Федеральное государственное бюджетное образовательное учреждение
высшего образования «Воронежский государственный университет»

На правах рукописи

Кузнецов Александр Владимирович

Модели движения, взаимодействия и сети связи мобильных агентов в иерархических системах на основе клеточных автоматов

Специальность 05.13.01 —
«Системный анализ, управление и обработка информации (радиотехника, автоматика, связь)»

Диссертация на соискание учёной степени
доктора физико-математических наук

Научный консультант:
доктор технических наук, профессор
Леденева Татьяна Михайловна

Воронеж — 2019

Оглавление

Стр.

Введение 7

Глава 1. Обзор предыдущих работ 35
1.1 Обзор классификации, методов и методологий агентных систем . . 35
1.2 Задачи, решаемые с помощью агентного подхода в естественных, технических и общественных науках 40

Глава 2. Модель движения и взаимодействия иерархически организованных агентов на основе клеточного автомата . . . 57
2.1 Термины и определения 57
2.2 Алгоритмы построения случайных ландшафтов с фиксированными характеристиками 61
2.2.1 Постановка задачи. Определения 61
2.2.2 Генерация ландшафта с заданной конфигурационной энтропией 62
2.2.3 Генерация ландшафта с заданным Total Edge 65
2.2.4 Генерация «естественных» ландшафтов 67
2.2.5 Взаимосвязь энтропии и Total Edge 68
2.3 Метод формального описания группового движения и конфликта агентов в содержащей препятствия среде 70
2.3.1 Непрерывная постановка задачи 70
2.3.2 Постановка задачи в терминах клеточных автоматов 73
2.3.3 Определение клеточного автомата 74
2.3.4 Состояния и алфавит 75
2.3.5 Входы и выходы 76
2.3.6 Сущности, необходимые для моделирования конфликта и системы связи 78
2.3.7 Определение весов, необходимых для моделирования движения 81
2.3.8 Описание функционирования автомата 85

Стр.

2.3.9 Определение функций локального тактового функционирования ОСВВ. 86
2.4 Моделирование рефлексии агента 87
2.4.1 Общее описание клеточного автомата, моделирующего агентов с памятью 87
2.4.2 Граница 94
2.4.3 Функция перехода субъективной реальности 94
2.4.4 Автоматы и теория категорий 95
2.5 Алгоритмы организации строя агентов 96
2.5.1 Метрика сходства графов и метрика, порожденная графом . 97
2.5.2 Синтез локальной функции перехода 98
2.5.3 Результаты симуляции 102
2.5.4 Псевдокод используемых алгоритмов 104
2.6 Методы тестирования алгоритмов группового движения агентов с использованием наборов случайных ландшафтов 107
2.6.1 Сравнение квазиоптимальных оптимальных траекторий . . 107
2.6.2 Движение агента в точку назначения как реализация броуновского моста 110
2.6.3 Выявление зависимости статистических свойств квазиоптимального маршрута от характеристик проходимой местности 111
2.6.4 Аналитический подход к нахождению распределения времени выхода на границу 113
2.6.5 Вычислительный эксперимент для времени выхода в конечную точку 116
2.6.6 Сравнение аналитически полученной функции распределения и экспериментальных данных 118
2.6.7 Вычислительный эксперимент для поиска распределения координат местоположения агента 119
2.7 Модель типа уравнения конвекции-диффузии для движения агентов 122
2.8 Методы квзиоптимального распределения авионики на основе клеточного автомата и муравьиного алгоритма 129
2.8.1 Постановка задачи 129

Стр.

2.8.2 Задача минимизации длины кабеля 131
2.8.3 Размещение устройств по местам установки 134
2.8.4 Назначение кабельных трасс 134

Глава 3. Модели и методы построения самоорганизующейся системы связи мобильных агентов 137
3.1 Общая схема модели системы связи 137
3.1.1 Ячейки и состояния 138
3.1.2 Граф системы связи 139
3.1.3 Модель передачи информации 143
3.2 Демонстрация работы алгоритма 147
3.2.1 Задачи, для решения которых проводится моделирование. . 153
3.3 Алгоритмы автоматической организации сети связи мобильных агентов 154
3.3.1 Введение 154
3.3.2 Постановка задачи 157
3.3.3 Недостатки модели 160
3.3.4 Распределение ресурсов в зависимости от роли агента и в условиях недостатка ресурсов 163
3.3.5 Структурное сходство 165
3.3.6 Структурирование графа потребностей как задача кластеризации 168
3.3.7 Пример системы связи, основанной на структурном подходе 172
3.3.8 Выводы и перспективы 175
3.4 Моделирование начального распределения частот 176
3.4.1 Введение 176
3.4.2 Клеточно-автоматная модель 177
3.4.3 Вычислительный эксперимент 181
3.4.4 Изменение прямой видимости агентов, движущихся по пересеченной местности 186
3.5 Алгоритмы организации защищенной сети связи 187
3.5.1 Система автоматической настройки беспроводной сети радиостанций 187
3.5.2 Предлагаемый алгоритм настройки радиосети 188

Стр.

3.5.3 Модель системы настройки радиосети специального назначения . . . 192
3.5.4 Моделирование преднамеренной помехи . . . 194
3.5.5 Имитационное моделирование системы настройки радиосети специального назначения . . . 195

Глава 4. Непрерывная постановка задачи движения агента по оптимальному по времени маршруту и связанные задачи . . . 198
4.1 Введение . . . 198
4.2 Непрерывная модель . . . 199
4.2.1 Тривиальный случай . . . 199
4.2.2 Движение агента в произвольной ограниченной области с непрерывной кусочно-гладкой границей . . . 201
4.3 Дискретная модель . . . 210
4.3.1 Оценки аппроксимации . . . 210
4.3.2 Построение клеточного автомата . . . 217
4.4 Задача многоагентного поиска оптимального пути . . . 220
4.4.1 Постановка задачи . . . 220
4.4.2 Маршруты и функционалы качества . . . 222
4.4.3 Существование оптимального решения для одного агента . 224
4.4.4 Оптимальное решение для всех агентов . . . 231
4.4.5 Заключение . . . 233

Глава 5. Программная среда многоагентного моделирования «Психоход» . . . 234
5.1 Модель движения . . . 234
5.2 Модель системы связи . . . 236
5.3 Соединение моделей движения и связи в среде «Психоход» . . . 240

Заключение . . . 243

Список литературы . . . 245

Список рисунков . . . 264

Список таблиц 268

Введение

Актуальность темы исследования. Технология многоагентных систем – это активно развивающееся научное направление, которое в настоящее время находится в стадии становления, но уже активно используется для решения сложных прикладных задач, допускающих декомпозицию на отдельные задачи, которые назначаются агентам. Последующая композиция найденных агентами частных решений позволяет определить решение глобальной задачи.

В настоящем исследовании под термином *агент* понимается реальная или виртуальная автономная сущность, функционирующая во внешней динамической среде, способная воспринимать состояние этой среды и взаимодействовать с другими агентами с помощью некоторого языка коммуникаций, проявляя независимое поведение, обусловленное знаниями и ментальными свойствами.

В широком смысле *многоагентная система* представляет собой модель, которая имитирует одновременные действия и взаимодействия нескольких *агентов* в попытке воссоздать течении некоторого процесса или состояние моделируемой сложной системы.

Можно выделить несколько направлений активных исследований в области теоретических основ многоагентных систем: формализация и моделирование основных понятия и компонентов, в частности, понятия агента и его ментальных свойств; организационное моделирование многоагентных систем на основе общей теории систем и организаций; разработка процедур обучения и самообучения агентов и многоагентной системы в целом; поддержка сетевого взаимодействия на основе обеспечение мобильности агентов и их согласованного поведения в различных средах; развитие агентно-ориентированного программирования, а также технологий параллельных и распределенных вычислений именно для многоагентных систем. Теоретическую базу для разработки многоагентных систем составляют подходы, основанные на сетях Петри; событийных моделях и имитационном моделировании; игровых, марковских моделях; J-сетях; клеточных автоматах.

В последнее время многоагентные системы широко используется для решения разнообразных задач: коллективное управление, управление различными процессами (логистика, дорожное движение) и динамическими объектами со

сложным поведением (например, управление орбитальной группировкой космических аппаратов); задач моделирования организованного и хаотического движения (перемещение военного строя, движение людей при катастрофах и панике, перемещение клеток в кровеносных сосудах и т.п.), социально-экономических отношений, рисковых ситуаций, динамики онлайн-дискуссий в сети интернет и др.; задач принятия решений в условиях неопределенности; задач, связанных с обработкой информации (организация сетевых вычислений, обеспечение устойчивости функционирования распределенных информационных систем, оптимизация параллельной обработки информации).

Продвижение технологии многоагентных систем в область практических приложений – это актуальная научно-техническая проблема, решение которой позволит повысить степень обоснованности принимаемых решений в социальных, экономических и технических системах; качество и адекватность моделей различных процессов перемещения дискретных объектов; эффективность функционирования технических и иных сложных систем, в частности, мобильных роботов, беспилотных летательных аппаратов и других интеллектуальных устройств при их групповой эксплуатации.

Одним из приложений технологии многоагентных систем является когнитивное радио – перспективное направление разработки радиосистем. Согласно отчету Сектора радиокоммуникаций Международного союза электросвязи SM.2152 Definitions of Software Defined Radio (SDR) and Cognitive Radio System (CRS), *когнитивное радио* – это система радиосвязи, способная получать сведения об особенностях собственной эксплуатации и на основе этих знаний корректировать параметры своей работы. Различные проблемы, связанные с адаптацией радиосети к изменяющимся помеховым условиям, связаны с самоорганизаций радиостанций на основе *когнитивной радиосети* (cognitive network), в которой все узлы обладают как «осознанием» изменений частотного спектра (spectrum sensing), что обусловлено свойствами когнитивного радио, так и общей «плоскостью знаний» (knowledge plane) сети, которая содержит сведения о типе, предназначении и состоянии узлов сети. В такой сети узлы согласованно занимают или освобождают те или иные частотные каналы в зависимости от своего положения в иерархии узлов, изменений помеховой обстановки или других обстоятельств на основе динамического управления спектром. Данная технология активно используется в разрабатываемых сейчас за рубежом перспективных системах военной связи (см., например, работы J. Redi и R. Ramanathan).

Диссертационная работа посвящена разработке теоретических основ многоагентного моделирования иерархических систем на основе клеточных автоматов.

Работа выполнена в рамках одного из основных научных направлений Воронежского государственного университета «Математическое моделирование, программное и информационное обеспечение, методы вычислительной и прикладной математики и их применение к фундаментальным исследованиям в естественных науках».

Степень разработанности темы исследования.

Многоагентному моделированию посвящено множество работ отечественных и зарубежных ученых: G. Antonelli, K. Batool, F. Bullo, M.A. Niazi, B. Beer, C.W. Reynolds, C. Tsatsoulis, L.-K. Soh, P. Stone, M. Veloso, A. Rousset, B. Herrmann, C. Lang, L. Philippe, A.S. Rao, M.P. Georgeff, R. Tempo, K.-K. Oh, M.-C. Park, H.-S. Ahn, L. Merghem, H. Lecarpentier, H. Ma, T. K. S. Kumar, S. Koenig, T. Leppänen, M. Liu, E. Harjula, A. Ramalingam, J. Ylioja, P. Narhi, J. Riekki, T. Ojala, K. Kravari, N. Bassiliades, A. Kesting, M. Treiber, D. Helbing, Z. Kan, L. Navaravong, J. M. Shea, E. L. Pasiliao, W. E. Dixon, J. Jones, R. Mayne, C. A. Iglesias, M. Garijo, J. C. González, J. P. Müller, M. P. Singh, L. Hamill, N. Gilbert, T. J. Gordon, E. Frias-Martinez, G. Williamson, V. Frias-Martinez, J. Ferber, A. J. Ellery, R. E. Baker, S. W. McCue, M. J. Simpson, R. Dutta, L. Sun, M. Kothari, R. Sharma, D. J. Pack, R. H. Bordini, L. Braubach, M. Dastani, A. Pokahr, A. Ricci, M. A. Niazi, O. Amir, G. Sharon, R. Stern, S. M. A. Abbas, В.И. Городецкий, В.Б. Евгенев, Д.А. Новиков, И.А. Каляев, С.Э. Парсегов, А.В. Проскурников, А.С. Матвеев, И.Д. Зайцев, О.В. Карсаев, Д. Будаев, К. Амелин, Г. Восчук, П. Скобелев, Н. Амелина, А. Борщев, А. Филиппов и других.

Выбор теории клеточных автоматов для развития теории многоагентных систем в диссертации обусловлен их широким использованием, что подтверждается работами Г. Г. Малинецкого, М. Е. Степанцова (модели движения групп людей), I. G. Papadopoulos, G. Ch. Sirakoulis (модели социальных систем), R. Mead, J.B. Weinberg, (передвигающиеся строем роботы), A. Adamatzky, A. Schumann (биологические и химические компьютеры), L. Johnson, G. N. Yannakakis, J. Togelius (процедурная генерация поверхностей), A. Ilachinski, G. C. McIntosh, D. P. Galligan (модели боевых действий) и других.

Объектом исследования являются иерархические системы мобильных агентов в динамической среде.

Предметом исследования являются закономерности функционирования иерархических систем мобильных агентов в динамической среде, а также модели, методы и алгоритмы управления и принятия решений агентами в динамической среде.

Цель диссертационной работы заключается в развитии теории и технологий моделирования иерархических систем коммуницирующих мобильных агентов, перемещающихся в динамической среде с препятствиями, для повышения качества моделей поведения агентов и эффективности управления такими системами.

Для достижения цели решались следующие **задачи**:

1. Анализ существующих подходов к многоагентному моделированию сложных систем на основе клеточных автоматов.
2. Синтез моделей ландшафта с динамическими характеристиками.
3. Разработка дискретной модели группового движения и взаимодействия систем интеллектуальных мобильных агентов с памятью на основе клеточных автоматов.
4. Разработка и исследование непрерывной модели группового движения агентов.
5. Разработка децентрализованной модели системы связи, включая модель канала связи агентов.
6. Разработка структуры программного комплекса, а также математического, алгоритмического и программного обеспечение для реализации многоагентной системы иерархически организованных, коммуницирующих мобильных агентов, перемещающихся в динамической среде с препятствиями.
7. Проведение вычислительного эксперимента для апробации предложенных моделей, анализ результатов и разработка рекомендаций по использованию многоагентной системы для практических приложений (когнитивное радио).

Научная новизна.

В диссертации получены следующие результаты, характеризующиеся научной новизной:

1. Для формализованного описания среды впервые понятие ландшафта и его компонентов введено как множество классов клеток клеточного

автомата с определенным набором свойств. Предложены методы генерации случайных ландшафтов с заданными характеристиками. При этом каждой клетке ландшафта ставится в соответствие максимально возможная скорость пересечения клетки, что позволяет оценить эффективность алгоритмов движения агента в заданной среде.

