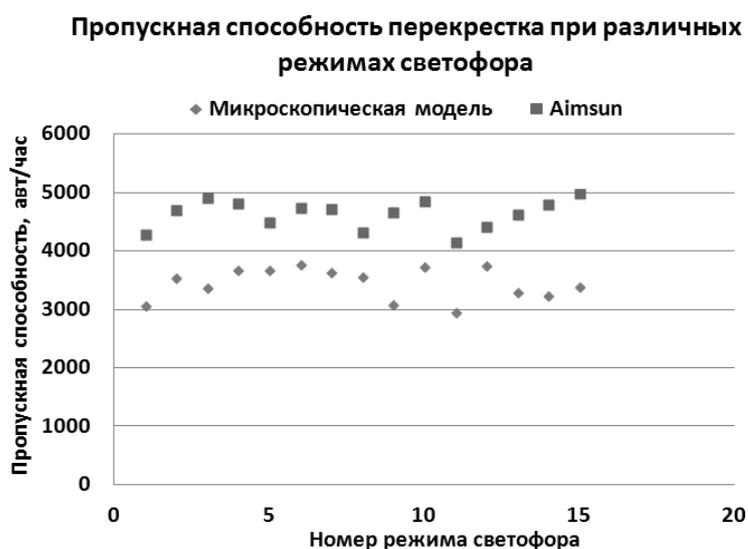


Рисунок 41 – Пропускная способность перекрестка при различных режимах работы светофора

По оси абсцисс отложены номера режимов светофоров, по оси ординат – пропускная способность перекрестка в автомобилях в час. Верхняя кривая (маркеры в виде квадратов) – расчет при помощи Aimsun, нижняя (маркеры в виде ромбов) – расчет при помощи CAM-2D.

Меняя длительность фаз светофоров, можно повысить пропускную способность перекрестка. Из рисунка можно сделать вывод о том, какие режимы обеспечивают наибольшую и наименьшую пропускную способность перекрестка. Полученные результаты для микроскопической модели CAM-2D и для Aimsun TSS качественно совпадают. Количественного совпадения мы не ожидаем, этот вопрос решается калибровкой моделей.

#### 4.2 Сравнение с экспериментальными данными

По мере накопления данных с датчиков и камер у исследователей появилась возможность изучать глобальные свойства транспортных потоков, анализируя картины пространственно-временных диаграмм и сформулировать эмпирические правила, по которым формируются, эволюционируют и исчезают заторы на автомагистралях. В частности, Б. Кернер сформулировал теорию трех фаз, опираясь на обработанные данные с дорожных детекторов. На рисунке 42 представлены типичные пространственно-временные структуры транспортного потока в реальных данных, измеренных с помощью детекторов, установленных на дорогах в Англии, Германии и США (из [4]). По оси абсцисс отложено время (ч), по оси ординат – расстояние вдоль трассы (км), оттенками серого представлена скорость автомобилей. Самые темные участки соответствуют затору (~0-10 км/ч), более светлые – синхронизированный поток (~60-70 км/ч), белые участки – свободный поток (~110 км/ч).

Измерения проводились на автомагистралях с въездами и съездами, в результате чего были выявлены, в частности, следующие свойства:

- переходы между фазами транспортного потока происходят последовательно: сначала происходит переход F-S, от свободного потока к синхронизированному (то есть, такому, где скорости еще высоки, хотя и ниже скоростей свободного движения, но автомобили вынуждены подстраивать свои скорости друг под друга), и лишь в синхронизированном потоке может

произойти S-J переход - от синхронизованного потока к широкому движущемуся кластеру (проще говоря, к пробке)

- передний фронт синхронизованного потока обычно фиксирован вблизи «бутылочного горла» (сужения, въезда на автомагистраль и т. п.), в то время как широкий движущийся кластер распространяется через любые состояния потока и все узкие места на дороге с постоянной скоростью

- скорость распространения возмущений в транспортном потоке универсальна: в свободном потоке она немного меньше скорости свободного движения и составляет примерно 70 км/ч для автомагистралей, при этом возмущения распространяются вдоль потока, а в плотном потоке эта скорость составляет примерно -15 км/ч, возмущения распространяются против потока.

Иначе говоря, смотря на результаты моделирования, мы ожидаем, во-первых, увидеть структуры, похожие на полосы, как на рисунке 42. Эти полосы должны быть параллельны, так как скорость распространения фазы J в синхронизированном потоке  $F$  – величина универсальная. Во-вторых, эти полосы не должны возникать сразу в свободном потоке, сначала поток должен стать синхронизованным.

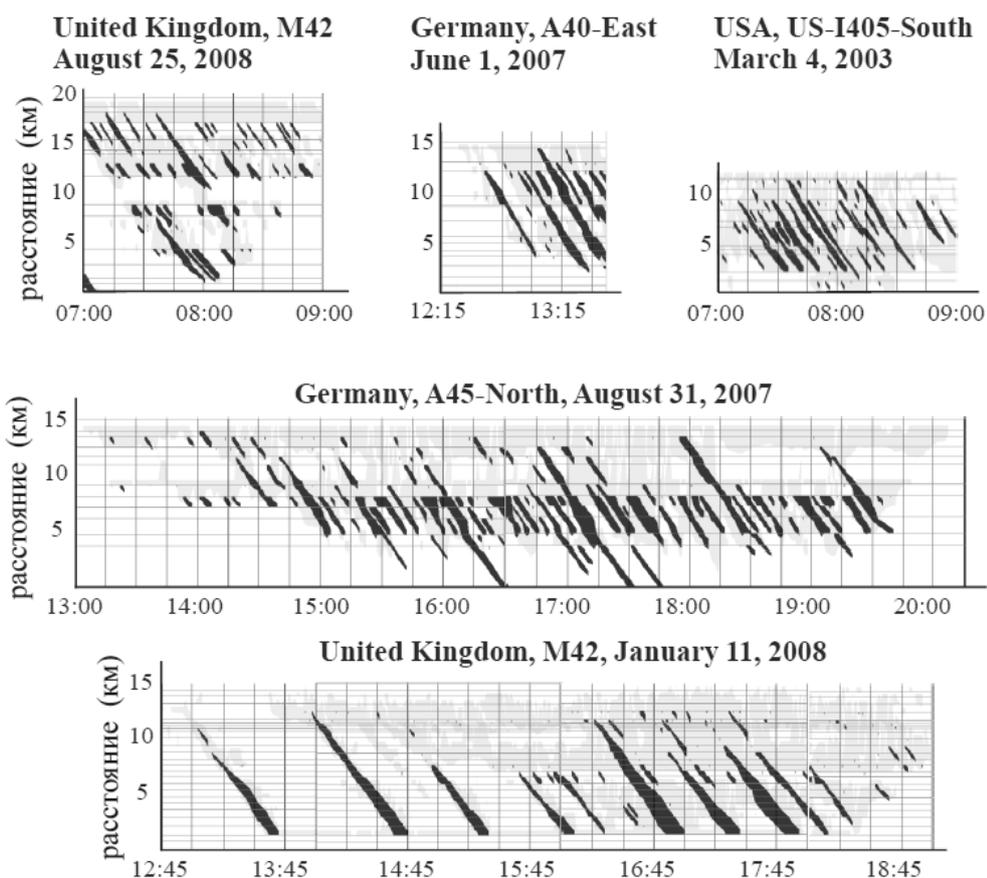


Рисунок 42 – Типичные эмпирические пространственно-временные структуры в транспортных потоках (из [4])

Для того чтобы проверить, насколько созданная автором диссертации модель адекватно описывает реальность, посмотрим, как она воспроизводит основные свойства.

#### 4.2.1 Различные способы осреднения данных

При расчетах с использованием модели клеточных автоматов, в качестве выходных данных мы получаем пространственное положение и мгновенную скорость каждого автомобиля на каждом шаге по времени. Для обработки и анализа данных и получения пространственно-временных структур на диаграмме скоростей нужно перейти от мгновенных локальных скоростей автомобилей к средним.