2. Предложена модель движения мобильных агентов в форме клеточного автомата, отличающаяся формализованным описанием среды с помощью понятия ландшафта и позволяющая на единой основе разрабатывать алгоритмы поведения агентов (в том числе организованное) в динамической среде с препятствиями.
3. Разработан оригинальный метод формализованного описания группового движения и конфликта системы агентов, соединенной с моделью системы связи, отличающийся от других подобных многоагентных моделей тем, что в ней как движение агентов, так и организация системы связи моделируется двумя сопряженными клеточными автоматами и тем, что местность содержит препятствия и может меняться в процессе движения.
4. Получены ранее не исследованные зависимости характеристик движения агентов и ландшафтных метрик. Предложена методика проведения экспериментов, позволяющих описывать закономерности движения агентов к заданной цели, отличающаяся использованием числовых характеристик количества и расположения препятствий типа конфигурационной энтропиии.
5. Предложена оригинальная непрерывная нелинейная модель движения агента по пересеченной местности (в виде задачи оптимального управления с нелинейными дифференциальными ограничениями). Указанная модель отличается от известных тем, что не требует гладкости траекторий агента и функции, описывающей распределение препятствий по *ландшафту*. Данная модель позволяет исследовать агента, движущегося по параметрически заданным траекториям с «изломами». Показано, что непрерывная нелинейная модель движения агента является предельным случаем дискретной модели.
6. Предложен метод самоорганизации системы мобильных агентов, располагающих набором каналов связи, на основе обмена маячками и

поочередного сканирования каналов в несколько сетей связи. Показано, что самоорганизация описывается моделью сегрегации Шеллинга типа II. Указанный алгоритм исследован с помощью вычислительного эксперимента, определены условия его сходимости.

7. Разработана структура программного комплекса «Психоход», его алгоритмическое обеспечение и программная реализация, основными функциональными возможностями которого являются следующие: моделирование агентов, их движения, взаимодействия; моделирование и анализ многоагентной системы; визуализация функционирования многоагентной системы. В отличие от аналогов типа NetLogo разработанный программный комплекс «Психоход» отличается быстродействием за счет оптимизации параметров и адаптации системы путем настройки на конкретные задачи исследования.

Теоретическая и практическая значимость. Теоретическая значимость заключается в развитии теории и разработке новых технологий моделирования для многих задач, связанных с движением и коммуникацией агентов. Поскольку в приложениях, как правило, известны лишь дискретные наборы данных, характеризующих движение агента, и управление агентом происходит тоже в дискретном времени, то использование дискретных моделей, на взгляд автора, более оправдано (например, в плане вычислительной сложности), чем реконструкция непрерывной модели движения по дискретным данным с последующей дискретизацией полученного решения. Описание системы агентов в виде ограниченного набора правил и графов позволяет моделировать движение и связь системы из значительного количества агентов без особых вычислительных трудностей.

Результаты диссертации ориентированы на непосредственное практическое применение. Они могут использоваться

– для системного подхода в анализе проведения и планирования локальных военных операций, например, антитеррористических, и для разработки соответствующих систем поддержки принятия решений;
– при разработке систем когнитивного радио, что подтверждается патентами на изобретение, которые были использованы в производстве АО «Концерн «Созвездие», № 2644523, №2549120 и актом об использовании изобретения от 23 сентября 2015 г.;
– для проектирования систем интегрированной модульной авионики;

– для проведения вычислительного эксперимента для задач управления строем роботов или беспилотных летательных аппаратов и отработки алгоритмов управления такими объектами;
– для моделирования телекоммуникационного трафика в небольших сетях подвижных агентов, движущихся по местности, на которой имеется большое количество препятствий, затрудняющих связь (свидетельства о регистрации программы для ЭВМ № 2016615934, № 2017616905).

Предложенные модели взаимодействия агентов можно использовать для разработки систем искусственного интеллекта и приложений, которые с ним связаны. Результаты диссертации используются в учебном процессе ФГБОУ ВО «Воронежский государственный университет» при чтении спецкурсов («Технологии искусственной жизни», «Основы агентных систем») и выполнении курсовых проектов и выпускных квалификационных работ.

Методология и методы исследования. В диссертации для моделирования движения, взаимодействия и связи агентов применялась методология системного анализа и теории систем, в частности, модели клеточных автоматов и многоагентные модели.

Говоря более точно, пространство, в котором происходит взаимодействие агентов, является графом, перемещение агентов возможно только между вершинами графа за дискретное время, сам агент является конечным автоматом.

Для решения отдельных задач задач построения строя агентов применялся аппарат теории графов и топологии (в части построения метрики несходства графов).

Для выяснения взаимосвязи получившейся дискретной клеточно-автоматной модели и известных непрерывных моделей движения, применялись методы функционального анализа и теории дифференциальных уравнений с разрывной правой частью.

Для исследования вероятностных свойств алгоритмов использовался аппарат математической статистики – критерии ω^2, метод максимального правдоподобия и другие.

При разработке программного комплекса использовались технологии объектно-ориентированного и акторного программирования.

Основные положения и результаты, выносимые на защиту:

1. Совокупность методов построения и тестирования алгоритмов группового движения агентов с использованием наборов случайных *ландшафтов* с фиксированными характеристиками (*соответствует п. 2 «Формализация и постановка задач системного анализа, оптимизации, управления, принятия решений и обработки информации», п. 11 «Методы и алгоритмы прогнозирования и оценки эффективности, качества и надежности сложных систем»*).
2. Метод формального описания группового движения и конфликта агентов по пересеченной местности, по которой случайным образом разбросаны препятствия разной проходимости, основанный на клеточном автомате (*соответствует п. 4 «Разработка методов и алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации»*).
3. Теорема о том, что кратчайший по времени путь агента в виде ломаной, полученный с помощью клеточного автомата, является приближенным решением непрерывной оптимизационной задачи поиска кратчайшего по времени пути в области, в каждой точки которой задано ограничение на максимальный модуль скорости агента (*соответствует п. 3 «Разработка критериев и моделей описания и оценки эффективности решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации»*).
4. Оценка точности приближения пути агента, зависящая от ширины клетки (*соответствует п. 3 «Разработка критериев и моделей описания и оценки эффективности решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации»*).
5. Клеточный автомат, моделирующий сеть связи движущихся по местности с препятствиями иерархически организованных агентов (*соответствует п. 4 «Разработка методов и алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации»*).
6. Алгоритм автоматической организации сетей связи (*соответствует п. 3 «Разработка критериев и моделей описания и оценки эффективности решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации», п. 4 «Разработка методов и*

алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации»).

7. Программная среда «Психоход», в которой были произведены все исследования диссертации (*соответствует п. 5 «Разработка специального математического и программного обеспечения систем анализа, оптимизации, управления, принятия решений и обработки информации», п. 12 «Визуализация, трансформация и анализ информации на основе компьютерных методов обработки информации»*).

Достоверность и обоснованность полученных в работе результатов и выводов подтверждается корректным применением математического аппарата; сопоставлением результатов теоретических исследований и экспериментальных данных с опубликованными результатами других исследователей; положительными результатами проведенных экспериментов и опытом практической эксплуатации разработанной сети связи, что подтверждается актами о внедрении. Код программного обеспечения, которое использовалось для получения экспериментальных данных, размещен в общедоступном репозитории `https://bitbucket.org/bokohodteam/bokohod`.

Апробация работы. Основные результаты работы докладывались в 2015–2018 гг. на международных конференциях «Радиолокация, Навигация, Связь» (RLNC-2016, RLNC-2018, г. Воронеж), «Математические методы в технике и технологиях» (ММТТ-30, г. С.-Петербург), «Воронежская Зимняя Математическая Школа» (ВЗМШ-2016, г. Воронеж), «Актуальные проблемы прикладной математики, информатики и механики» (АППМИМ-2015, АППМИМ-2017, г. Воронеж), «Информационные технологии и нанотехнологии» (ITNT-2017, ITNT-2018, г. Самара), «Современные сложные системы управления» (СССУ-2017, г. Липецк), BICT 2017 – 10th EAI International Conference on Bio-Inspired Information and Communications Technologies (организатор – European Alliance for Innovation, г. Хобокен, Нью-Джерси, США), 27th International Conference in Operator Theory (2018 г., Западный университет Тимишоара, г. Тимишоара, Румыния), «Социофизика и социоинженерия’2018» (ИПУ РАН, г. Москва). Результаты диссертации обсуждались на семинаре «Теория управления организационными системами», проводимом в Институте проблем управления им. В.А. Трапезникова РАН (2017–2018 гг.), а также на ежегодных научных сессиях Воронежского государственного университета (2016–2019 гг.).

Личный вклад. В диссертации представлены теоретические результаты, выполненные лично автором или под его непосредственным руководством. Экспериментальные данные получены с помощью программного комплекса, разработанного автором, или при его участии. В научных работах, выполненных в соавторстве, личный вклад соискателя состоит в следующем: [5] – теоретические результаты, касающиеся метрики несходства графов и ее приложениям; [3;14;15;24;33;34;36] – постановка задач исследования, теоретическое обоснование метода решения задачи (при участии соавторов, при этом результаты соавторов не относятся к положениям, выносимым на защиту), разработка программного обеспечения; анализ результатов эксперимента и формулировка выводов; [18] – разработка метода решения задачи, экспериментальное исследование и формулировка выводов.

Публикации. Основные результаты по теме диссертации изложены в 19 статьях в ведущих рецензируемых журналах, рекомендованных ВАК, которые либо индексируются в базах данных Web of Science: Core Collection [1–6], базах данных Scopus [7], базах данных MathSciNet: [8; 9], либо в иных журналах, которые входят в перечень рецензируемых научных изданий ВАК Минобрнауки России: [10–19], а также в 13 докладах на международных конференциях [20–23] (индексируются Scopus) и [24–32]. Получено два патента на изделия, реализующие алгоритмы [33; 34], два свидетельства о государственной регистрации программы для ЭВМ [35; 36].

Объем и структура работы.

Диссертация состоит из введения, пяти глав, заключения и приложений.

Полный объём диссертации составляет 268 страниц, включая 65 рисунков и 7 таблиц. Список литературы содержит 157 наименований.

Краткое содержание диссертации.

Автором была разработана модель движения и взаимодействия системы агентов, которая представляет инструментарий, позволяющий моделировать поведение сложно организованных групп на местности с заданным количеством случайных препятствий разных типов. С математической точки зрения, модель представляет собой два связанных между собой клеточных автомата. Можно считать эти автоматы как двумерными решетками, по которым перемещаются автономные агенты (т. н. World-state cellular automaton), так и одномерными автоматами, в которых окрестность задается исходя из необходимости поддержания строя между агентами (Robot-state cellular automaton). В первом случае ячейке

автомата в определенной мере соответствует квадрат на местности, в котором может находиться агент, препятствие и т. п., во втором – ячейкой является сам агент, а в параметры ячейки входят текущие координаты агента, свойства местности, в которой находится агент и т.п.

Концептуально модель состоит из подсистем, показанных на рис. 0.1.

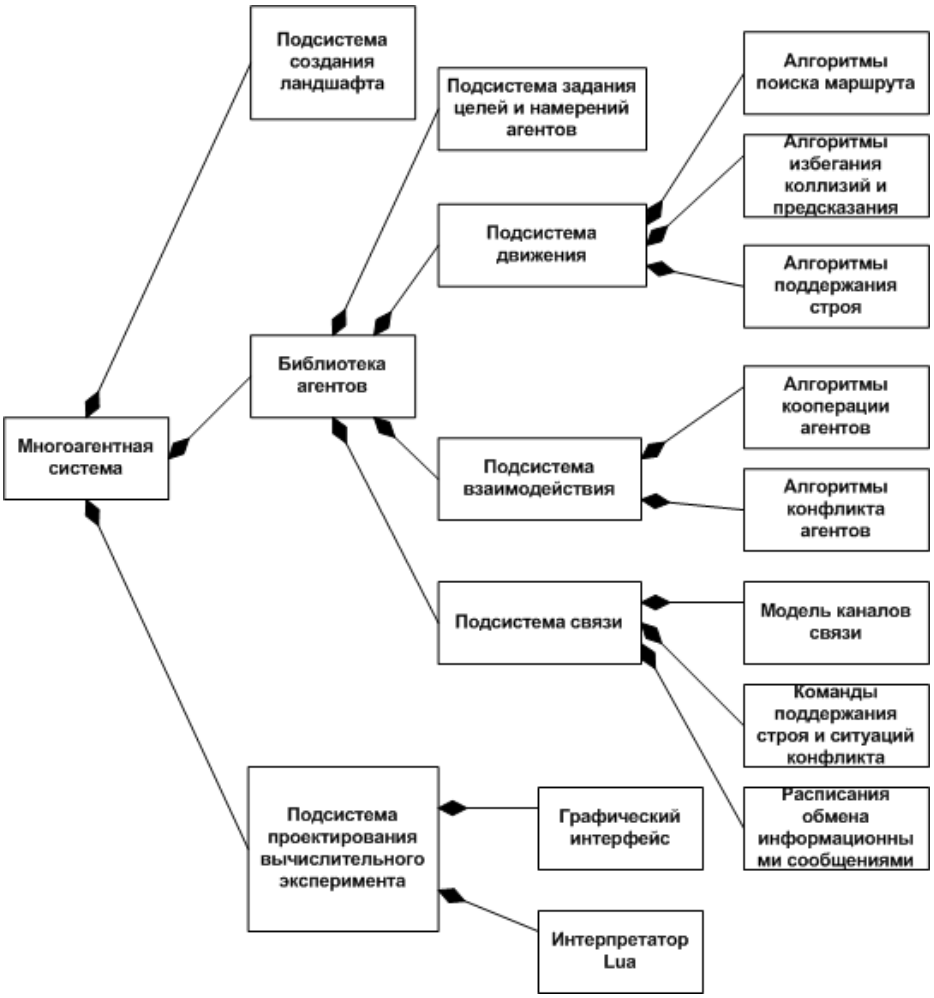


Рисунок 0.1 — Схема многоагентной системы

Перечислим подсистемы, показанные на схеме подсистем модели на на рис. 0.1.

Библиотека агентов – это совокупность алгоритмов, моделирующих поведение и свойства агентов, включая наборы заранее заготовленных агентов с заданными свойствами. В нее входят подсистемы движения, взаимодействия и связи агентов, а также подсистема управления целями и намерениями агентов. В подсистему движения (которая основана на клеточном автомате), входят:

1. Алгоритмы поиска наиболее быстрого маршрута – включая стандартные алгоритмы поиска на графах, типа алгоритма Дейкстры, и разные

эвристические алгоритмы применяемые на не вполне разведанных агентом местностях.

2. Алгоритмы избегания коллизий агентов, которые необходимы для обеспечения пропуска агентами друг друга в узких местах и алгоритмы предсказания поведения агентов, которые нужны для организации строя и предупреждения скучивания агентов в одном месте.
3. Алгоритмы поддержания строя, которые позволяют агентам самоорганизовываться в определенные формации.

В подсистему взаимодействия агентов входят

1. Алгоритмы кооперации агентов по разведке территории, в которых агенты при принятии решений руководствуются информацией, полученной другими агентами.
2. Механизмы конфликта агентов разных типов, влекущие уничтожение противоборствующих агентов.

Подсистема связи агентов состоит из

1. Расписаний обмена информационными сообщениями – для моделирования сетевого трафика, агенты рассылают сообщения в установленные моменты времени.
2. Передачи команд, предназначенных для поддержания строя и обработки ситуаций конфликта – агенты призывают себе помощь, подают сигнал отступления и т.п.
3. Модели каналов связи, в которой моделируется потеря пакетов данных под действием помех.

Подсистема создания *ландшафта* состоит из различных алгоритмов, предназначенных для случайной расстановки препятствий и моделирования местностей с заданными свойствами. Подсистема проектирования вычислительного эксперимента состоит из различных средств автоматизации работы с агентами, включая модуль интерпретатора скриптового языка Lua и графический интерфейс.

Следует уточнить, что в предлагаемой модели присутствуют несколько видов агентов и несколько типов взаимодействия. Агенты (см. рис. 0.2) могут быть как абстракциями неких сущностей (агенты I рода), движущихся по «географической» местности, так и принадлежащими этим сущностям абстракциями

средств связи (агенты II рода). На рисунке показаны группы противоборствующих агентов I рода. Видно, что один агент I рода может «обладать» несколькими агентами II рода.

Взаимодействие агентов I рода включает следующие типы:

1. Агенты могут, намеренно или нет, блокировать движение других агентов.
2. Агенты могут уничтожать друг друга, руководствуясь определенной внутренней логикой.
3. Агенты призывают союзных агентов к себе в случае конфликта.
4. Агенты обмениваются сообщениями, имитирующими трафик телекоммуникационной сети.

Взаимодействие агентов II рода включает два типа:

1. Построение сети связи в соответствии с заданным образцом.
2. Самоорганизация сети связи по типу сегрегации (модель Шеллинга типа II).

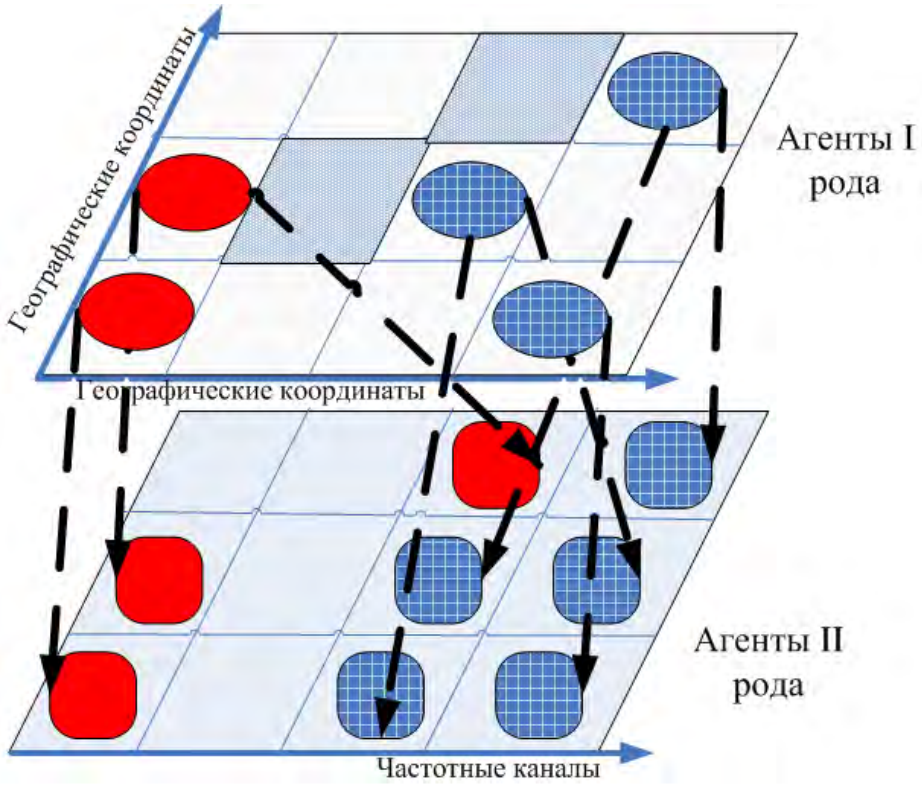


Рисунок 0.2 — Слои взаимодействия агентов

Агенты обладают внутренней памятью и являются рефлексивными BDI агентами. Иначе говоря, каждый агент хранит внутри себя сведения об известных ему свойствах местности, постоянно обновляя их, и о движении других агентов. Восприятие агентом обстановки может быть искажено под влиянием

тех или иных факторов. У каждого агента задано множество целей и есть набор алгоритмов для их достижения.

В **первой главе** произведен обзор существующих работ по многоагентным системам, которые так или иначе повлияли на работу автора. Рассматриваются работы общего характера, описывающие основные направления многоагентного моделирования:

1. моделирование индивидуального и коллективного поведения агентов в многоагентной системе;
2. теоретико-игровое моделирование взаимодействия агентов;
3. лингвистическое направление, связанное с моделированием речевых актов для построения протоколов коммуникации между агентами;
4. организационное моделирование многоагентных систем, связанное с общей теорией систем и организаций;
5. построение моделей распределенных систем на базе сетей Петри;
6. программистское направление, связанное с моделями акторов, параллельным и агентно-ориентированным программированием;
7. логико-семиотическое направление, связанное с моделированием ментальных свойств (интенциональных характеристик) агентов с помощью неклассических логик;
8. сетевые системы дифференциальных уравнений и их дискретные аналоги.

Также приводятся применения многоагентных систем и близких к ним абстракций, в особенности функционирующих в дискретном пространстве и времени, в математике, механике, биологии и медицине, неклассических вычислениях (т.е. в вычислениях на основе биологических, химических и т.п. компьютерах), транспорте, связи. Особо затрагиваются социально-экономические приложения многоагентных систем: моделях движения толпы, боевых действий, экономические модели.

Во **второй главе** вводится понятие *ландшафта*, рассматриваются способы генерации тестовых *ландшафтов*, исследуются различные характеристики *ландшафтов* и их зависимости. Определяется модель движения агента по пересеченной местности и описываются клеточные автоматы, моделирующие этот процесс, а также децентрализованное построение и поддержание строя агентов во время движения. Исследуются характеристики автоматов.

Определение 1. *Однородной структурой со входами и выходами (далее – ОСВВ, клеточным автоматом), называется набор* $(\mathbb{Z}^k, \mathscr{W}, V)$, *где* $\mathscr{W}$ *– конечный автомат с* n *основными и* p *боковыми входными каналами,* m *основными и* p *боковыми выходными каналами,* l *состояниями, помещенный в каждом узле целочисленной решетки* $\mathbb{Z}^k$ *и задаваемый системой уравнений:*

$$\begin{cases} y(t) = \varphi_1(x(t), b(t), \mathfrak{s}(t)), \\ a(t) = \varphi_2(\mathfrak{s}(t)), \\ \mathfrak{s}(t+1) = \varphi(x(t), b(t), \mathfrak{s}(t)), \end{cases}$$

$$\varphi_1 : \mathscr{E}^n \times \mathscr{E}^p \times \mathscr{S} \to \mathscr{E}^m,$$

$$\varphi_2 : \mathscr{S} \to \mathscr{E}^p,$$

$$\varphi : \mathscr{E}^n \times \mathscr{E}^p \times \mathscr{S} \to \mathscr{E}^m,$$

где $x = (x_1, \ldots, x_n)$, $b = (b_1, \ldots, b_p)$, $a = (a_1, \ldots, a_p)$, $y = (y_1, \ldots, y_m)$ *обозначают соответственно основной вход, боковой вход, боковой выход и основной выход,* $\mathfrak{s}$ *– состояние автомата,* $t = \overline{0,N}$ *– такт дискретного времени,* $\mathscr{E}$ *– входной и выходной алфавит,* $\mathscr{S}$ *– множество состояний автомата* $\mathscr{W}$, $V = (\alpha_1, \ldots, \alpha_p)$, $\alpha_i \in \mathbb{Z}^k$, $i = \overline{1,p}$, *– шаблон соседства: для автомата в узле* α *каждый* $\alpha_i \in V$ *определяет конечный автомат с координатами* $\alpha + \alpha_i$, *с* $i-$*м боковым выходным каналом которого соединен* $i-$*й боковой входной канал автомата в узле* α.

Агентной функции соответствует здесь локальная функция перехода клеточного автомата, восприятию – боковой и основной входы, а действию агента – боковой и основной выходы клеточного автомата.

Определение 2. *Назовем ландшафтом (см. рис. 2.2) в момент времени* t *множество классов клеток определенного подмножества замощения* $\mathcal{L}(\Omega_h, l) = \{u_{ij} | u_{ij} = \mathrm{u}(t, \omega_{ij}), \omega_{ij} \in \Omega_h\}$, *такого что* u *принимает на* $\{\omega_{ij} | i = \overline{1,n}, j = \overline{1,m}\}$ *не более* l *значений, причем к классу* i *принадлежит* N_i *клеток, то есть* $\sum_{i=1}^{l} N_i = M$, $M = |\mathcal{L}(\Omega_h, l)|$.

Определение 3. *Total Edge (TE) определяется как общее количество соприкосновений сторон клеток в* $\mathcal{L}(\Omega_h, l)$, *принадлежащих к разным классам. Будем далее обозначать Total Edge ландшафта* $\mathcal{L}(\Omega_h, l)$ *как* $TE(\mathcal{L}(\Omega_h, l))$.

Определение 4. *Total Edge Density (TED) для ландшафта $\mathcal{L}_{n\times m}(l)$ определяется как отношение $TE(\mathcal{L}(\Omega_h, l))$ к общему количеству клеток $\mathcal{L}(\Omega_h, l)$:*

$$TED(\mathcal{L}(\Omega_h, l)) = TE(\mathcal{L}(\Omega_h, l))/M.$$

Отметим, что для *ландшафтов*, генерируемых для тестирования алгоритмов обхода препятствий и поиска кратчайшего пути, клетки *ландшафта* будут делиться на классы по максимально возможной скорости пересечения клетки.

В ходе проведенных вычислительных экспериментов выяснилось, что практически среднее значение TED не зависит от количества клеток в *ландшафте*, а зависит лишь от количества классов клеток. Из эмпирических соображений будем искать зависимость среднего значения $TED(\mathcal{L}(n \times m,l))$ от l и конфигурационной энтропии S в форме

$$\overline{TED} = a_1(l)\sigma(Sb_1(l)/\ln l) + a_1(l)/2,$$

где $\sigma(x) = (1 + e^{-x})^{-1}$, a_1, b_1 – неизвестные функции. Подбирая параметры по экспериментальным данным, можно получить (с коэффициентами детерминации $r^2 = 0.999999$ для a_1 и $r^2 = 1.0$ для b_1) аппроксимации

$$a_1(l) = 3.90302 + 5.25218(x - 10.9641)^{-2.19235},$$
$$b_1(l) = 1.15228\ln(0.990096x - 1.18397).$$

Динамика концентрации агентов на местности с препятствиями.

Автор разработал методику проведения экспериментов, позволяющих описывать закономерности движения агентов к заданной цели в зависимости от численных характеристик количества и расположения препятствий типа конфигурационной энтропии, а также выявил ряд таких закономерностей.

Например, изучается зависимость среднего числа агентов от времени и положения на *ландшафте*. Генерируется 150 случайных *ландшафтов* с заданной энтропией по соответствующему алгоритму. Группа, образованная из $u_0 = 48$ агентов, перемещается с одной стороны квадратного *ландшафта* в противоположную. Вычисляется зависимость среднего (по всем генерируемым *ландшафтам*) числа агентов c на x-й строке *ландшафта* от дискретного времени t, $x = \overline{1, x_{\ max}}$, $x_{\max} = 48$ и выясняется, что эта зависимость определяется, главным образом, не видом конкретного *ландшафта*, а конфигурационной энтропией *ландшафта* S.

Выявлено, что такая «концентрация» агентов c может быть описана системой уравнений типа уравнений конвекции-диффузии

$$\frac{\partial c}{\partial t} = C_1 \frac{\partial^2 c}{\partial x^2} + C_2 \frac{\partial c}{\partial x},$$

где C_1, C_2 – некоторые рациональные функции от x, $x^{1/2}$, t, $t^{1/2}$, и получены зависимости параметров таких уравнений от конфигурационной энтропии *ландшафта*, по которому перемещаются агенты.

Начальные условия задаются как

$$c(x,0) = c_0 \delta(x-1),$$

$$\delta(x) = \begin{cases} 1, & x = 0, \\ 0, & x \neq 0, \end{cases}$$

а асимптотическое условие записывается как

$$\lim_{t\to\infty} c(x,t) = c_0 \theta(x - x_{\max}),$$

$$\theta(x) = \begin{cases} 1, & x \geqslant 0, \\ 0, & x < 0. \end{cases}$$

Статистические свойства квазиоптимальных маршрутов.

Движение агента по случайному *ландшафту* можно рассматривать как реализацию броуновского моста. Существуют работы, где рассматривалась модель движения броуновского моста, которая обеспечивает биологически обоснованное приближение пути движения животного на основе дискретных данных местоположения и является мощным методом количественной оценки распределения использования (utilization distribution, т.е. распределения, сконструированного из данных о местоположении животного в пространстве в различные моменты времени). В данном разделе исследуются статистические свойства квазиоптимальных траекторий агента, получаемых с помощью предложенного в диссертации клеточного автомата. Эти траектории также интерпретируются как реализации броуновского моста.

Исследованы вероятностные характеристики получаемого квазиоптимального маршрута в зависимости от количества и характера расположения препятствий. Были определены функции плотности распределения времени прихода в

конечную точку маршрута

$$PDF_\zeta(z) = \frac{e^{-\frac{\nu^2(t)}{2\sigma^2(t)}}}{\sqrt{2}} \sum_{k=0}^{\infty} \frac{\nu^{2k}(t)}{2^k (k!)^2 \sigma^{2k+1}(t)} \left(\Gamma\left(k+\frac{1}{2}, \frac{z^2}{2b^2\sigma^2(t)}\right) - \Gamma\left(k+\frac{1}{2}, \frac{z^2}{2\sigma^2(t)}\right)\right)$$

и плотности распределения отклонения координат маршрута от оптимального (extreme value distribution)

$$evd(\alpha(t), \beta(t); x) = \frac{e^{\frac{\alpha(t)-x}{\beta(t)} - e^{\frac{\alpha(t)-x}{\beta(t)}}}}{\beta(t)}.$$

Здесь α, β, ν, σ, b – параметры функций плотности, определяющей характер распределения препятствий.