Осреднение данных – важная задача, так как от выбранного способа осреднения может сильно зависеть полученный результат.

Например, рассмотрим среднее арифметическое (мгновенное) значение скорости на фиксированном отрезке дороги (17).

$$\langle V(t) \rangle = \frac{1}{n(t)} \sum_{\alpha=1}^{n(t)} v_{\alpha}(t) \quad (17)$$

Здесь  $n(t)$  - число машин на конкретном сегменте, а  $v_{\alpha}(t)$  - скорость автомобиля с номером  $\alpha$ . Способ плох, например, тем, что в какие-то временные промежутки  $n(t)$  на определенных участках дороги может принимать значение 0. Также проблему представляет тот факт, что здесь речь идет о моментальной скорости, а у нас в модели время дискретно, и мы можем говорить о моменте времени только с точностью до 1 шага (см. более подробное объяснение ниже).

Если вместо фиксированного пространственного интервала рассмотреть фиксированное число автомобилей  $\Delta N$  (например,  $\Delta N = 10$ ) и считать среднее арифметическое значение скоростей этих  $\Delta N$  автомобилей (18), то указанная выше проблема снимается.

$$V(x, t) = \langle v_{\alpha} \rangle = \frac{1}{\Delta N} \sum_{\alpha=\alpha_0}^{\alpha_0+\Delta N-1} (v_{\alpha}) \quad (18)$$

Однако, в обоих случаях из-за дискретности скорости, принимающей в модели значения от 0 до 4 (а на самом деле, от 0 до 108 км/ч), возникает большой разброс значений этой скорости. Как известно (см., например, [71]), это является показателем того, что среднее арифметическое в данном случае не подходит как способ осреднения.

Похожая проблема возникает при обработке данных с детекторов, снимающих на автодорогах реальные значения скорости ([23], часть I п. 3). Из-за того, что значения средней мгновенной скорости могут существенно меняться за характерный временной интервал  $\Delta t$  (в случае с данными с

детекторов, это временной интервал агрегирования, то есть сбора данных, принимающий значения от 20 с до 5 мин), предлагается усреднить среднюю по пространству моментальную скорость еще и по времени. Полученная таким образом средняя по пространству и времени скорость является средней гармонической скоростью (19):

$$\langle V \rangle \approx \frac{\Delta N}{\sum_{\alpha=\alpha_0}^{\alpha_0+\Delta N-1} \frac{1}{v_\alpha}} = \frac{1}{\left\langle \frac{1}{v_\alpha} \right\rangle} = V_H \quad (19)$$

В модели САМ-2D временной интервал тоже конечный, хотя и гораздо меньше ( $\Delta t=1\text{с}$ ). В нормальных условиях за одну секунду в транспортном потоке, движущемся в обычном режиме, не происходит существенных изменений. Однако, из-за дискретности модели это не так: за шаг по времени скорость может измениться на 1, что составляет примерно 27 км/ч – это существенная величина.

Проблема больших ускорений – известная проблема моделей класса клеточных автоматов, которая, конечно, требует внимания и аккуратной интерпретации. В данном случае, можно сделать вывод, что для нашей модели наиболее подходящий способ работы с данными – использовать гармоническую среднюю скорость (19).

#### *4.2.2 Тестовая задача 2 о въезде на автомагистраль с постоянными входящими потоками. Сравнение методов осреднения.*

Рассмотрим задачу (Тест 2) о въезде на автомагистраль (рисунок 43). Въезд на автомагистраль – стандартный случай «бутылочного горла», поблизости от которого происходит формирование заторов.

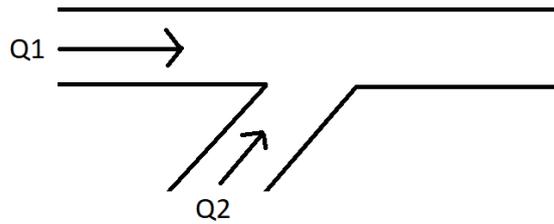


Рисунок 43 — Въезд на автомагистраль

Поток автомобилей по главной дороге  $Q_1=1380$  авт/(полоса×час). Поток по въезду  $Q_2=480$  авт/(полоса×час). Потоки на протяжении расчета не меняются. В нашем случае длина автомагистрали – 20 км, въезд находится примерно посередине. Время расчета - 200 минут. На Рисунке 44 слева представлен расчет, выполненный с использованием модели САМ-2D с осреднением скорости по формуле (19); справа – тот же расчет с осреднением по формуле (18). По оси абсцисс отложено время в минутах, по оси ординат – расстояние в километрах, цветом (оттенками серого) представлена скорость.

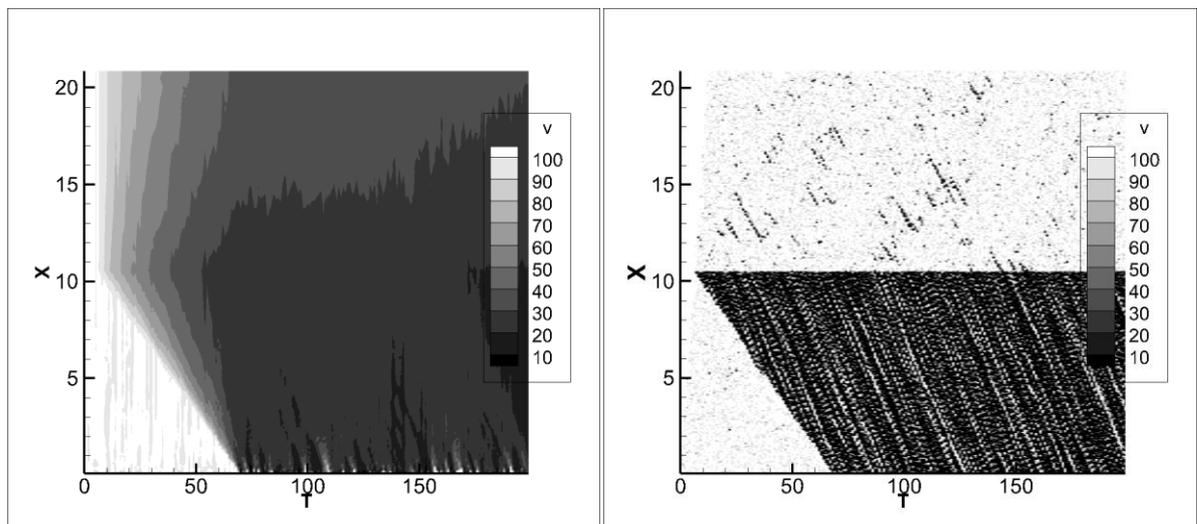


Рисунок 44 — моделирование задачи о въезде на автомагистраль. Пространственно-временные структуры, полученные с использованием средней арифметической скорости (слева), и средней гармонической скорости (справа)

При сравнении обоих рисунков преимущество использования средней гармонической скорости для построения картины пространственно-временных структур очевидно. Хотя в обоих случаях видно влияние въезда на значение скорости, поток переходит от свободного к синхронизованному

и лишь затем к широким движущимся кластерам, на второй картинке полосы, показывающие эволюцию широких движущихся кластеров видны гораздо четче. Картина в целом гораздо больше похожа на реальные экспериментальные данные (рисунок 42). Помимо этого, видно, что фронт уплотнения фиксирован в районе въезда, как ему и положено, исходя из наблюдений.

Так как данная задача является модельной, и в реальных условиях потоки не могут держаться постоянными по несколько часов, мы не ожидаем полного аналога эксперименту, где широкие движущиеся кластеры появляются и исчезают с течением времени. В остальном, полученная при моделировании картина отвечает всем ожидаемым критериям соответствия эксперименту.

Далее предлагается сравнить полученные при помощи модели САМ-2D результаты с результатами других исследователей.