Сами же функции $\alpha(t)$ и $\beta(t)$ при $t > 20$, например, для *ландшафта* с равномерно рассеянными препятствиями ($\mathcal{L}(100 \times 100, 9)$, $S = 2.125$) приближаются линейными

$$\alpha(t) + \gamma\beta(t) = \mathbb{E}x(t) = 0.979616 + 0.199427t,$$

где γ – постоянная Эйлера,

$$\alpha(t) = 0.352148 + 0.170925t.$$

Построение строя агентов.

Пусть Ag – множество агентов, Ag_1, $Ag_2 \subseteq Ag$ – множества вершин графов Γ_1, Γ_2. Для решения задачи построения строя была разработана метрика несходства графов

$$\mathrm{dist}(\Gamma_1(t), \Gamma_2(t)) = m_0 + m_1 + \sum_{ag \in Ag_c} \|\mathrm{coo}_{\Gamma_1}(t, ag) - \mathrm{coo}_{\Gamma_2}(t, ag)\|,$$

где Ag_c – множество вершин наибольшего общего подграфа Γ_1 (текущего строя) и Γ_2 (желаемого строя), $m_0 = |Ag_0 \setminus Ag_c|$, $m_1 = |Ag_1 \setminus Ag_c|$, и алгоритм предсказания агентом положения других агентов на следующем ходе на основании предыдущих наблюдений за свойствами местности и движением этих агентов. Пусть заданы

1. Функция проходимости $\mathrm{u}^c : [0,T] \times \Omega \to \mathbb{R}_{\geqslant 0}$, $\mathrm{u}^c(t,r)$ – это максимально возможная норма скорости в момент времени $t \in [0,T]$ в точке $r \in \Omega$.
2. Желаемый граф строя $\Phi = (Ag, E, \mathrm{coo}^c_\Phi)$, $E \subseteq Ag^2$ – множество ребер, $\mathrm{coo}^c_\Phi : [0,T] \times Ag \to \Omega$ – функция, дающая желаемые для сохранения строя координаты агента в Ω.

3. Фактический граф строя $\Gamma = (Ag, E, \mathrm{coo}^c_\Gamma)$, $\mathrm{coo}^c_\Gamma : [0,T] \times Ag \to \Omega$ – функция, дающая реальные координаты агента в Ω.

Необходимо найти такие траектории агентов, чтоб время $T = \max_{k=\overline{1,n}} T_k$ было минимальным, т.е. надо минимизировать все функционалы T_k, сопоставляющие траектории r_k агента ag_k продолжительность прохождения по этой траектории. Более формально, нужно решить задачу минимизации

$$T(r_1, \ldots, r_n) = \max_{k=\overline{1,n}} T_k(r_k) \to \min, \quad r_k \in \mathcal{Y}, k = \overline{1,n}. \tag{1}$$

Также необходимо, чтоб в любой момент времени $t \in [0,T]$ отличие графов Φ и Γ было бы минимально

$$\mathrm{dist}(\Gamma(t), \Phi(t)) \to \min. \tag{2}$$

Таким образом, необходимо найти траектории $r_k \in \mathcal{Y}, k = \overline{1,n}$, являющиеся решениями задачи оптимизации с двумя целевыми функциями (1), (2).

Так как ни одному из агентов может быть не известна полная структура графа $\Gamma(t)$, то агенты решают вместо задачи (1), (2) приближенную задачу: они стараются выбрать такое направление движения, которое было бы компромиссным между направлением, минимизирующим $\mathrm{dist}(\Phi(t), \Gamma(t))$ и направлением в сторону кратчайшего по времени пути. Отметим, что в известных работах об организации строя агентов (Н.С. Морозова, Ross Mead, Jerry B. Weinberg) влияние *ландшафта* с препятствиями на построение строя в полной мере не учитывается.

Квзиоптимальное распределение авионики.

Модульная система организации аппаратуры на борту самолета называется интегрированной модульной авионикой (IMA). Устройства IMA, именуемые модулями, параллельно выполняют несколько функций бортовых систем с использованием стандартизированного программного обеспечения. В системе IMA периферийные устройства соединены с единой системой интерфейсов ввода-вывода. Все модули IMA соединены единой широкополосной сетью с высокой пропускной способностью. Параллельное выполнение нескольких функций воздушного судна в одном модуле позволяет сократить количество авиационной радиоэлектронной аппаратуры и вместе с этим массу и стоимость системы авионики, а также – расходы на разработку, производство и техническое обслуживание.

В настоящее время системы аппаратуры IMA внедрены на всех современных гражданских и военных самолетах, самолетах бизнес-класса и вертолетах. В самолетах Boeing 787 и A350 реализованы системы IMA второго поколения, так называемая распределенная интегрированная модульная авионика (DIMA).

Предлагается использование клеточного автомата в виде сетки с движущимися по ней агентами для решения двух задач DIMA: пространственное распределение устройств и присваивания периферийных проводов. В роли подвижных агентов будут выступать сами устройства, пытающиеся найти для себя наиболее «удобное» место на самолете. Несмотря на кажущееся отличие задачи от предыдущих, она может быть решена в рамках все той же модели движения системы агентов и с помощью того же программного обеспечения «Психоход».

Дополнительно предлагается алгоритм квазиоптимального распределения устройств по местам установки в пространстве с использованием модифицированного социального потенциала – т.е. некоторые агенты-устройства стремятся сблизиться друг с другом, а некоторые пытаются максимально разойтись.

В **третьей главе** конструируется модель системы связи мобильных агентов, моделируется процесс самоорганизации такой системы связи.

Данная глава посвящена моделированию системы связи, подобной тем, что используются в тактическом звене управления воинским формированием у нас в стране и за рубежом. Одной отличительной чертой моделируемой системы связи является то, что канал связи между двумя агентами может состоять из нескольких каналов, организованных совершенно различно как на физическом уровне, так и на уровне протоколов. Другой чертой является неустойчивость системы военной связи. В один момент времени между агентами, которые, как правило, оснащены как высокоскоростными средствами с протоколами типа WiMAX или Wi-Fi, так и низкоскоростными узкополосными радиостанциями с нестандартизированными протоколами передачи данных, может быть организован высокоскоростной канал, а в другой – только низкоскоростной, а в некоторые промежутки времени невозможно организовать никакого канала или возможно организовать только составной канал.

Вышесказанное делает довольно затруднительным моделирование канала связи на сетевом или канальном уровне. Более того, в военных условиях интерес представляет не доставка пакета, а именно доставка сообщения в установленный срок, так как ценность доставленной со значительной задержкой информации

может оказаться нулевой. Поэтому моделирование будет проводиться на уровне сообщений, которыми обмениваются агенты, а не пакетов.

Предлагаемая модель сети связи состоит из двух частей – из меняющегося во времени в зависимости от местоположения связывающихся агентов графа системы связи, имеющего ребра, соответствующие каналам связи разного вида, и присвоенных вершинам графа системы связи расписаний рассылки сообщений по другим вершинам, причем скорость доставки сообщения зависит от вида канала между узлом-отправителем и узлом-адресатом. Таким образом, задачи, возникающие при исследовании модели, в некоторой степени напоминают динамические задачи теории расписаний, типа задач управления транспортными системами, однако, в отличие от многих задач этого типа, допустимо одновременное обслуживание произвольного количества требований. Классические модели из теории телетрафика не вполне подходят – поступление сообщений в канал не случайно, а обусловлено расписанием и потеря сообщения исключена. С другой стороны, в разных каналах применяются различные протоколы, не всегда документированные, с разными максимальными скоростями передачи и дисциплинами обслуживания пакетов, что затрудняет имитационное моделирование в известных средах типа ns-3 или OMNeT++.

Клеточно-автоматная модель сети агентов.

Модель системы связи рассматривается как граф с «растяжимыми» ребрами (рис. 3.2), вершинами которого являются агенты из модели движения, описанной во второй главе, а ребрами – каналы связи между агентами. Ребра могут изменять свою «толщину» (пропускную способность) в зависимости от расстояния между агентами и *ландшафта* препятствий. Каждому агенту модели движения (агенту I-го рода) сопоставляется одно или несколько абстрактных средств связи (агенты II-го рода).

Модель самоорганизации системы связи.

Агенты получают информацию о других агентах и состоянии каналов, сканируя спектр и обмениваясь маяками. Один такт клеточного автомата соответствует полному циклу обмена пилот-сигналами и маяками. Подробное техническое решение, соответствующее предлагаемой модели, описано в патенте «Средство передачи данных телекоммуникационной сети и телекоммуникационная сеть» №RU 2 549 120.

Пример конечного состояния автомата показан на рис. 3.9. Более светлый тон ячеек на изображении соответствует лучшему качеству соответствующих

каналов. Мы можем видеть сеть $Net_1 = \{ag_i | i = \overline{1,4}\}$, образованную на канале f_1, и $Net_2 = \{ag_i | i = \overline{5,6}\}$, образованную на канале f_2.

С помощью модели системы связи получены результаты, позволяющие сказать, что самоорганизация агентов II рода в модели каналов связи подобна самоорганизации в модели Шеллинга II и описывается теми же дифференциальными уравнениями

$$\frac{dx}{dt} = [xR_X(x) - y]x, \tag{3}$$

$$\frac{dy}{dt} = [yR_Y(y) - x]y. \tag{4}$$

В системе (3)–(4) x и y – это количество агентов, принадлежащих к разным сетям, на одном канале связи, R_X и R_Y – функции, зависящие от конкретного алгоритма сегрегации (т. е. самоорганизации сетей на разных частотных каналах без внешних управляющих воздействий), например,

$$R_X(x) = \frac{e^{-6x} - e^{-6}}{1 - e^{-6}},$$

$$R_Y(y) = \frac{e^{-6y/k} - e^{-6/k}}{2(1 - e^{-6/k})}.$$

Клеточный автомат Шеллинга представляет собой сетку, где каждая клетка соответствует человеку. Человек может иметь один из нескольких «цветов», который символизирует его принадлежность к определенной социальной группе. Цвета распределяются случайным образом по сетке, а также имеется некоторое количество незанятых ячеек. Каждый человек хочет, чтобы определенная часть людей вокруг него была похожа на него. Если количество соседних ячеек одного цвета ниже заданного порога, то человек переходит в следующую свободную ячейку, в противном случае он остается на месте. Однако, в отличие от двумерной классической модели Шеллинга, предложенный автомат практически одномерен: координата n агента нам нужна только для упрощения описания динамики автомата.

Предлагаемый в описываемом подразделе алгоритм был разработан в качестве попытки ответить на вопрос: возможно ли создать полностью автономную сеть Cognitive Radio, без базовых станций или начального распределения спектра, или же координация с базовой станции необходима для распределения радиоканалов между агентами за конечное число итераций?

Целью настоящего подраздела является попытка разъединить самоорганизацию сети и ее радиофизическую основу и изучить самоорганизацию как чисто математическую задачу сегрегации.

Распределение частот в иерархической системе связи.

Формализована задача автоматизированного распределения частот в сложно организованной системе связи, на примере системы военной связи в тактическом звене управления. Автор решил рассматривать конкретную реализацию распределения частот по устройствам как значение определенной производящей функции вариантов распределения, причем данная функция содержала в себе информацию о ранее вычисленных вариантах распределения и каждый следующий результат был ближе к желаемому. Параметрами вышеуказанной функции является граф потребностей в связи, ребрами которого являются желаемые связи, а вершинами – типы устройств, а также множество реально наличествующих в системе устройств вместе с их типами. Таким образом можно уйти от жесткого задания варианта распределения частот до начала функционирования системы связи. Тогда, например, при существенном изменении помеховой обстановки, агенты смогут самостоятельно вычислить новый вариант распределения частот.

Задача распределения ограниченных ресурсов и построения оптимального расписания в сетевых структурах, в том числе и иерархических структурах транспортного типа, достаточно хорошо известна, и каждый год появляется большое количество статей по этой теме. Обычно в таких задачах необходимо распределить ресурсы между вершинами некоторого графа, причем поток ресурсов через ребра графа так или иначе ограничен. Решением таких задач будет, например, оптимальное в каком-либо смысле расписание или распределение ресурсов (не являющихся элементами самого графа) по вершинам графа.

В данном разделе описывается принципиально иной тип задач распределения ресурсов, часто встречающийся в задачах планирования военной связи (и, в меньшей степени, в гражданской связи), точнее, в задаче распределения и назначения частот. В описываемой задаче ресурсом являются сами ребра некоторого графа (соответствующие каналам связи), которые необходимо распределить между парами вершин в соответствии с заданным шаблоном. Сам шаблон, который далее будет именоваться графом потребностей, аналогичен сети потребностей и возможностей из работ В.А. Виттиха и П.О. Скобелева.

Вершинами – потребителями ресурсов являются пользователи системы связи или/и их технические средства.

При этом характер взаимоотношений между агентами определенных типов в системе намного более постоянен, чем конкретный состав агентов. Решением такой задачи является вариант распределения каналов, отвечающий некоторым требованиям. Будет предложен подход к решению таких задач, ориентированный на работу не с конкретными агентами системы, а с абстрактной структурой системы агентов, «отвязанной» от самих агентов.

Перспективная система связи с возможностью защиты информации.

На основе предыдущих результатов разработана и смоделирована система связи, в которой с помощью маячков и сканирования эфира осуществляется не только распределение агентов по доступным каналам связи, но и исключение постороннего доступа в процесс самоорганизации, равно как и закрытие рассылаемой между агентами информации.

Предлагаемая радиосеть состоит из $nodenum$ радиостанций и одной радиостанции – центра управления связью (ЦУС), которые в целях настройки могут работать на $freqnum$ частотных каналах ($freqnum > 1$), причем конкретный номер канала заранее неизвестен. Предположим, что используемые радиочастотные каналы пронумерованы числами из множества $F = \{1, \ldots, freqnum\}$. Предполагается, что заранее сгенерирована ключевая пара из секретного ключа, который хранится в ЦУС, и открытого ключа, который распространен по всем радиостанциям сети.

В качестве шифрования используется асимметричное шифрование, например RSA (однако, вопросы собственно шифрования, стойкости криптосистемы и т.п. в разделе не затрагиваются, производится лишь конструирование и исследование обменивающейся сигналами-маячками системы агентов). Закрытый ключ должен будет содержаться лишь в одной радиостанции, находящейся в контролируемой зоне, а все другие радиостанции должны будут иметь только открытые ключи, сгенерированные на базе закрытого ключа. Распространение сеансовых ключей между радиостанциями и ЦУС происходит с помощью шифрования сеансового ключа открытым ключом ЦУС.

Организации защищенной сети связи происходит в защищенном режиме посредством передачи по случайному радиоканалу сигналов-маячков, содержащих информацию для аутентификации, проверки целостности сообщения и

асимметричного шифрования, одними радиостанциями (ЦУС), и сканирования набора каналов другими радиостанциями.

Исследуются вопросы перенастройки сети при возникновении помехи на одном из каналов, на котором организована подсеть агентов.

В **четвертой главе** исследуется взаимосвязь клеточно-автоматной и дифференциальной модели движения агента по пересеченной местности. Получены оценки клеточного приближения траектории, доказаны теоремы существования решения.

Непрерывная задача – предельный случай клеточного автомата.

Автор нашел связь между предложенными им клеточными автоматами и непрерывной постановкой оптимизационной задачи многоагентного поиска кратчайшего пути

$$\|\dot{r}_k(t)\| = u(t,r_k(t)) - v_{k,\varepsilon}(t, r_1(t), \ldots, r_m(t)),$$

$$v_{k,\varepsilon}(t, r_1(t), \ldots, r_m(t); \mathcal{I}^k) = \\ = \begin{cases} u(t,r_k(t)), \quad \xi_k(r_1(t), \ldots, r_m(t)) \leqslant \varepsilon, \\ \chi_\varepsilon(\kappa,\xi_k(r_1(t), \ldots, r_m(t))),\ \varepsilon + \kappa \geqslant \xi_k(r_1(t), \ldots, r_m(t)) > \varepsilon, \\ 0, \quad \xi_k(r_1(t), \ldots, r_m(t)) > \varepsilon + \kappa, \end{cases}$$

$$\chi_\varepsilon(\kappa,\xi) = u(t, r_k(t)) \left(\frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{z_0(\kappa + 2\varepsilon - 2\xi)}{\kappa} \right) \right),$$

$$\xi_k(r_1, \ldots, r_m) = \frac{1}{m-1} \sum_{i \in \mathcal{I}^k} \|r_k - r_i\|,$$

$$\Theta(r_k,l) = \min\{t \in (0,T] | r_k(l_k(t)) \in \partial\Omega\} \to \min.$$

Здесь r_k – маршрут k-го агента, l – его натуральная параметризация, Ω – область, в которой перемещаются агенты, u – функция расположения препятствий, κ, ε – параметры. Получены оценки точности приближения оптимального пути клеточным маршрутом. Данная задача затруднялась тем, что применение традиционных вариационных методов для решения невозможно, так как обычно невозможно получить функционал времени прихода в конечную точку в явном виде, а даже если и возможно, то такой функционал оказывается недифференцируемым.

В главе применяются стандартные обозначения: $\|u\| = \sqrt{\sum_{i=1}^{n} u_i^2}$ – евклидова норма вектора $u \in \mathbb{R}^n$, $u = (u_1, \ldots, u_n)$, $W_p^m(A; \mathbb{R}^n)$ – пространства Соболева определенных на A функций со значениями в $\mathbb{R}^n$, которые вместе со своими m слабыми производными содержатся в $L_p(A; \mathbb{R}^n)$ с нормой

$$\|u\|_{W_p^m(A;\mathbb{R}^n)} = \sqrt{\sum_{i=1}^{n} \Big(\sum_{|\alpha| \leqslant m} \int_A |D^\alpha u_i(x)|^p dx \Big)^{2/p}},$$

$C(A; \mathbb{R}^n)$ – пространство определенных на A непрерывных функций со значениями в $\mathbb{R}^n$ с нормой $\|u\|_{C(A;\mathbb{R}^n)} = \max_{x \in A} \|u(x)\|$, $C^m(A; \mathbb{R}^n)$ – пространство m раз непрерывно-дифференцируемых определенных на A функций с нормой $\|u\|_{C^m(A;\mathbb{R}^n)} = \sum_{|\alpha| \leqslant m} \max_{x \in A} \|D^\alpha u(x)\|$.

В качестве множества маршрутов мы будем рассматривать

$$\mathcal{Y} \subset W_p^1(0,L; \mathbb{R}^2),$$

состоящее из всех вектор-функций из $C([0,L]; \mathbb{R}^2)$, которые

– представимы в виде не более чем счетного множества дважды непрерывно дифференцируемых кусков, т. е. для любой $r \in \mathcal{Y}$ существует такое разбиение $\mathbb{R}$ точками $\{t_i\}$, что $r|_{(t_i,t_{i+1})} \in C^2((0,L); \mathbb{R}^2)$;

– производные всех функций из $\mathcal{Y}$ равномерно ограничены (кроме, возможно, счетного множества точек, в которых производной не существует):

$$\max_{l \in [0,L]} \|r'(l)\| \leqslant C_1;$$

– вторые производные всех функций из $\mathcal{Y}$ равномерно ограничены в сумме:

$$\sum_i \max_{l \in (L_i, L_{i+1})} \|r''(l)\|^p \leqslant C_s,$$

где L_i – точки разрыва первой и второй производных r.

Существование оптимального решения задачи поиска кратчайшего пути во множестве функций $\mathcal{Y}$ доказывается с помощью обычных средств функционального анализа, а именно, ищется минимизирующая функционал Θ последовательность, сходимость которой доказывается с помощью теорем о компактном вложении. Доказывается

Теорема 1. *Функционал* $\Theta : \mathcal{Y} \to \mathbb{R}$ *достигает минимума на* $\mathcal{Y}$.

Далее строится алгоритм численного решения задачи построения скорейшего маршрута, в котором используется приближение гладкого маршрута r агента ломаной r_h, в которой отдельные звенья могут стыковаться только под 0, 45, 90, 180 градусов (множество таких ломаных обозначается $\mathcal{Y}(h)$), а функции расположения препятствий u – ступенчатой функцией. Показано, что справедлива

Лемма 6. *Пусть* $r = (x(l), y(l)) \in \mathcal{Y}$, $r_h \in \mathcal{Y}(h)$ *и* h *достаточно мало. Тогда справедлива оценка*

$$\|r - r_h\|_{C[0,L]} \leqslant h\sqrt{2}. \tag{5}$$

Доказывается, что при определенных условиях полученные приближения сходятся к оптимальному маршруту по норме $C([0,L],\mathbb{R}^2)$:

Теорема 3. *Существуют такие* $r_h^* \in \mathcal{Y}(h)$, *минимизирующие функционал* Θ_h, *и такой* $r_0^* \in \mathcal{Y}$, *минимизирующий функционал* Θ, *что* $r_h^* \to r_0^*$ *в* $C([0,L],\mathbb{R}^2)$ *при* $h \to 0$.

В **<u>пятой главе</u>** описывается программа «Психоход», применяемая для постановки вычислительных экспериментов, связанных с результатами диссертации и разработанная автором.

Модель основана на абстракции, в которой воинским формированиям или иным группам иерархически организованных сущностей того или иного уровня соответствуют совместно перемещающиеся агенты соответствующего уровня. Время перемещения агентов дискретно, перемещаются они на фиксированный и зависящий от вида агента промежуток за один такт дискретного времени. Каждый агент обладает определенным искусственным интеллектом, позволяющим самостоятельно принимать решения по атаке, отступлению, обходу препятствий.

Программная среда «Психоход» реализована на языке C++14 с применением библиотеки Qt5. Сценарий конкретного эксперимента, определяющий тип *ландшафта*, поведения агентов, их вид, иерархию, цели и задачи и т.п. задается, как правило в виде скрипта на языке Lua, загружаемого через встроенный в среду интерпретатор языка Lua или, в простых случаях, с помощью графического интерфейса пользователя. Ход эксперимента выводится на экран и (или) в текстовый файл в соответствии с настройками. Количество клеточных автоматов, поддерживаемых средой, неограниченно расширяемо, так как все остальные

подсистемы среды взаимодействуют с автоматом не напрямую, а через интерфейсный абстрактный клеточный автомат самого общего вида.

Глава 1. Обзор предыдущих работ

1.1 Обзор классификации, методов и методологий агентных систем

В одном из первых (1999 г.) обзоров [37] было предложено следующее деление методологий построения агентных систем на

1. Объектно-ориентированные методологии, в которых агенты (в том числе и BDI-агенты) рассматриваются как активные объекты.
2. Методологии инженерии знаний (Knowledge engineering, KE), в которых используются онтологии и решатели задач.
3. Формальные подходы, основанные на применении формальных языков и темпоральной модальной логики.
4. Смешанные варианты.

В более поздней работе [38] содержится более подробный вариант вышеприведенной классификации:

1. Моделирование индивидуального и коллективного поведения агентов в многоагентной системе.
2. Теоретико-игровое моделирование взаимодействия агентов.
3. Лингвистическое направление, связанное с моделированием речевых актов для построения протоколов коммуникации между агентами.
4. Организационное моделирование многоагентных систем, связанное с общей теорией систем и организаций.
5. Построение моделей распределенных систем на базе сетей Петри.
6. Программистское направление, связанное с моделями акторов, параллельным и агентно-ориентированным программированием.
7. Логико-семиотическое направление, связанное с моделированием ментальных свойств (интенциональных характеристик) агентов с помощью неклассических логик.

В обзоре 2000 г. [39] многоагентные системы (Multiagent Systems) рассматривались вместе с распределенным решением задач (Distributed Problem Solving) как составные части распределенного искусственного интеллекта (Distributed Artificial Intelligence). В указанном обзоре излагались различные обычные сценарии для многоагентных систем и перечислялись возможности для применения

техник машинного обучения. Предлагалась классификация систем агентов по двум измерениям:

1. Степень неоднородности агентов (гомогенные и гетерогенные системы).
2. Количество и вид взаимодействия агентов друг с другом.

По первому измерению системы агентов можно подразделить на гомогенные и гетерогенные, по второму – различать общающихся и необщающихся агентов, кооперативных и соревнующихся агентов и т.п.

Обзоры многоагентных методологий, особенно в части логико-семиотического направления, уже существуют, в том числе и на русском языке (например, [38]). Автор попытался подобрать показавшиеся ему интересными многоагентные модели из разных отраслей науки и техники, уже реализованные в виде программного (или аппаратного) обеспечения, чтобы создать у читателей общую картину практического применения многоагентной методологии.

В докладе 2004 г. [40] различалось аналитическое (с помощью уравнений) и имитационное моделирование (Simulation), применяемое в случае затруднительности решения или формулировки аналитической модели. В имитационном моделировании выделялись такие подходы, как System Dynamics, дискретно-событийное моделирование и агентное моделирование. При этом демонстрировалось, что решение в агентной модели при достаточно большом количестве агентов соответствует решению в рамках System Dynamics и предлагался довольно известный сейчас инструмент мультипарадигменного моделирования AnyLogic.

К 2006 г. уже существовали десятки сред разработки агентных систем. Например, в обзоре [41] перечислялись декларативные, императивные и смешанные языки описания агентов и многие существовавшие на тот момент IDE. Одним из ключевых свойств агентных систем является способность к параллельному и распределенному вычислению и различные решения в этой области рассмотрены в обзоре 2014 г. [42]. В упомянутом обзоре сравнивалась производительность систем Repast Simphony, Mason, NetLogo, Gama, а также нативно поддерживающих параллельные вычисления систем RepastHPC, D-Mason, Pandora, Flame и JADE. Один из наиболее полных обзоров поддерживаемых в настоящее время агентных платформ приведен в [43]. В числе прочего, там указано, что наиболее популярной средой агентного моделирования является написанная на java JADE.

Существуют работы, в которых многоагентные системы исследуются с точки зрения классической теории управления (так называемое сетевое управление) [44]. Для сетевого управления характерно сочетание классического аппарата теории управления с методами теории графов. Например, в работе [45] указывается, что задачи управления, такие как управление коллективом роботов, распределенный интеллект, роевой интеллект, распределенное принятие решений, распределенное познание, управление перегрузками в сетях, коллективное движение в биологии, синхронизация осцилляторов в физике, распараллеливание в теории оптимизации, распределенная оценка, совместная оценка, равновесие в экономике, моделирование социальных взаимодействий и теория игр могут быть проанализированы в соответствии с теорией взаимосвязанных динамических систем. Эти науки имеют несколько взаимопересекающихся исследовательских сообществ; по этой причине они характеризуются различными определениями и различными подходами, начиная от строгого математического анализа и заканчивая экспериментальным исследованием. По этой же причине отсутствует общая терминология: например, динамические системы могут именоваться роботами, агентами, узлами, процессорами и объектами. Ансамбль называется группой, сетью, взводом, роем, командой и кластером, а алгоритмы получают разные названия, такие как контроллеры, протоколы и динамика. В цитируемой работе термин «агент» используется для обозначения отдельной динамической системы, а «сеть» или «коллектив» – для ансамбля. Формулируются задачи нахождения консенсуса, построения строя, оптимального размещения ресурсов на языке динамических систем.

В диссертации [46] рассматриваются задачи поиска консенсуса и синхронизации агентов, возникающие во многих отраслях науки. Агенту соответствует некоторая динамическая система, консенсус и синхронизация исследуются в сетях отдельных динамических систем, связанных в соответствии с некоторой топологией, которая определяется графом связи. Сами системы моделируются обыкновенными дифференциальными уравнениями. В вышеупомянутой диссертации описывается куб сложности связанных систем (рис. 1.1). Если положить, что связь между отдельными системами описывается некоторой топологией, то можно выделить три независимых измерения сложности: сложность отдельных систем, сложность топологии связности и сложность самих связей.

В диссертации [46] выделяются диффузионно связанные системы, которые, в свою очередь, подразделяются на статически связанные и динамически

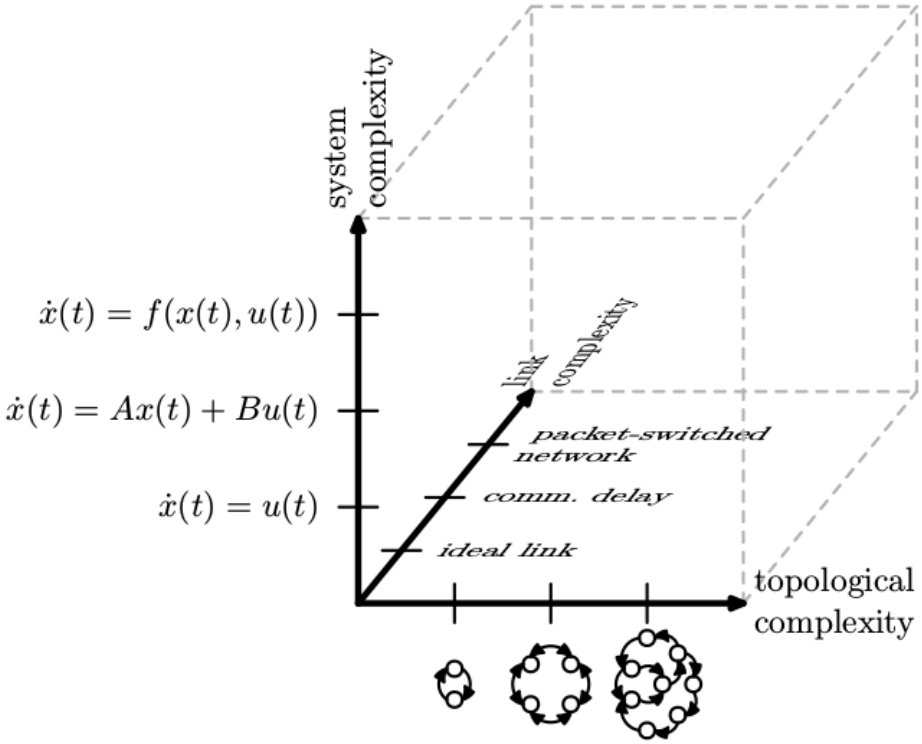


Рисунок 1.1 — Куб сложности для связанных динамических систем

связанные. В статически связанных системах, например (1.1)–(1.2), есть лишь зависимость от мгновенных значений состояний и нет состояния связи. Входы системы связаны с относительными состояниями алгебраическим уравнением:

$$\dot{x}(t) = u(t), \tag{1.1}$$

$$u(t) = -L(t)x(t), \tag{1.2}$$

где $x(t) \in \mathbb{R}^n$ – вектор состояний систем, $u(t) \in \mathbb{R}^n$ – вектор входов систем, $L(t)$ – зависящая от времени матрица Кирхгофа связи отдельных систем (в зарубежной литературе матрица Кирхгофа чаще называется матрицей Лапласа или дискретным лапласианом). Для графа $\mathcal{G} = (V, E, A)$ с матрицей весов ребер A матрица Кирхгофа определяется как

$$L(A) = \begin{cases} -a_{ij}, i \neq j, \\ \sum_{i \neq j} a_{ij}, i = j. \end{cases}$$

В противоположность этому динамически связанные системы характеризуются тем свойством, что они зависят от мгновенного значения относительных состояний и от состояния связи, т.е. сами описываются динамической системой.