#### *4.2.3 Тестовая задача 3 о въезде на магистраль с переменным потоком $Q_1$ на главной дороге. Сравнение результатов с данными, полученными при помощи модели Кернера-Конхойзера*

Следующая тестовая задача 3 взята из [23], стр. 141. Здесь тоже рассматривается участок автомагистрали с въездом (рисунок 40), но поток по главной дороге  $Q_1$  переменный и меняется со временем согласно графику на рисунке 45: сначала в течение 20 минут нарастает от 1100 авт/(полоса×час) до 1800 авт/(полоса×час), а затем в течение часа спадает обратно до 1100. После двухчасовой отметки до конца расчета поток остается постоянным с этим значением. Поток по второстепенной дороге  $Q_2=400$  авт/(полоса×час).

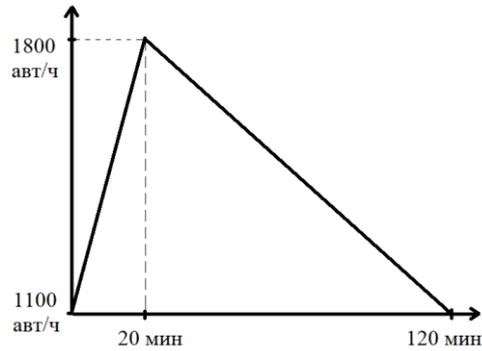


Рисунок 45 — Постановка тестовой задачи 3. Изменение величины потока по главной дороге  $Q_I$  со временем

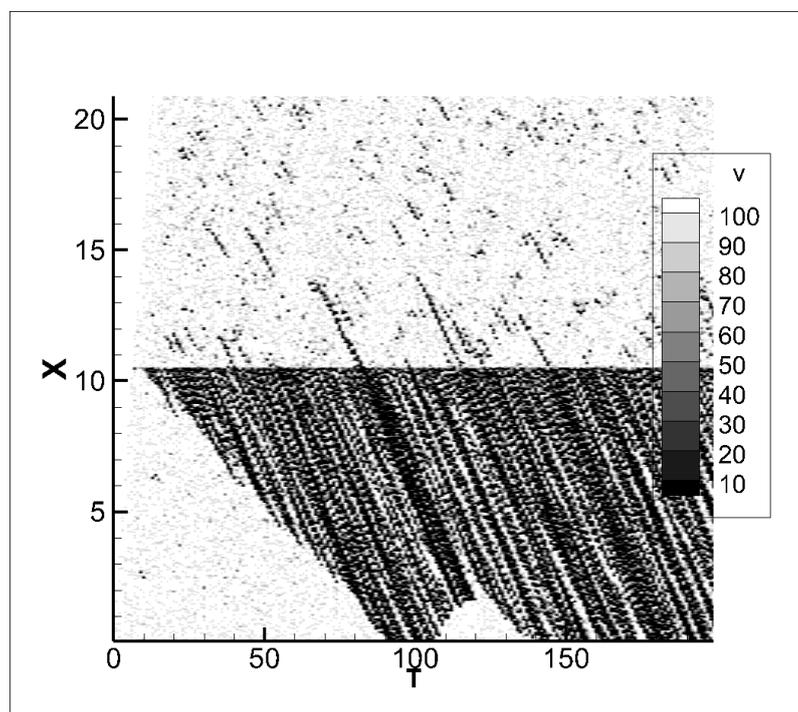


Рисунок 46 — Результат моделирования тестовой задачи 3, полученный при помощи модели SAM-2D

На рисунке 46 представлен результат моделирования тестовой задачи 3, полученный с использованием модели SAM-2D. Условные обозначения аналогичны предыдущим представленным результатам: по оси абсцисс отложено расчетное время в минутах, по оси ординат — расстояние вдоль главной дороги (автомагистрали), оттенками серого представлена осредненная скорость транспортного потока. Использовалось гармоническое среднее.

Результат расчета этой тестовой задачи при помощи модели Кернера-Конхойзера представлен на рисунке 47.

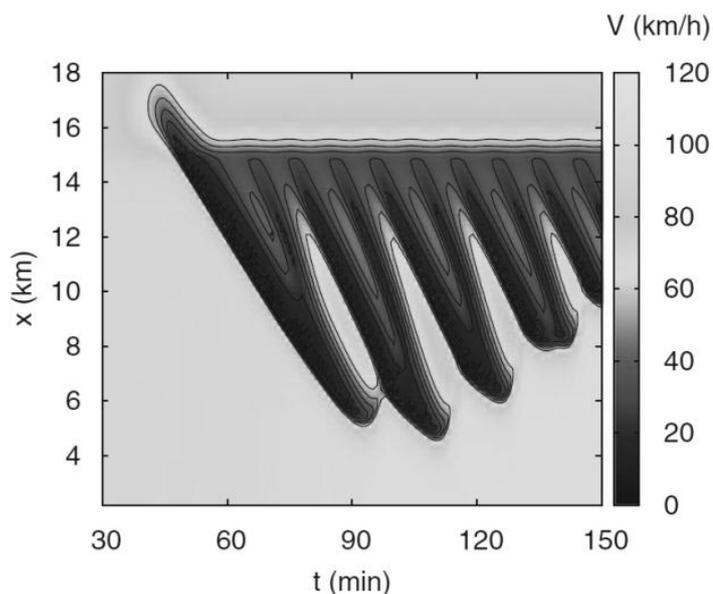


Рисунок 47 — Результат моделирования тестовой задачи при помощи модели Кернера-Конхойзера (взято из [23])

Сравнивая рисунки, можно сделать вывод, что, хотя качественно результаты моделирования совпадают, однако, результат, полученный при помощи СА-модели гораздо ближе к характерным экспериментально наблюдаемым паттернам скоростей на пространственно-временной диаграмме.

#### 4.2.4 Тестовая задача 4 о движении на участке магистрали с выездом и въездом

Следующая тестовая задача 4 также взята из книги Трайбера [23], стр. 144. В задаче рассматривается автомагистраль протяженностью 18 км с выездом и въездом, снабженными дополнительными расширениями дороги для торможения и разгона соответственно при выполнении маневров, как показано на рисунке 48.

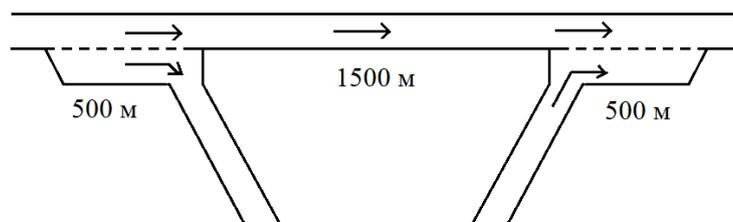


Рисунок 48 — Участок автомагистрали с выездом и въездом

Потоки по всем участкам составляют 500 авт/(полоса×час). Результат моделирования представлен на рисунке 49.

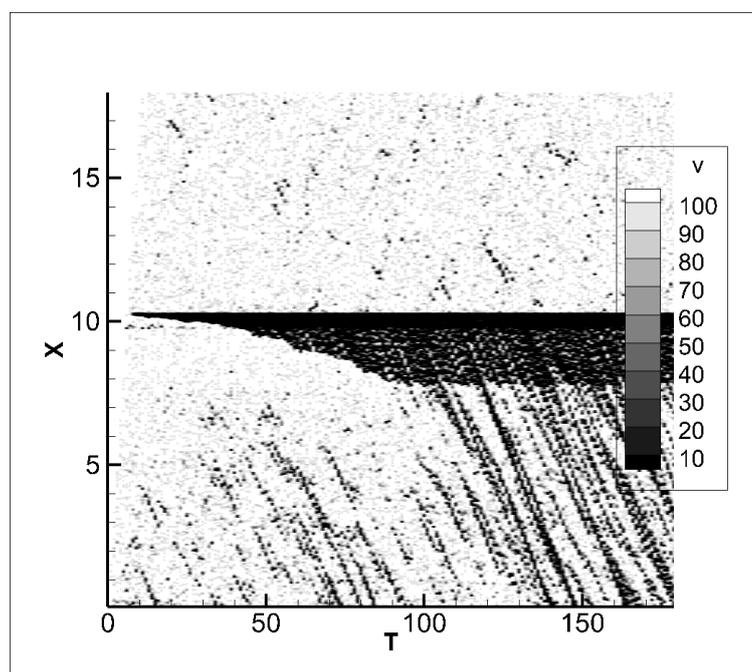


Рисунок 49 — Результат моделирования тестовой задачи 3, полученный при помощи созданной авторами СА-модели

На этот раз сравнение проводится с расчетом, выполненным при помощи газокинетической модели (ГКТ) [72]. Результат представлен на рисунке 50. Как утверждается в работе, полученная картина верно воспроизводит волны трафика, наблюдаемые экспериментально.

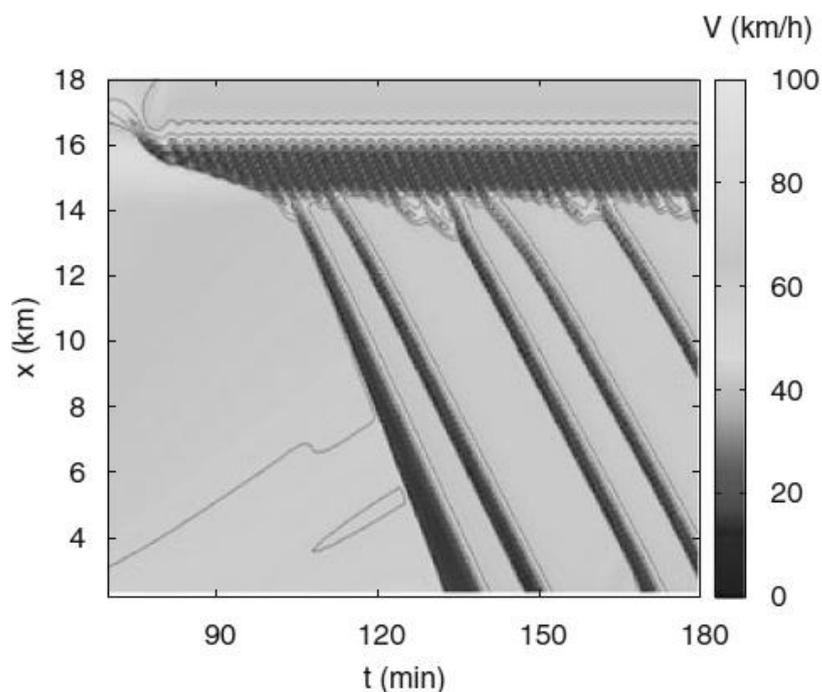


Рисунок 50 — Результат моделирования тестовой задачи при помощи модели GKT (взят из [23])

Как и в случае с задачей 2, полученная при помощи SAM-2D картина качественно совпадает с представленной в книге Трайбера, однако, она имеет вид, более приближенный к реальным данным, получаемым с дорожных датчиков.

#### Выводы по Главе 4

Проведена верификация многополосной модели транспортных потоков SAM-2D на основе решения ряда тестовых задач. При этом были использованы экспериментальные данные, взятые из литературы, и известные эмпирические свойства транспортных потоков.

Был проведен сравнительный анализ различных способов осреднения скоростей отдельных автомобилей для получения значений средней локальной скорости и построения пространственно-временных диаграмм. Выбранный способ был теоретически обоснован и подтвержден при помощи численного эксперимента.

Полученные результаты демонстрируют важность выбора способов

осреднения для получения верной картины пространственно-временных структур и показывают, что неверный выбор может значительно исказить результаты даже на качественном уровне.

С другой стороны, результаты позволяют говорить об адекватности предложенной модели и о возможности ее успешного использования для реальных расчетов.

Полученные в ходе расчетов пространственно-временные структуры на диаграмме скорости транспортного потока совпадают с наблюдаемыми экспериментально. Сравнение вычислений с результатами из литературы показывает большее сходство с реальными данными. Все известные свойства транспортного потока созданная модель САМ-2D хорошо воспроизводит.

## ЗАКЛЮЧЕНИЕ

В рамках диссертационной работы было выполнено исследование истории развития транспортного моделирования и теории клеточных автоматов, проведен анализ методов и подходов к моделированию транспортных потоков, выполнено обоснование выбора направления исследования.

Были разработаны:

1. Двумерная многополосная микроскопическая модель транспортных потоков на основе теории клеточных автоматов CAМ-2D,
2. Алгоритмы движения на различных элементах улично-дорожной сети для созданной модели
3. Различные поведенческие алгоритмы водителей для созданной модели
4. Комплекс программ для моделирования движения автотранспорта на городской улично-дорожной сети, адаптированный для высокопроизводительных вычислительных систем, интегрированный с модулем пользовательского интерфейса и визуализации результатов расчетов.

Были проведены:

5. Вычислительные эксперименты, верифицирующие созданную модель, алгоритмы и программный комплекс.

Результаты, представленные в Главах 2-4, были опубликованы в работах [73-90].

В будущем планируется дальнейшее усложнение созданной модели, например, адаптация ее для описания движения грузового и общественного автотранспорта. Планируется при помощи пользовательского интерфейса реализовать интерактивное управление непосредственно в процессе расчета: смену светофорных режимов, добавление неподвижных препятствий и так далее. Планируется провести исследование оптимизации загрузки

процессоров при параллельных расчетах, поскольку на данном этапе загрузка каждого процессора недостаточна для высокоэффективных вычислений. Сложность данной задачи в том, что расчетные модули должны быть универсальными, а чем больше и сложнее каждый отдельный модуль, тем больше вариантов модулей нужно создать, чтобы моделировать движение на городской дорожной сети. Для решения этой проблемы необходимо найти «золотую середину» между размером модуля (и, соответственно, более оптимальной загрузкой вычислительного узла) и его универсальностью. Один из возможных путей решения – анализ конфигураций связанных элементов УДС, наиболее часто встречающихся на карте города.

Созданная модель, алгоритмы и комплекс программ могут служить основой для создания интеллектуальной транспортной системы города, позволяя решать как краткосрочные, так и долгосрочные задачи транспортного управления и планирования.

**СПИСОК ЛИТЕРАТУРЫ**

1. Nagel K., Schreckenberg M. A Cellular automaton model for freeway traffic // *J. Phys. I France.* – 1992. – V. 2. – P. 2221-2229.
2. Buslaev A.P., Yashina M.V., Strusinskiy P.M., Sokolov P.A. On Qualitative Properties of Cluster Model for Flows on Regular Networks // *Composites, New Challenges for the Greening of Transport, Springer-ECCOMAS, CM3 ECCOMAS Thematic Conference.* – 2015. – P. 24-27.
3. Buslaev A.P., Tatashev A.G., Yashina M.V. On cellular automata, traffic and dynamical systems in graphs // *Int. J. of Eng. & Techn.* – 2018. – V.7 (2.28) – P. 351–356.
4. Kerner B. *The physics of Traffic* – Berlin: Springer, 2004. – 682 p.
5. Kerner B., Klenov S., Schreckenberg M. Simple cellular automaton model for traffic breakdown, highway capacity, and synchronized flow // *Phys. Rev. E* 84 – 2011. – P. 046110.
6. Kerner B., Klenov S., Hermanns G., Schreckenberg M. Effect of driver over-acceleration on traffic breakdown in three-phase cellular automaton traffic flow models // *Physica A: Statistical Mechanics and its Applications.* – 2013. – V.392 (18) – P. 4083-4105.
7. Vranken T.P.E., M. Schreckenberg M. Cellular Automata Intersection Model // *Collective Dynamics.* – 2020. – V.5 A80 – P. 1-25.
8. Zhao H.T., Liu X.R., Chen X.X., Lu J.C. Cellular automata model for traffic flow at intersections in internet of vehicles // *Physica A: Statistical Mechanics and its Applications.* – 2018. – V.494 – P. 40-51.
9. Zhao H.T., Yang S., Chen X.X. Cellular automata model for urban road traffic flow considering pedestrian crossing street // *Physica A: Statistical Mechanics and its Applications.* – 2016. – V.462 – P. 1301-1313.
10. Gao K., Jiang R., Wang B.-H., Wu Q.-S. Discontinuous transition from free flow to synchronized flow induced by short-range interaction between vehicles in a three-phase traffic flow model // *Physica A: Statistical Mechanics and its Applications.* – 2009. – V.388 (15–16) – P. 3233-3243.