В динамических соединениях входы систем связаны с относительными состояниями дифференциальным уравнением. Пример динамически связанных систем:

$$\dot{x}_k(t) = Ax_k(t) + Bu_k(t), \tag{1.3}$$

$$y_k(t) = Cx_k(t), \tag{1.4}$$

$$\dot{z}_k(t) = Ez_k(t) + F\delta_k(t), \tag{1.5}$$

$$u_k(t) = Gz_k(t) + H\delta_k(t), \tag{1.6}$$

$$\delta_k(t) = \sum_{j=1}^{N} w_{k,j}(t)(y_k(t) - y_j(t)), \tag{1.7}$$

где $x_k(t) \in \mathbb{R}^n$ – вектор состояний k-й системы; $u_k(t) \in \mathbb{R}^p$ – вектор ее входов; $y_k(t) \in \mathbb{R}^q$ – выходов; $z_k(t) \in \mathbb{R}^m$ – состояние связи; $\delta_k(t)$ – относительный выходной сигнал.

Следует отметить сборник лекций [47], в котором обобщается и систематизируется проблематика, связанная с сетевыми системами, т.е. с системами взаимосвязанных агентов (вид связей определяется графом). В указанном сборнике производится следующая классификация сетевых систем:

1. линейные системы, например, модели динамики мнений, компартментальные модели живых систем, модели оптимального патрулирования периметра и т.п., включающие
   а) системы усреднения с дискретным временем вида
   $$x(k+1) = Ax(k),$$
   где $A = (a_{ij})$ – матрица $n \times n$, $a_{ij} \geqslant 0$, $\sum_{j=1}^{n} a_{ij} = 1$, $i,j = \overline{1,n}$.
   б) системы усреднения с непрерывным временем вида
   $$\dot{x}(t) = Ax(t),$$
   включая модели образования роя, непрерывные модели динамики мнений, модели соединения RC-элементов электрической цепи и пр. Во всех случаях, матрица A связана с графом связи отдельных систем через матрицу Кирхгофа.
2. нелинейные системы, включая
   а) модели Лотка–Вольтерра популяционной динамики вида
   $$\dot{x}_1 = x_1(r_1 + a_{11}x_1 + a_{12}x_2),$$
   $$\dot{x}_2 = x_2(r_2 + a_{21}x_1 + a_{22}x_2),$$

б) модели распространения вирусов вида

$$\dot{x}(t) = \beta(I - \operatorname{diag}(x(t)))Ax(t),$$

в) сети соединенных осциляторов с моделью Курамото вида

$$\dot{\theta}_i = \omega_i - K/n \sum_{j=1}^{n} \sin(\theta_i - \theta_j), \quad i = \overline{1,n},$$

г) модели построения строя роботов, в которых необходимо поддерживать постоянное расстояние между определенными элементами строя.

Задачи сетевого управления освещаются во многих статьях отечественных авторов (например, [48–51]): П.Ю. Чеботарева, Р.П. Агаева, Н.О. Амелиной, С. Э. Парсегова и др.

1.2 Задачи, решаемые с помощью агентного подхода в естественных, технических и общественных науках

1.2.1 Математика

Существуют многочисленные многоагентные подходы (роевая оптимизация и близкие методы) к решению задач дискретной оптимизации. Из них наиболее известны генетические алгоритмы, алгоритм муравьиной колонии и метод роя частиц. Менее известны алгоритм пчелиной колонии, искусственные иммунные системы, алгоритмы разумных капель, алгоритм кукушки и другие алгоритмы, основанные на биологических и физических метафорах. Данные подходы хорошо исследованы и им посвящено большое количество обзоров, например [52]. Однако следует отметить, что наблюдается тенденция к злоупотреблению подобными метафорами, когда под экзотической аналогией скрывается уже хорошо известный метод [53].

1.2.2. Механика

В механике твердого тела, в механике жидкости и газа существуют подходы, в которых среда рассматривается как множество последовательно движущихся по дискретной решетке или сталкивающихся фиктивных частиц, т.е.

как клеточный автомат. Примерами таких методов является movable cellular automaton (MCA, [54]), Lattice Gas cellular automata и Lattice Boltzmann methods (LBM).

Пусть $f_i(\vec{x},t)$ – одночастичная функция распределения, дающая вероятность нахождения частицы со скоростью $\vec{v}_i$ в клетке $\vec{x}$ в момент времени t. Скорость $\vec{v}_i$ может принимать значение из некоторого конечного множества, зависящего от типа решетки, например $\vec{v}_i = (\cos(2\pi i/6), \sin(2\pi i/6))$ для гексагональной сетки. Функция перехода клеточного автомата в LBM обычно состоят из двух шагов вида:

1. Релаксация (также говорят «шаг коллизий»)

$$f_i^t(\vec{x},t+\delta_t) = f_i(\vec{x},t) + \frac{1}{\tau}(f_i^{eq} - f_i),$$

   где $\vec{x}$ – вектор координат клетки в решетке; f_i^{eq} – локальная функция равновесия, аппроксимирующая распределение Больцмана $f_{eq}(\vec{v}_i) = \frac{\rho}{(2\pi RT)^{D/2}} e^{-\frac{(\vec{v}_i-\vec{u})^2}{2RT}}$ (R – универсальная газовая постоянная; T – температура; ρ – плотность среды; u – макроскопическая скорость газа; D – размерность пространства); τ – время релаксации; δ_t – шаг дискретного времени.

2. Распространение потоков:

$$f_i(\vec{x}+\vec{e}_i\delta_t,t+\delta_t) = f_i^t(\vec{x},t+\delta_t).$$

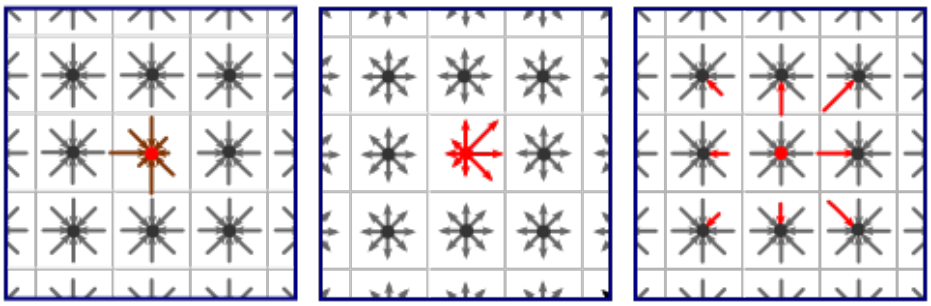


Рисунок 1.2 — Тактовое функционирование клеточного автомата для LBM вида D_2Q_9

В работе С. Вольфрама [55] рассматривается применение LBM к моделям жидкостей, описываемых уравнениями Навье–Стокса. Отметим, что в этих моделях применяют автоматы с треугольными, квадратными, шестиугольными и другими ячейками. Общая схема классификации моделей имеет вид D_nQ_m. Здесь D_n – это размерность n решетки, Q_m – количество m соседей для клетки,

включая ее саму. Преимуществом данных методов над традиционными методами вычислительной механики является легкость применения параллельных вычислений. Существуют специализированные обзоры таких подходов, составленные специалистами по механике (например, [56]), автору же данного обзора хочется отметить, что именно клеточные автоматы, первоначально описывающие динамику газа, породили многие из моделей поведения толпы.

1.2.3 Биология и медицина

Биомедицинские многоагентные модели обычно напоминают модели из предыдущего подраздела. В качестве клеток клеточного автомата в модели обычно берутся живые клетки. Например, в статье [57] описан двухмерный клеточный автомат (рис. 1.3), моделирующий развитие опухоли. В качестве ячеек автомата рассматриваются нормальные и опухолевые клетки, а функция перехода автомата учитывает уровень насыщенности кислородом, близость клетки к сосуду и разное поведение нормальных и раковых клеток. Показывается, что ток крови и неоднородность красных кровяных телец играют главные роли в развитии таких колоний клеток, даже когда красные кровяные тельца проходят через сосуды в нормальной, здоровой ткани.

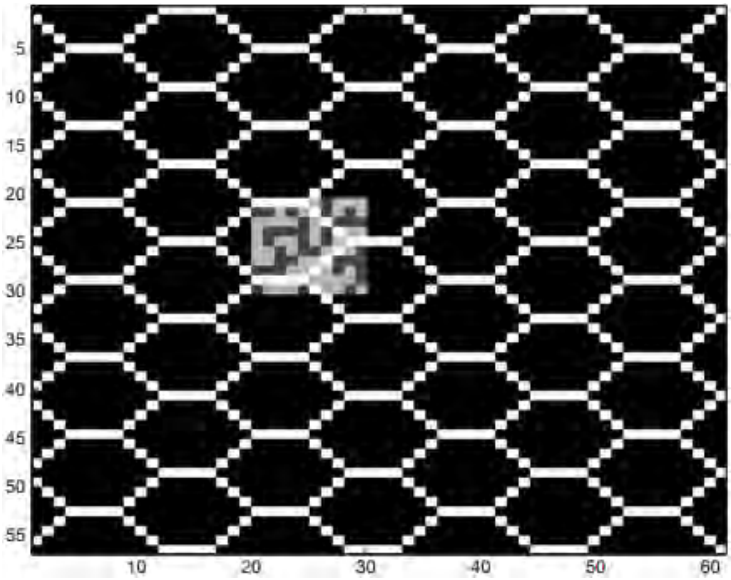


Рисунок 1.3 — Начальное распределение элементов клеточного автомата. Черные клетки пустые, белые заняты сосудами, светло-серые – раковыми клетками и темно-серые – нормальными клетками

Правила эволюции вышеуказанного клеточного автомата таковы:

1. Пустой или занятый сосудом элемент не развивается, за исключением случая, когда в соседнем занятом элементе происходит деление клетки.

2. Распределение кислорода рассчитывается путем решения соответствующей краевой задачи

$$\frac{\partial P}{\partial t} = D_P \nabla^2 P - k(\vec{x})P, \tag{1.8}$$

$$-D_P \mathbf{n}_w \cdot \nabla P = \mathcal{P}(P_B - P), \tag{1.9}$$

$$\mathbf{n}|_{\partial\Omega} \cdot \nabla P = 0, \tag{1.10}$$

где P – распределение внеклеточного кислорода, D_P – коэффициент диффузии, $k(\vec{x})$ – скорость потребления кислорода в позиции $\vec{x}$, $\mathbf{n}_w$ – вектор внешней к стенке сосуда единичной нормали, $\mathcal{P}$ – проницаемость сосуда, P_B – уровень кислорода внутри сосуда, $\mathbf{n}|_{\partial\Omega}$ – вектор внешней единичной нормали к границе области. Величина $k(\vec{x})$ обновляется на каждом шаге функционирования клеточного автомата и зависит от клеток в позиции $\vec{x}$.

3. Тип клетки в занятом элементе хранится в векторе состояния. Клетки пытаются делиться на каждом шаге.

4. Правила процесса деления зависят от типа клетки. Для нормальной клетки определяется локальная концентрация кислорода. Если концентрация кислорода ниже порогового значения, клетка погибает, в противном случае пытается делиться. Пороговое значение определяется по состоянию ближайших соседей. Если клетка окружена по большей части нормальными, а не раковыми клетками, то фиксируется пороговое значение N_{T1}, в противном случае фиксируется $N_{T2} > N_{T1}$.

Правила для раковых клеток аналогичны правилам для нормальных клеток, за исключением того, что сравнение между уровнем кислорода и пороговым значением определяет только то, будет ли клетка делиться. Правила гибели раковой клетки определяются по-другому. Если с клеткой соседствует больше раковых клеток, чем нормальных, тогда для нее фиксируется пороговое значение C_{T1}, в противном случае фиксируется $C_{T2} > C_{T1}$.

Неравенства $C_{T1} < C_{T2}$ и $N_{T1} < N_{T2}$ означают, что клетки с большей вероятностью делятся, если они, в основном, окружены клетками того же типа.

5. Элементы, занятые раковыми клетками, в которых локальная концентрация кислорода ниже порогового, входят в покоящееся состояние, в течение которого большая часть функций клетки приостанавливается, включая пролиферацию. При входе в это состояние запускается таймер, увеличивающийся на единицу каждый шаг, когда ячейка остается в состоянии покоя. Если таймер достигает заданного значения, клетка погибает. Однако, если в какой-то момент

уровень кислорода перейдет выше порогового значения, ячейка вернется к размножающемуся состоянию, а ее таймер сбросится до нуля.

6. Элементы, занятые нормальными и раковыми клетками, являются потребителями кислорода.

7. Если уровень кислорода в элементе, занятом клеткой, выше порога, то клетка будет делиться, если в ее окрестности радиуса R есть пустые элементы. Если свободное место есть, то новая клетка переходит в свободный элемент с наибольшей концентрацией кислорода. Иначе клетка не может делиться и погибает.

Модель роста опухоли в статье [58] также основана на клеточном автомате, подобном вышеупомянутому, но учитывает генетические особенности злокачественных клеток, делающих их нечувствительными к сигналам прекращения роста, выключающими программу клеточной смерти (апоптоза) и т.п.

В статье [59] моделируется процесс функциональной адаптации костной ткани с помощью гибридного (т.е. сочетающего дискретные и непрерывные правила) клеточного автомата, подобного используемым в механике твердого тела.

В статье [60] приведена простая многоагентная модель эпидемии. Модель была предназначена для исследования влияния таких факторов, как начальное количество инфицированных, плотность популяции и максимальное количество клеток, через которое может распространяться инфекция. Далее это направление исследований получило определенную популярность и сейчас существуют агентные модели эпидемии, учитывающие образцы человеческого поведения, полученные из реальных данных, мобильность и т.д., например, [61].