11. <https://www.aimsun.com>
12. <https://www.kiam.ru/MVS/resources/k100.html>
13. Lighthill M.H., Witham G.B. On kinematic waves: A theory of traffic flow on long crowded roads // Proc. Royal Soc. Ser. A. – 1955. – V.229 – P. 317–345.
14. Kerner B., Konhäuser P. Cluster effect in initially homogeneous traffic flow // Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics. – 1993. – V.48 – P. R2335-R2338.
15. Kerner, B., Konhäuser, P., Schilke, M. Deterministic spontaneous appearance of traffic jams in slightly inhomogeneous traffic flow // Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics. – 1995. – V. 51 – P. 6243-6246.
16. Сухинова А. Б., Трапезникова М. А., Четверушкин Б. Н., Чурбанова Н. Г. Двумерная макроскопическая модель транспортных потоков // Матем. моделирование. – 2009. – т. 21 №2 – С.118–126.
17. Prigogine I., Andrews F. C. A boltzmann-like approach for traffic flow // Opns Res. – 1960. – Vol. 8 – P. 789–797.
18. Paveri-Fontana S.L. On Boltzmann like treatments for traffic flow. // Transportation Research. – 1975. – V.9 No 225.
19. Helbing D. Gas-kinetic derivation of Navier-Stokes-like traffic equations. // Phys. Rev. E. – 1996. - V.53 No 2366.
20. Helbing D. Structure and instability of high-density equations for traffic flow. // Phys. Rev. E. – 1998. – V.57 No 6176.
21. Bando, M., Hasebe, K., Nakayama, A., Shibata, A. and Sugiyama, Y. Structure Stability of Congestion in Traffic Dynamics // Japanese Journal of Industrial and Applied Mathematics. – 1994. – V. 11 – P. 203-223.
22. Bando, M., Hasebe, K., Nakanishi, K., Nakayama, A., Shibata, A., Sugiyama, Y. Phenomenological Study of Dynamical Model of Traffic Flow // Journal of Physics I France. – 1995. – V. 5 – P. 1389-1399.
23. Treiber M., Kesting A. Traffic Flow Dynamics. Data, Models and Simulation – Berlin-Heidelberg: Springer, 2013. – 503 p.

24. Treiber M., Hennecke A., Helbing D. Congested traffic states in empirical observations and microscopic simulations // *Physical Review E*. – 2000. – V. 62 (2) – P. 1805–1824.
25. Von Neumann J. The general and logical theory of automata // *Cerebral Mechanisms in Behavior*. – 1948. – P. 1–41.
26. Фон Нейман Дж. Теория самовоспроизводящихся автоматов - М.: УРСС, 2010. – 384 с.
27. Gardner M. Mathematical Games - The Fantastic Combinations of John Conway's New Solitaire Game 'Life' // *Scientific American*. – 1970. – V. 223 – P. 120–123.
28. S. Wolfram. A new kind of science. <https://www.wolframscience.com/nks/>
29. <http://atlas.wolfram.com/>
30. Бугаев А. С., Буслаев А. П., Козлов В. В., Таташев А. Г., Яшина М. В. Обобщенная транспортно-логистическая модель как класс динамических систем // *Матем. моделирование*. – 2015. – Т. 27 №12. – С. 65–87.
31. Maerivoet S., De Moor B. Cellular automata models of road traffic // *Physics Reports*. – 2016. – V. 419 – P. 1-64.
32. Li X., Sun J.-Q. Effects of turning and through lane sharing on traffic performance at intersections // *Physica A: Statistical Mechanics and its Applications*. – 2016. – V. 444 – P. 622-640.
33. Vasic J., Heather J., Ruskin H.J. Cellular automata simulation of traffic including cars and bicycles // *Physica A: Statistical Mechanics and its Applications*. – 2012. – V. 391 (8) – P. 2720-2729.
34. Lárraga M.E., Alvarez-Icaza L. Cellular automaton model for traffic flow based on safe driving policies and human reactions // *Physica A: Statistical Mechanics and its Applications*. – 2010. – V. 389 (23) – P. 5425-5438.
35. Combinido J.S.L., Lim M.T. Modeling U-turn traffic flow // *Physica A: Statistical Mechanics and its Applications*. – 2010. – V. 389 (17) – P. 3640-3647.

36. Zhu H.B. Numerical study of urban traffic flow with dedicated bus lane and intermittent bus lane // *Physica A: Statistical Mechanics and its Applications*. – 2010. – V. 389 (16) – P. 3134-3139.
37. Qian W.L., Wang B., Lin K., Machado R.F., Hama Y. A mesoscopic approach on atability and phase transition between different traffic flow states // *International Journal Of Non-Linear Mechanics*. – 2017. - V. 89 - P. 59-68.
38. Майоров Н.Н., Романек В.А. Вопросы выбора математических моделей для исследования пассажирских потоков в транспортных системах // *Системный Анализ И Логистика*. – 2017. - т. 1 №14 - С. 39-45.
39. Мазурин Д.С. Комплексная методика моделирования транспортных систем на основе моделей динамического распределения потоков // *Труды Института Системного Анализа Российской Академии Наук*. – 2017. – т. 67 №3 – С. 3-12.
40. Кармадонов В.Ю., Косов А.А. Моделирование транспортного потока на основе виртуальной вычислительной среды // *Информационные Технологии И Проблемы Математического Моделирования Сложных Систем*. – 2017. - № 18 – С. 80-87.
41. Сорокин А.А., Яковлев С.В., Орлова А.Ю. Имитационная модель транспортного узла с режимом оптимизации параметров // *Вестник Севкавгти*. – 2017. – т. 4 №31 – С. 246-250.
42. Разумов Д.С., Катаев М.Ю., Шелестов А.А. Алгоритмы управления потоком автотранспорта в городских условиях // *Вестник Современных Исследований*. – 2018. – т. 6.1 №21 – С. 471-473.
43. Нургалиев Е.Р. Имитационное моделирование улично-дорожной сети города // *Актуальные Направления Научных Исследований XXI Века: Теория и Практика*. – 2017. – т.5 №7-1 (33-1) – С. 172-176.
44. Zhou J., Zhang H.L., Wang C.P., Shi Z.K. A new lattice model for single-lane traffic flow with the consideration of driver's memory during a period of time. // *International Journal Of Modern Physics C*. – 2017. - V. 28 No 7. – P. 1750086

45. Jin D., Zhou J., Zhang H.L., Wang C.P., Shi Z.K. Lattice hydrodynamic model for traffic flow on curved road with passing. // *Nonlinear Dynamics*. – 2017. – V. 89 No 1 – P. 107-124.
46. Kaur R. Sharma S. Analysis of driver's characteristics on a curved road in lattice model // *Physica A: Statistical Mechanics And Its Applications*. – 2017. – V.471 – P. 59-67.
47. Qian Y., Zeng J., Wang N., Zhang J., Wang B. A traffic flow model considering influence of car-following and its echo characteristics // *Nonlinear Dynamics*. – 2017. – V.89 No 2 – P. 1099-1109.
48. Курц В.В., Ануфриев И.Е. Модель автомобильного трафика с запаздывающим аргументом – исследование устойчивости на кольце // *Математическое Моделирование*. – 2017. – т. 29 № 4 – С. 88-100.
49. Kuang H., Xu Z.P., Li X.L., Lo S.M. An extended car following model accounting for the honk effect and numerical tests // *Nonlinear Dynamics*. – 2017. – V. 87 No 1 – P. 149-157.
50. Wang P., Yu G., Wu X., Wang Y. Linear and nonlinear stability analysis of an extended car following model considering pedestrians on adjacent lane // *Nonlinear Dynamics*. – 2017. – V. 88 No 1 – P. 777-789
51. Qi L., Zheng Z., Gang L. A cellular automation model for ship traffic flow in waterways // *Physica A: Statistical Mechanics And Its Applications*. – 2017. – V. 471 – P. 705-717.
52. Chen J., Jiang R., Lin L. Assigning on ramp flows to maximize capacity of highway with two on-ramps and one off-ramp in between // *Physica A: Statistical Mechanics And Its Applications*. – 2017. – V.465 – P. 347-357
53. Qian Y.S., Feng X., Zeng J.W. A cellular automata traffic flow model for three phase theory // *Physica A: Statistical Mechanics And Its Applications*. – 2017. – V. 479 – P. 509-526.
54. Guzman H.A., Larraga M.E., Alvarez-Icaza L., Carvajal J. A cellular automata model for traffic flow based on kinetics theory, vehicles capabilities and driver

- reactions. *Physica A: Statistical Mechanics And Its Applications*. –2018. – V.491 – P. 528-548.
55. Dong P., Wang X., Yun L., Fan H. Research on the characteristics of mixed traffic flow based on an improved bicycle model // *Simulation*, Sage Publications. – 2018. – V.94 N 5 – P. 451-462.
56. Zeng J.W., Qian Y.S., Wang H., Wei X.T., modeling and simulation of traffic flow under different combination setting of taxi stop and bus stop // *Modern Physics Letters B*. – 2018. – V. 32 N 25 – P. 1850301
57. Kaur R.1, Sharma S. Modeling and simulation of driver's anticipation effect in a two lane system on curved road with slope // *Physica A: Statistical Mechanics And Its Applications*. –2018. – V. 499 – P. 110-120.
58. Su Z., Liu S., Deng W., Li W., Cai X., transportation dynamics on networks of heterogeneous mobile agents // *Physics Letters A*. – 2019. – V. 523 – P. 1379-1386.
59. Yao W., Jia N., Zhong S., Li L., best response game of traffic on road network of non-signalized intersections // *Physica A: Statistical Mechanics And Its Applications*. – 2018. – V. 490 – P. 386-401.
60. Herty M., Fazekas A., Visconti G., a two-dimensional data-driven model for traffic flow on highways // *Networks And Heterogeneous Media*. – 2018. – V. 13 N 2 – P. 217-240.
61. Степанов Е.П. Анализ эффективности демультимплексирования транспортных потоков // *Моделирование и анализ информационных систем*. –2019. – т.26 №1(79) – С. 170-190.
62. Морозов И. И., Гасников А. В., Тарасов В. Н., Холодов Я. А., Холодов А. С. Численное исследование транспортных потоков на основе гидродинамических моделей // *Компьютерные исследования и моделирование*. – 2011. – т.3:4 – С. 389–412.
63. Холодов Я.А., Алексеенко А. Е., Васильев М. О., Холодов А. С. Построение математической модели дорожного перекрестка на

- основе гидродинамического подхода // Компьютерные исследования и моделирование. – 2014. – т.6:4 – С. 503–522.
64. Прокопцев Н. Г., Алексеенко А. Е., Холодов Я. А., Использование сверточных нейронных сетей для прогнозирования скоростей транспортного потока на дорожном графе // Компьютерные исследования и моделирование. – 2018. – т. 10:3 – С. 359–367.
65. Веренцов С. И., Магеррамов Э. А., Виноградов В. А., Гизатуллин Р. И., Алексеенко А. Е., Холодов Я. А. Байесовская вероятностная локализация автономного транспортного средства путем ассимиляции сенсорных данных и информации о дорожных знаках // Компьютерные исследования и моделирование. – 2018. – т. 10:3 – С. 295–303.
66. Введение в математическое моделирование транспортных потоков. / Гасников А.В., Кленов С.Л., Нурминский Е.А., Холодов Я.А., Шамрай Н.Б.; Приложения: Бланк М.Л., Гасникова Е.В., Замятин А.А. и Малышев В.А., Колесников А.В., Райгородский А.М; Под ред. А.В. Гасникова. — М.:МФТИ, 2010. — 362 с.
67. Smirnova M., Bogdanova A., Smirnov N., Kiselev A., Nikitin, V., Manenkova A. Multi-Lane Unsteady-State Traffic Flow Models // Journal of Mechatronics. – 2014. – V.2.
68. Smirnov N., Kiselev A., Nikitin V., Silnikov M., Manenkova A. Hydrodynamic traffic flow models and its application to studying traffic control effectiveness // WSEAS Transactions on Fluid Mechanics. – 2014. – V. 9. – P. 178-186.
69. Barlovic R. e. a. Metastable states in cellular automata for traffic flow // The European Physical Journal B-Condensed Matter and Complex Systems. – 1998. – V. 5 No. 3 – P. 793–800.
70. Герман М.С., Ермаков А.В. Архитектура и функциональные возможности системы визуализации потоков автотранспорта на элементах улично-дорожной сети // Препринты ИПМ им. М.В.Келдыша. – 2017. – № 145 – 21 с.

71. Измайлова М. О., Рахманкулов И. Ш. Категория «средняя величина» и ее методологическое значение в научном исследовании. - Казань: Изд. Казанского Университета, 1982. - 143 с.
72. Helbing D., Hennecke A., Shvetsov V., Treiber M. MASTER: Macroscopic traffic simulation based on a gas-kinetic, non-local traffic model // *Transportation Research Part B: Methodological*. – 2001. – V. 35 – P. 183-211.
73. Churbanova N.G., Chechina A.A., Furmanov I. R., Trapeznikova M.A. Microscopic Model for Simulation of Traffic Flows on Multilane Highways and Crossroads // *ECCOMAS 2012 - European Congress on Computational Methods in Applied Sciences and Engineering, e-Book Full Papers*. – 2012. – P. 11 - 17.
74. Трапезникова М.А., Чечина А.А., Чурбанова Н.Г., Поляков Д.Б. Математическое моделирование потоков автотранспорта на основе макро- и микроскопических подходов // *Вестник АГТУ Сер.: Управление, вычислительная техника и информатика*. – 2014. - №1 - С. 130-139.
75. Chechina A., Churbanova N., Trapeznikova M. Different Approaches to the Multilane Traffic Simulation // *Traffic and Granular Flow '13*. – 2015. – No. 19 – P. 361-368.
76. Трапезникова М.А., Чечина А.А., Чурбанова Н.Г. Описание динамики транспортных потоков на элементах улично-дорожной сети с использованием двумерных математических моделей // *Препринты ИПМ им. М.В. Келдыша*. – 2016. – №93 – 20 с.
77. Чечина А.А., Герман М.С., Ермаков А.В., Трапезникова М.А., Чурбанова Н.Г. Моделирование и визуализация потоков автотранспорта на элементах улично-дорожной сети с использованием комплекса программ САМ-2D // *Препринты ИПМ им. М.В.Келдыша*. – 2016. – № 124 – 17 с.
78. Трапезникова М.А., Чечина А.А., Чурбанова Н.Г. Двумерная модель клеточных автоматов для описания динамики транспортных потоков на элементах улично-дорожной сети // *Математическое моделирование*. – 2017. – т. 19 № 9 – С. 110-120.