В работе [62] рассматривалось поведение молекул или клеток в переполненном окружении. Многие биологические среды переполнены макромолекулами, органеллами и клетками, которые могут препятствовать переносу других клеток и молекул. В отличие от предыдущих исследований, которые стремились описать эти эффекты, используя либо модели случайного блуждания, либо уравнения диффузии дробного порядка, в вышеупомянутой работе рассматривается перенос как одного агента, так и популяции агентов через среду, содержащую препятствия различного размера и формы, относительные плотности которых взяты из определенного распределения. Результаты моделирования для одного агента указывают на то, что более мелкие препятствия более эффективны при замедлении транспорта, чем большие препятствия, эти результаты согласуются с симуляциями коллективного движения популяций агентов. В попытке

выяснить, можно ли описать эти виды случайного моделирования случайного блуждания, используя структуру уравнения диффузии дробного порядка, авторы калибровали решение такого дифференциального уравнения с полученной ими информацией об усредненной плотности вещества. Подход авторов предполагал, что широко используемые модели дифференциальных уравнений следует использовать с осторожностью, поскольку не всегда возможно согласовать решение уравнения диффузии дробного порядка с данными авторов согласованным образом в течение конечного периода времени.

1.2.4 Неклассические вычисления

Многоагентный подход применяется в неклассических вычислениях, например, в биологических вычислениях. Слизевик *Physarum polycephalum* (гигантское многоядерное одноклеточное существо, рис. 1.4) часто применяется как модельный организм для биологических вычислений, который может быть запрограммирован различными внешними стимулами. В ответ на эти стимулы он растет, меняет конфигурацию, выпускает ложноножки и т.п. [63], что и кодирует определенные операции над числами, помимо этого слизевик способен к обучению и самообучению. Например, в свое время этот слизевик был использован в качестве системы управления роботом, избегающим света.

Рисунок 1.4 — *Physarum polycephalum* на стволе дерева (фотография frankenstoen – flickr, CC BY 2.5, https://commons.wikimedia.org/w/index.php?curid=18930526)

Существует сайт http://www.phychip.eu/, посвященный проекту PhyChip – построению вычислительной системы на основе слизевиков *Physarum*. В статье [64] рассматривается многоагентная среда, симулирующая поведение *Physarum polycephalum*.

Поведение слизевиков привело к созданию моделей движения толпы [65]. Предлагается основанная на клеточном автомате модель *Physarum*, которая имитирует процесса поиска и диффузии пищи, роста организма, создания ложноножек для каждого организма. При этом выбору оптимального пути эвакуацией толпы соответствует оптимальный выбор ложноножки моделируемого организма *Physarum*.

Также существуют связанные с поведением слизевиков модели из различных видов неклассической логики и теории игр [66–68]. В статье [69] рассматривается клеточные автоматы с правилами, меняющимися на каждом ходу. Используя эти автоматы, возможно формализовать рефлексивные игры на основе контекста для k игроков на разных конечных или бесконечных уровнях рефлексии. Каждый игрок имеет собственное правило принятия решения, определяемое логической функцией на своих выигрышах в контексте, который постоянно изменяется. Это правила нулевого уровня рефлексии. Если игрок 1 следует правилу принятия решения, которое является логической комбинацией с решающим правилом игрока 2 нулевого уровня рефлексии, то мы говорим, что правило решения игрока 1 относится к первому уровню. Между тем, если игрок 2 следует правилу принятия решения, которое является логической комбинацией с решающим правилом игрока 1 первого уровня рефлексии, то говорится, что правило решения игрока 2 относится к второму уровню и т.д.

В работе [70] рассматривается роевое поведение алкоголиков с точки зрения неклассической логики. В качестве роев рассматриваются небольшие группы собутыльников. Было обнаружено, что алкоголики образуют сеть, состоящую из нескольких небольших групп. Задача оптимизации обычных напитков решена не небольшой группой, а всей сетью, т.е. несколькими группами, члены которых взаимосвязаны.

Дело в том, что каждая небольшая группа алкоголиков появляется и исчезает в разных условиях, но сеть, к которой относятся эти алкоголики, остается практически постоянной. Оказалось, что небольшие группы лиц, страдающих алкоголизмом, нестабильны и, обмениваясь своими членами, могут слиться или

разделиться для оптимизации потребления алкоголя. Те же поведенческие закономерности наблюдаются у слизевиков: сплавление и расщепление перед аттрактантами для оптимизации их захвата.

Внешние стимулы (аттрактанты) для слизевиков представляют собой кусочки питательных веществ, разбросанных перед этим организмом. Аттрактанты для алкоголиков представлены местами, где они могут спокойно пить в небольших группах. 38% респондентов предпочитают пить в одном месте и 62% в разных местах. Аргументы при выборе мест следующие: короткое расстояние от дома, низкая цена, качество напитков.

1.2.5 Транспорт

Агентные (микроскопические) модели применяются для моделирования транспортных потоков, обычно однополосных. В основе подходов лежит концепция «о желании придерживаться при движении безопасной дистанции до лидера». В монографии [71] приводятся, например, модель оптимальной скорости Нагеля–Шрекенберга, основанная на клеточных автоматах, которая является простейшей. В этой модели на каждом шаге $m \to m+1$ состояние всех автотранспортных средств в системе обновляется в соответствии со следующими правилами (здесь $v_n \leqslant v_{\max}$ – скорость, а s_n – координата n-го транспортного средства):

1. ускорение (отражает тенденцию двигаться как можно быстрее, не превышая максимально допустимую скорость):

$$v_n(m+1) = \min\{v_n(m)+1, v_{\max}\};$$

2. торможение (гарантирует отсутствие столкновений с впереди идущими автотранспортными средствами):

$$v_n(m+1) = \min\{v_n(m), s_{n+1}(m) - s_n(m) - d\},$$

   где $d \sim 7{,}5$ м;

3. случайные возмущения (учитывают различия в поведении транспортных средств):

$$v_n(m+1) = \begin{cases} \max\{v_n(m)-1{,}0\}, \text{ с вероятностью } p, \\ v_n(m), \text{ с вероятностью } 1-p; \end{cases}$$

4. движение:

$$s_n(m+1) = s_n(m) + v_n(m).$$

В работе [72] приводятся модели кратковременной динамики следования за лидером, которые описывают непрерывные задачи обратной связи (ускорение и торможение) и задачи дискретного выбора как ответ на окружающий транспортный поток. Стиль вождения агентов характеризуется параметрами модели – временем реакции, желаемой скоростью и т.п. Также используются переменные внутреннего состояния, соответствующие уму агента, для описания опыта водителя. На поведение водителя действуют соседние транспортные средства и другие внешние воздействия.

1.2.6 Модели толпы

Предположительно самая первая работа, описывающая агентный подход к моделированию роя или стаи, – это статья Рейнольдса [73]. В ней выделено три основных принципа построения роя:

- избегание столкновений: избегать столкновений с ближайшими соседями по рою;
- совпадение скоростей: пытаться двигаться с той же скоростью, что и ближайшие соседи по рою;
- центрирование по рою: пытаться держаться близко к ближайшим соседям по рою.

Автор предлагал применять вышеизложенные принципы для графического отображения роя частиц, но вскоре они нашли и другие применения.

В работе [74] использован двухмерный клеточный автомат (так называемый world-space cellular automaton, рис. 1.5, изображение из цитируемой ниже работы Ross Mead) с правилами, близкими к *Game of Life* Конвея, симулирующий образование роя (swarm) агентов с помощью поля социального потенциала. Агенты отталкиваются друг от друга, если слишком сильно приблизятся или притягиваются, если находятся достаточно далеко друг от друга. Сила отталкивания и (или) притяжения между двумя агентами описывается как

$$f(d) = -\frac{c_1}{d^{\alpha_1}} + \frac{c_2}{d^{\alpha_2}},$$

где c_1 , $c_2 \geqslant 0$, $\alpha_1 >> \alpha_2 >> 0$ – параметры агентов, d – расстояние между агентами. Если агент A обнаруживает соседей $N_1, \ldots N_k$ на расстояниях $d_1, \ldots, d_k$, то сила, примененная к агенту i соседом N_i будет

$$f_A(d_i) = -\frac{c_{1i}}{d_i^{\alpha_{1i}}} + \frac{c_{2i}}{d_i^{\alpha_{2i}}}.$$

Суммарная сила, примененная к агенту A будет

$$F(A) = \sum_{i=1}^{k} f_A(d_i).$$

Предполагалось использование миниатюрных роботов, снабженных сенсорами и системой связи, с таким поведением при ликвидации чрезвычайных ситуаций.

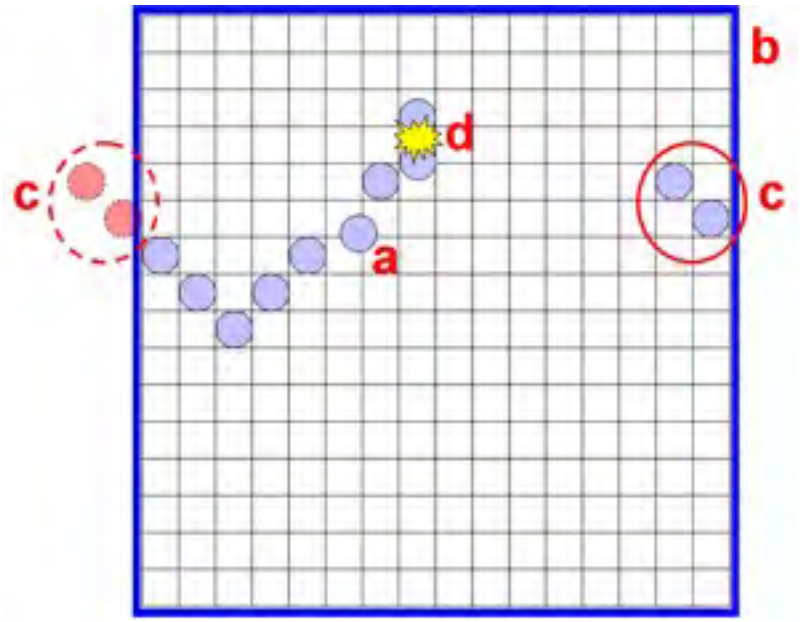


Рисунок 1.5 — World-Space Cellular Automaton и его ограничения: (а) робот между клетками сетки; (б) граница, окружающая автомат; (в) автомат оборачивается вокруг границы; (г) два робота пытаются занять одну и ту же клетку

Так вышло, что самая первая работа по моделированию движения группы людей, которую автор настоящего обзора прочел и которая определила его интерес к агентным системам – это статья [75]. В ней применяется клеточный автомат с вероятностной локальной функцией перехода, изначально предложенный для моделирования динамики газов, для построения модели движения неорганизованной группы людей между препятствиями, например, между ограждениями в метро.

Следует отметить, что в России сложилась своя собственная школа математического моделирования поведения толпы, и стоит упомянуть в связи с этим монографию [76], посвященную математическим моделям управления толпой, поведение которой описывается как пороговое (конформное) коллективное принятие решений ее элементами.

1.2.7 Управление строем

Стоит отметить, что для данного направления характерно, что одни и те же результаты используются и для моделирования строя агентов на компьютере, и для проектирования искусственного интеллекта наземных роботов и БПЛА.

В [77–80] и других работах того же коллектива авторов описан алгоритм, позволяющий организовывать строй (formation) агентов, определенный с помощью одной или нескольких математических функций. Формация агентов рассматривается как одномерный клеточный автомат (так называемый robot-space cellular automaton), каждая клетка которого содержит координаты агента на плоскости, его реальное и желаемое взаимное расположение с соседними агентами (см. рис. 1.6).

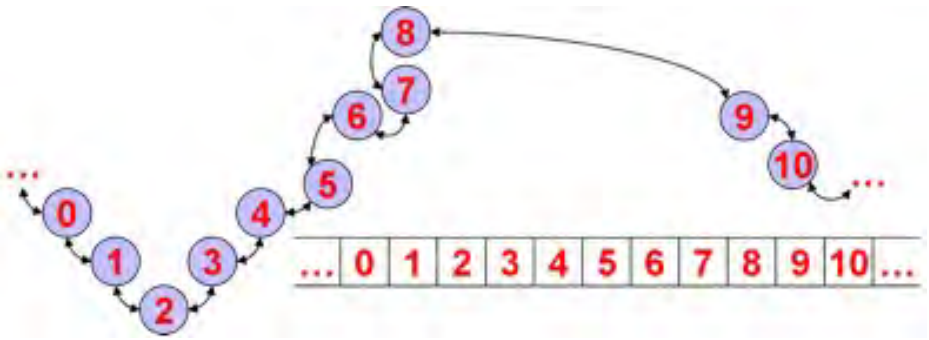


Рисунок 1.6 — Роботы как ячейки в одномерном клеточном автомате пространства роботов (Robot-Space Cellular Automaton)

Каждый агент (робот) представим клеткой c_i одномерного клеточного автомата. Каждая клетка имеет окрестность $\{c_{i-1}, c_i, c_{i+1}\}$. Пусть желаемая формация задана функцией $f(x)$, которая известна одному из агентов c_{seed} и каждый агент c_i знает свои координаты p_i. *Желаемое отношение* $r_{i\to j,des}$ между c_i и c_j определяется как вектор с началом в p_i и концом в точке пересечения $f(x)$ и окружности радиуса R с центром в p_i, R – это желаемое расстояние между агентами. Используя данные сенсоров, агент выясняет *действительное отношение* $r_{i\to j,act}$. Данные о желаемой формации, заданной через $f(x)$, распространяются локально от c_{seed} к соседям и от них далее. Все агенты вычисляют свои желаемые и действительные отношения с соседями и движутся таким образом, чтоб минимизировать $\|r_{i\to j,des} - r_{i\to j,act}\|$. Агенты (наземные роботы) могут сохранять как уже созданный строй, так и создавать новый. Демонстрируется работа алгоритма на примере колесных роботов, распознающих друг друга по цветным штрих-кодам и общающихся по радиосети XBee.

В работе [81] для решения сходной проблемы применен иной подход. Желаемый строй агентов задан графом и задача построения строя сводится к деформации взвешенного графа, описывающего исходное расположение агентов, к желаемому виду, с учетом ограничений по возможностям связи агентов друг с другом (1.7). При этом необходим поиск изоморфизма между желаемым и существующим графом строя. Координаты агентов с номером i задаются как обыкновенное дифференциальное уравнение вида (т.н. интегратор первого порядка)

$$\dot{q}_i = u_i,$$

где q_i – координаты агента, а u_i – его скорость, являющаяся в данном случае управлением.