79. Чечина А.А. Новые алгоритмы перестроения автомобилей для микроскопической модели транспортных потоков на основе теории клеточных автоматов // Препринты ИПМ им. М.В. Келдыша. – 2017. – №136 – 14с.
80. Chechina A., Churbanova N., Trapeznikova M. Multilane Traffic Flow Modeling Using Cellular Automata Theory // EPJ Web of Conferences. – 2018. – V. 173 – P. 06003.
81. Chechina A., Churbanova N., Trapeznikova M., Ermakov A., German M. Traffic flow modelling on road networks using cellular automata theory // Int. J. of Eng. & Techn. – 2018. – V. 7 N. 2.28 – P. 225-227.
82. Чечина А.А. Воспроизведение экспериментальных пространственно-временных структур в транспортных потоках при помощи математической модели на основе теории клеточных автоматов // Препринты ИПМ им. М.В.Келдыша. – 2018. – № 236 – 16 с.
83. Chechina A., Churbanova N., Trapeznikova M. Reproduction of experimental spatio-temporal structures in traffic flows using mathematical model based on cellular automata theory // Periodicals of Engineering and Natural Sciences. – 2019. – V.7 No. 1 – P. 76-81.
84. Churbanova N., Chechina A., Trapeznikova M., Sokolov P. Simulation of traffic flows on road segments using cellular automata theory and quasigasdynamic approach // Mathematica Montisnigri. – 2019. – V. XLVI – P. 72-90.
85. Churbanova N., Chechina A., Trapeznikova M., Ermakov A., German M., Sokolov P., Bozorov O. CMMSE-2019: Simulation and visualization of vehicular traffic on road networks using high performance computing systems // Computational and Mathematical Methods. – 2020 - V. 2 No. 3 - e1082.
86. Chechina A., Churbanova N., Trapeznikova M. Modelling traffic on road junctions on parallel computing systems using cellular automata approach // International Journal of Interactive Mobile Technologies (iJIM). – 2020. – V. 14 No. 10 – P. 178-185.

87. Trapeznikova M.A., Churbanova N. G., Chechina A.A., Ermakov A.V., German M.S. Supercomputer Technology for Traffic Simulation in a Metropolis // 2020 International Conference on Engineering Management of Communication and Technology (EMCTECH, Vienna, Austria, 2020) IEEE. – 2020. – P. 1-4.
88. Chechina A., Churbanova N., Trapeznikova M. Driver Behaviour Algorithms for the Cellular Automata-Based Mathematical Model of Traffic Flows // EPJ Web Conf. – 2021. – V. 248 – P. 02002.
89. Чечина А.А., Чурбанова Н.Г., Трапезникова М.А.. Сравнение воспроизведения пространственно-временных структур транспортных потоков при использовании различных способов осреднения данных // Математическое моделирование. – 2021. – т. 33(1) – С. 25-31.
90. Чечина А.А. Алгоритмы поведения водителей на нерегулируемых перекрестках с приоритетом и при объезде препятствий // Математическое моделирование. – 2021. – т. 33(9) – С. 47-59.

**ПРИЛОЖЕНИЕ 1 ЛИЦЕНЗИОННЫЙ ДОГОВОР И АКТ ПРИЕМА-ПЕРЕДАЧИ ПО ПРОЕКТУ «Разработка архитектуры, принципов построения и программного обеспечения суперкомпьютерной информационно-аналитической платформы прикладного моделирования, прогнозирования и экспертиз транспортных процессов и систем в виде сетевой компьютерной лаборатории»**

**ЛИЦЕНЗИОННЫЙ ДОГОВОР № Р178/2016**

г. Москва

" 23 " декабря 2016 г.

Федеральное государственное учреждение «Федеральный исследовательский центр Институт прикладной математики им. М.В. Келдыша Российской академии наук» (ИПМ им. М.В. Келдыша РАН), именуемое в дальнейшем «Лицензиар», в лице директора Аптекарева Александра Ивановича, действующего на основании Устава ИПМ им. М.В. Келдыша РАН, с одной стороны, и Государственное унитарное предприятие «Научно-исследовательский и проектный институт Генерального плана города Москвы» (ГУП «НИ и ПИ Генплана Москвы»), именуемое в дальнейшем «Лицензиат», в лице директора Гармаш Оксаны Алексеевны, действующей на основании Устава ГУП «НИ и ПИ Генплана Москвы», с другой стороны, именуемые в дальнейшем Стороны, а по отдельности – Сторона, и руководствуясь Договором о дальнейшем использовании результатов прикладных научных исследований № 5-14/436 от 18.04.2014, заключенным между Сторонами, заключили настоящий договор (далее – Договор) о нижеследующем.

**1. ПРЕДМЕТ ДОГОВОРА**

1.1. Лицензиар предоставляет Лицензиату на условиях, изложенных в настоящем Договоре, неисключительную лицензию на использование следующих программ для ЭВМ:

- «Унифицированная аналитическая платформа многокритериальной поддержки решений», свидетельство о государственной регистрации № 2015661754;
- «Программа микромоделирования многополосных автомобильных дорог и перекрёстков BTSSIM», свидетельство о государственной регистрации № 2016610003;
- «Программа микроскопического моделирования многополосных транспортных потоков САМ-2D», свидетельство о государственной регистрации № 2016619894;
- «Программа формирования маршрутов в динамическом транспортном потоке NXFLOW-T», свидетельство о государственной регистрации № 2016663279;
- «Информационно-аналитическая система поддержки принятия транспортных решений и учебно-методической работы при подготовке специалистов», свидетельство о государственной регистрации № 2016663280.

1.2. Лицензиар гарантирует, что является правообладателем исключительного права на программы для ЭВМ.

1.3. Лицензиар передает лицензиату, коды вышеперечисленных программ для ЭВМ и копии свидетельств к ним на электронном носителе.

В подтверждение факта передачи прав пользования Лицензиар оформляет Акт приема-передачи в 3-х экземплярах.

1.4. Права пользования, указанные в п.1.1 Договора, считаются предоставленными Лицензиату с момента подписания Сторонами Акта приема-передачи.

## 2. ПОРЯДОК ИСПОЛЬЗОВАНИЯ ОБЪЕКТА ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

2.1. Лицензиату передаются следующие права: на полное или частичное воспроизведение в любой форме и любыми способами.

2.2. Лицензиар не передает Лицензиату следующие права:

- право на распространение экземпляров программы для ЭВМ;
- право заключать сублицензионные договоры без предварительного письменного согласия Лицензиара.

2.3. Лицензиар обязуется оказывать Лицензиату по его запросу консультационную помощь в освоении программ для ЭВМ по Договору, а также для обучения персонала Лицензиата методам и приемам работы, относящимся к переданным программам. Лицензиар по просьбе Лицензиата командировывает на предприятия Лицензиата необходимое количество специалистов. Порядок возмещения расходов Лицензиара, связанных с направлением специалистов на предприятия Лицензиата, определяется дополнительным соглашением к Договору.

2.4. В случае если к Лицензиату будут предъявлены претензии или иски по поводу нарушения прав третьих лиц в связи с использованием лицензии по Договору, Лицензиат известит об этом Лицензиара. Лицензиат по согласованию с Лицензиаром обязуется урегулировать такие претензии или обеспечить судебную защиту.

2.5. Лицензия, выдаваемая Лицензиату по настоящему Договору, является неисключительной. Лицензиар вправе выдавать лицензии на использование программы для ЭВМ другим лицам.

2.6. Использование Лицензиатом программ для ЭВМ допускается на всей территории Российской Федерации.

## 3. СРОК ДОГОВОРА И ТЕРРИТОРИЯ ИСПОЛЬЗОВАНИЯ БАЗЫ ДАнных/ПРОГРАММЫ ДЛЯ ЭВМ

3.1. Настоящий Договор заключен сроком на 5 лет.

3.2. Договор вступает в силу с момента его подписания сторонами.

3.3. Лицензионный Договор действует на территории Российской Федерации.

## 4. ЛИЦЕНЗИОННОЕ ВОЗНАГРАЖДЕНИЕ

4.1. Программы для ЭВМ, передаваемые в рамках настоящего Лицензионного Договора, разработаны в рамках совместного выполнения работ по Договору от 18 апреля 2014 г. № 5-14/436, в связи с чем Лицензионное вознаграждение за

предоставленное право использования базы данных/программы для ЭВМ, предусмотренное настоящим Договором, отсутствует.

## 5. ОТВЕТСТВЕННОСТЬ СТОРОН

5.1. В течение срока действия настоящего Договора Стороны обязуются незамедлительно информировать друг друга обо всех произведенных ими усовершенствованиях и улучшениях, касающихся программ для ЭВМ (п.1.1).

5.2. Лицензиат обязуется сохранять конфиденциальность информации, полученной от Лицензиара, и предоставлять ее только по сублицензионным договорам с согласия Лицензиара.

5.3. За неисполнение или ненадлежащее исполнение Сторонами обязательств, принятых на себя в соответствии с настоящим Договором, Стороны несут ответственность в соответствии с действующим российским законодательством и настоящим Договором.

5.4. Лицензиат несет ответственность перед Лицензиаром за действия сублицензиатов по сублицензионным договорам.

5.5. Все изменения и дополнения к Договору действительны, если совершены в письменной форме и подписаны обеими Сторонами. Соответствующие дополнительные соглашения Сторон являются неотъемлемой частью Договора.

5.6. Стороны освобождаются от ответственности за неисполнение или ненадлежащее исполнение своих обязательств по настоящему Договору в случае действия обстоятельств непреодолимой силы, прямо или косвенно препятствующих исполнению настоящего Договора, то есть таких обстоятельств, которые не зависят от воли Сторон, не могли быть ими предвидены в момент заключения Договора и предотвращены разумными средствами при их наступлении.

## 6. УСЛОВИЯ ДОСРОЧНОГО РАСТОРЖЕНИЯ ДОГОВОРА

6.1. Каждая из Сторон имеет право досрочно расторгнуть настоящий Договор путем направления письменного уведомления не менее чем за 10 дней до предполагаемой даты расторжения, если другая Сторона не выполнит какое-либо существенное условие настоящего Договора.

6.2. Если настоящий Договор будет расторгнут до истечения срока его действия вследствие нарушения какого-либо условия Договора Лицензиатом, то Лицензиат лишается переданных ему по настоящему Договору прав и обязуется возвратить Лицензиару всю документацию, связанную с базой данных/программой для ЭВМ и переданную Лицензиату.

6.3. Споры и разногласия, возникающие из настоящего Договора или в связи с ним, будут решаться Сторонами путем переговоров.

6.4. Все споры по настоящему договору, не урегулированные сторонами самостоятельно, рассматриваются в судебном порядке

## 7. ЗАКЛЮЧИТЕЛЬНЫЕ ПОЛОЖЕНИЯ

7.1. Договор составлен в 3 (трех) экземплярах, имеющих одинаковую юридическую силу.

7.2. Адреса, реквизиты и подписи Сторон:

### Лицензиар

Федеральное государственное учреждение «Федеральный исследовательский центр Институт прикладной математики им. М.В. Келдыша Российской академии наук» (ИПМ им. М.В. Келдыша РАН)  
Адрес места нахождения:  
Российская Федерация, 125047, г. Москва, Миусская пл. 4  
Платежные реквизиты:  
ИНН 7710063939, КПП 771001001, УФК по г. Москве (л/сч 20736Ч33640) в Отделении 1 Москва  
р/сч 40501810600002000079  
ОКПО 02699381,  
ОКТМО МО Тверской: 45382000  
БИК 044583001

### Лицензиат

Государственное унитарное предприятие «Научно-исследовательский и проектный институт Генерального плана города Москвы» (ГУП «НИ и ПИ Генплана Москвы»)  
Адрес места нахождения:  
125047, г. Москва, ул. 2-я Брестская, дом 2/14  
ИНН 7710030490,  
КПП 771001001  
р/с 40602810738040100009  
Тверское отделение №7982/01536  
Сбербанка России ОАО, г. Москвы  
к/с 30101810400000000225,  
БИК 044525225, ОКПО 03997732

Директор



А.И. Аптекарев

М.П.

Директор



О.А. Гармаш

**Лицензиар**

Федеральное государственное  
учреждение «Федеральный  
исследовательский центр Институт  
прикладной математики им. М.В.  
Келдыша Российской академии наук»  
(ИПМ им. М.В. Келдыша РАН)  
125047, г. Москва, Миусская пл., д. 4

**Лицензиат**

Государственное унитарное  
предприятие «Научно-  
исследовательский и проектный  
институт Генерального плана города  
Москвы» (ГУП «НИ и ПИ Генплана  
Москвы»)  
125047, г. Москва, 2-ая Брестская ул.,  
д. 2/14

**АКТ**

приёма-передачи неисключительных прав на использование результатов  
интеллектуальной деятельности, разработанных в ходе выполнения  
ИПМ им. М.В. Келдыша РАН

Плана-графика исполнения обязательств по Соглашению о предоставлении субсидии  
от «30» июня 2014 г. № 14.604.21.0052

Тема: «Разработка архитектуры, принципов построения и программного обеспечения  
суперкомпьютерной информационно-аналитической платформы прикладного  
моделирования, прогнозирования и экспертиз транспортных процессов и систем в  
виде сетевой компьютерной лаборатории»

составлен «26» декабря 2016 г.

Мы, нижеподписавшиеся, представитель Лицензиара директор ИПМ им. М.В.  
Келдыша РАН Александр Иванович Аптекарев, с одной стороны, и  
представитель Лицензиата директор ГУП «НИ и ПИ Генплана Москвы» Оксана  
Алексеевна Гармаш, с другой стороны,

составили настоящий акт о том, что:

в соответствии с Лицензионным договором № P178/2016 от «23» декабря 2016 г.  
Лицензиар передал, а Лицензиат принял следующие программы для ЭВМ:

	Наименование программного продукта	Кол-во
1	Унифицированная аналитическая платформа многокритериальной поддержки решений	1
2	Программа микромоделирования многополосных автомобильных дорог и перекрёстков BTSSIM	1
3	Программа микроскопического моделирования многополосных транспортных потоков CAM-2D	1
4	Программа формирования маршрутов в динамическом транспортном потоке NXFLOW-T	1
5	Информационно-аналитическая система поддержки принятия транспортных решений и учебно-методической работы при подготовке специалистов	1

1. Права передаются на условиях простой (неисключительной лицензии), а их объем ограничен в соответствии с условиями договора № P/78/2016 от «23» декабря 2016 г.
2. Лицензиар передает лицензиату, коды вышеперечисленных программ для ЭВМ и копии свидетельств к ним на электронном носителе.
3. Вознаграждение переданных неисключительных прав на использование результатов интеллектуальной деятельности (программ для ЭВМ) на основании лицензионного договора отсутствует.
4. Лицензиар выполнил все обязательства в полном объеме в срок с надлежащим качеством.
5. Лицензиат претензий к Лицензиару не имеет.

Настоящий Акт составлен в 3-х экземплярах, идентичных по содержанию и имеющих одинаковую юридическую силу: 2 экземпляра – для Лицензиара и 1 экземпляр – для Лицензиата.

**Лицензиар**

Директор  
ИПМ им. М.В. Келдыша РАН

А.И. Аптекарев

М.П.



**Лицензиат**

Директор  
ФУИ «НИИ и ПИ Генплана Москвы»

О.А. Гармаш



**ПРИЛОЖЕНИЕ 2 СВИДЕТЕЛЬСТВО О ГОСУДАРСТВЕННОЙ  
РЕГИСТРАЦИИ ПРОГРАММЫ САМ-2D**

**РОССИЙСКАЯ ФЕДЕРАЦИЯ**



**СВИДЕТЕЛЬСТВО**

о государственной регистрации программы для ЭВМ

**№ 2016662572**

**Программа микроскопического моделирования  
многополосных транспортных потоков САМ-2D**

Правообладатель: *Федеральное государственное учреждение  
«Федеральный исследовательский центр Институт прикладной  
математики им. М.В. Келдыша Российской академии наук»  
(ИПМ им. М.В. Келдыша РАН) (RU)*

Автор: *Чечина Антонина Александровна (RU)*

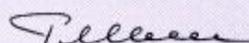
Заявка № **2016619894**

Дата поступления **22 сентября 2016 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **15 ноября 2016 г.**

Руководитель Федеральной службы  
по интеллектуальной собственности

 **Г.П. Ивлиев**