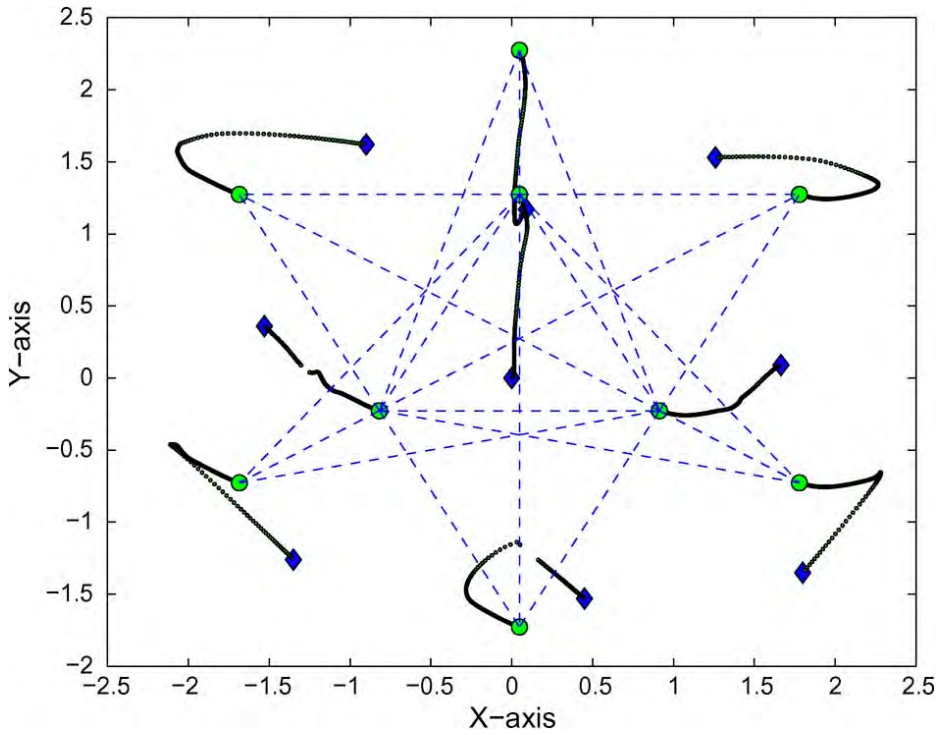


Рисунок 1.7 — Траектории агентов, занимающих желаемое положение

Статья [82] описывает построение строя роботов во многом аналогично ранее цитированной статье, однако, предлагаемый в работе подход корректно обрабатывает ситуации возможной поломки, присоединения новых агентов к строю и смены положения цели, работоспособность подхода проверена при помощи компьютерного моделирования. Также проанализирована работа алгоритма в условиях введения ошибки измерений и при наличии препятствий.

В статье [50] также решается задача управления формациями, а именно задача обобщения алгоритма равноудалённого расположения агентов на заданном фиксированном отрезке. Показана связь между задачей расположения на отрезке

и классической задачей достижения консенсуса. Поведение агентов здесь описывается, из соображений большей физичности модели, интеграторами второго порядка:

$$\ddot{q}_i = u_i,$$

где q_i – координаты агента, а u_i – его ускорение, являющаяся в данном случае управлением.

Работа [83] описывает построение такого строя БПЛА, который позволял бы поддерживать сеть связи БПЛА в максимально связном состоянии. Положения агентов задаются системой дифференциальных уравнений и задача рассматривалась с точки зрения теории устойчивости.

Довольно большой обзор по задачам построения строя приведен в [84].

1.2.8 Модели системы связи

В этом направлении существуют два (возможно пересекающихся) класса агентных моделей. Один из них описывает построение реального протокола связи внутри телекоммуникационной сети, а второй предназначен для компьютерных симуляций телекоммуникационных сетей.

В одном из ранних обзоров [85] перечисляются различные многоагентные модели построения сетей связи. В качестве агентов рассматриваются, естественно, сами узлы телекоммуникационной сети (рис. 1.8), при этом для таких агентов также может применяться модель BDI. Приводятся примеры распределенного и децентрализованного управления сетью, муравьиных алгоритмов маршрутизации и прочие решения, основанные на агентном подходе. Многие из них сохраняют актуальность и развиваются и по сей день.

В статье [86] было, вероятно, впервые описано применение многоагентного подхода к моделированию сети связи. В статье указывалось, что такие традиционные модели телекоммуникационной сети, как теория очередей, сети Петри или исчисление процессов, не вполне подходят для представления мобильных сетей или основанных на применении политик (policy-based) сетей, так как приводят к сложным и трудноразрешимым уравнениям, игнорируют индивидуальное поведение узлов и требуют задания ограниченного числа возможных состояний сети. Моделировалась система связи с сообщениями трех классов по QoS и пятью типами поведения узлов, включающими поведение узлов с динамической и статической маршрутизацией, а также разными способами построения очереди сообщений.

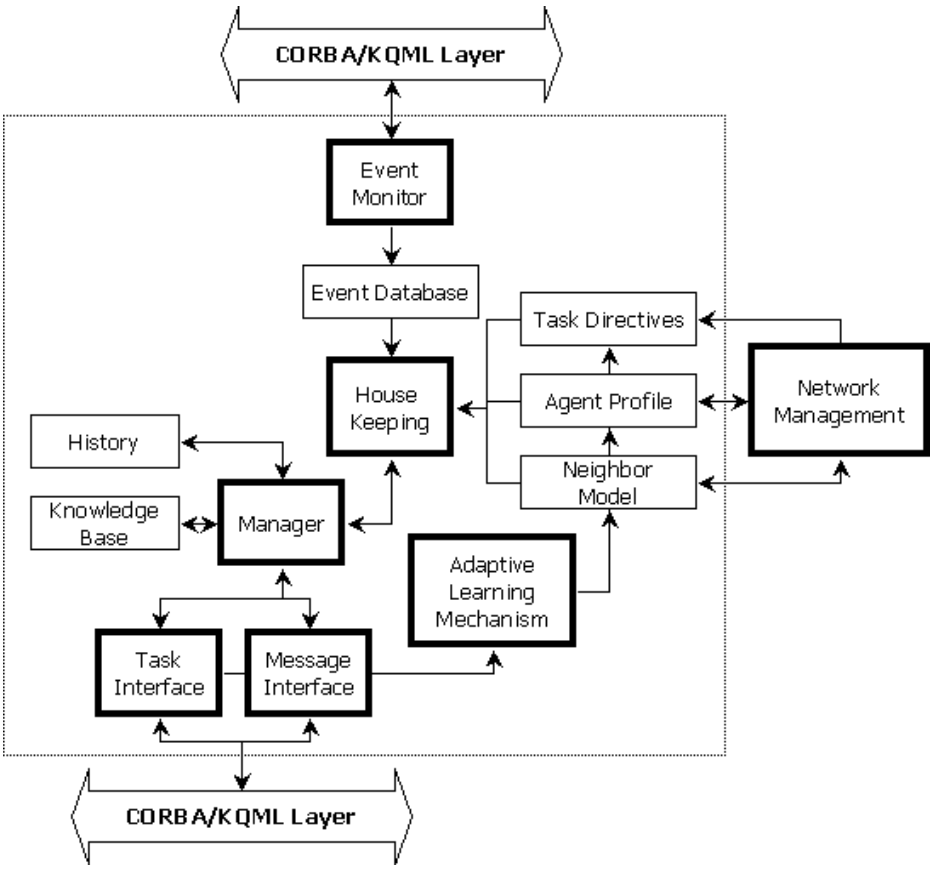


Рисунок 1.8 — Пример структуры агента телекоммуникационной сети

В связи с появлением концепции «интернета вещей» многоагентные системы как основа для разработки [87] и моделирования [88] сети связи становятся все более актуальными. В этом случае реальным прототипом агентов являются смартфоны, встраиваемые сетевые устройства и т.п., оснащенные специальными программными компонентами, позволяющими отображать в Web данные, задачи и службы всей системы агентов.

В работе [16] решается задача, во многом аналогичная задаче построения строя агентов, однако в ней с помощью графа задается не пространственные взаимоотношения между агентами, а желаемая форма сети связи. Агенты, обмениваясь сообщениями о фрагментах желаемой системы связи, организуют каналы связи со своими соседями в соответствии с полученными данными так, чтоб разница между желаемым и возможным для построения фрагментом графа связи была минимальным.

1.2.9 Модели боевых действий

Многоагентные модели применяются при моделировании боевых действий, особенно современных локальных конфликтов (так называемых stability operations), для которых оказались не применимыми традиционные модели,

оперирующие, главным образом, некоторыми интегральными показателями для больших военных соединений. С конца 1990-х – начала 2000 гг. получили определенную известность среды моделирования конфликта ISAAC и EINSTein, разработка US Marine Corps Combat Development Command. В монографии [89] дано подробное описание алгоритмов и интерфейсов вышеуказанных систем, а также исследованы различные феномены типа самоорганизующейся тактики агентов.

Также известны, например, модель Map Aware Non-uniform Automata (MANA), разработанная Operations Analysis group в Defence Technology Agency (DTA), Новая Зеландия или же Pythagoras – разработка Marine Corps Warfighting Laboratory министерства обороны США [90; 91]. Кроме того, конфликты моделируют и в средах агентного моделирования общего назначения типа NetLogo.

Помимо упомянутых систем, автором настоящего обзора была разработана программа моделирования движения и боевых действий иерархически организованных агентов «Бокоход» (номер свидетельства о государственной регистрации Роспатентом 2016615934), которая имела в своей основе алгоритм, близкий к описанному в [89]. Данный алгоритм изложен в работе [92], и, в отличие от ранее упомянутых алгоритмов ISAAC, сфокусирован, главным образом, на конфликте агентов в условиях неизвестного и, возможно, постоянно изменяющегося окружения, состоящего из ячеек разной степени проходимости. Для этой цели в программе «Бокоход» реализован поиск локально-оптимальных маршрутов агентов. Также используется особый математический формализм, в котором агент не является отдельной от среды сущностью, а является частью состояния ячейки world-space cellular automaton, моделирующего окружающую среду. В результате этого достигается определенный прирост производительности программы. При этом если имеется несколько агентов, претендующих на одну ячейку, ячейка разыгрывается между ними по определенному алгоритму.

1.2.10 Социально-экономические науки

Наконец, многоагентное моделирование применяется и в социально-экономических науках. Применение таких моделей в экономике стало распространенным, по всей вероятности, после мирового финансового кризиса 2008 г., как утверждается в монографии [93]. Преимущество агентного моделирования в том, что оно хорошо отражает сложность взаимоотношений различных людей,

домохозяйств и фирм в динамике. Агентное моделирование позволяет непосредственно связать микроуровень и макроэкономическую ситуацию и не требует равновесия экономической системы.

При этом следует отметить, что обзорные статьи на тему применения агентного моделирования в экономике появились намного раньше, как, например статья [94], изданная в виде отчета еще в 2005 г.

Агентные модели используются и для описания социальных сетей, как естественно и давно существующих в обществе, так и для компьютерных социальных сетей. Например, работа [95] предлагает многоагентную модель добавления «друзей» в социальной сети Facebook, построенную по результатам анализа данных студентов Калифорнийского технологического института (Caltech).

В работе [96] приведен большой обзор (по сути, учебник) моделей динамики мнений в социальных сетях. Рассматриваются предложенные в 1950–90 гг. модель эволюции мнений Френча–де Гроота

$$x(k+1) = Wx(k), k = 0, 1, \ldots,$$

где $W = (w_{ij})$ – $n \times n$ матрица весов влияния, $w_{ij} \geqslant 0$, модель Абельсона

$$\dot{x}(t) = -L[A]x(t),$$

где $L[A]$ – матрица Кирхгофа, модель Фридкина-Джонсена

$$x(k+1) = \Lambda W x(k) + (I - \Lambda)u,$$

где $\Lambda = diag(\lambda_1, \ldots, \lambda_n)$, где $\lambda_i \in [0, 1]$ означает восприимчивость $i-$го агента к процессу социального влияния, u – постоянный вектор предубеждений агентов, и Тэйлора

$$\dot{x}_i(t) = \sum_{j=1}^{n} a_{ij}(x_j(t) - x_i(t)) + \sum_{k=1}^{m} b_{ik}(s_k - x_i(t)),$$

где $A = (a_{ij})$ – $n \times n$ матрица весов, $B = (b_{ik})$ – $n \times m$ неотрицательная матрица «констант убедительности». Также исследуется взаимосвязь этих моделей с современными многоагентными моделями.

В статье [97] исследуется поведение группового консенсуса, который зависит от нескольких утверждений, одно из которых является ложным. Показывается, как существование логических ограничений на убеждения влияет на

коллективную сходимость к общей системе убеждений и, напротив, на то, как набор произвольно связанных убеждений, разделяемых меньшинством, может стать разделяемым большинством. В качестве примера рассматривается динамика мнений жителей США относительно ввода войск в Ирак при изменении их уверенности в утверждениях

1. У Саддама Хусейна есть запас оружия массового уничтожения.
2. Оружие массового уничтожения у Саддама Хусейна является реальным и представляет опасность для региона и всего мира.
3. Упреждающее вторжение в Ирак является справедливой войной.

Математической моделью динамики убеждений является обобщение модели Фридкина–Джонсена.

$$X(k+1) = AWX(k)C^{\top} + (I - A)X(0), k = 0,1, \ldots,$$

где $X(k)$ – $n \times m$ матрица позиций n индивидуумов по m взаимозависимым вопросам; $W = (w_{ij})$ – $n \times n$ матрица весов влияний; $A = (a_{ij})$ – диагональная $n \times n$ матрица со значениями на главной диагонали $a_{ii} = 1 - w_{ii}$, $i = \overline{1,n}$; C – $m \times m$ матрица зависимостей между вопросами; I – $n \times n$ тождественная матрица. В данной модели консенсус может не достигаться.

Глава 2. Модель движения и взаимодействия иерархически организованных агентов на основе клеточного автомата

2.1 Термины и определения

Определение 5. *Агент – это реальная или виртуальная автономная сущность, работающая во внешней среде, способная воспринимать и действовать в этой среде. Агент может общаться с другими агентами, проявлять независимое поведение, которое может быть рассматриваемо как следствие его знаний, взаимодействия с другими агентами и целей, которых он должен достичь [98].*

Часто то, чем агенты воспринимают внешнюю среду, называют сенсорами, а то, чем агенты воздействуют на среду – эффекторами (или актуаторами). Под данное определение подпадает довольно большое количество сущностей – от компьютерных программ и роботов до людей.

Таким образом, многоагентная система – это сложная система, которая состоит, в основном, из агентов. Агентные модели представляют собой своего рода микромасштабную модель, которая имитирует одновременные действия и взаимодействия нескольких агентов в попытке воссоздать и предсказать появление сложных явлений. Этот процесс является результатом выхода из нижнего (микро) уровня систем на более высокий (макро) уровень.

В одном из наиболее полных учебников по искусственному интеллекту [99, P. 47] приводится следующая классификация агентов на основе их степени их «интеллектуальности», которая, по мнению авторов, охватывает все агентные системы:

- Простые рефлекторные агенты, которые осуществляют действие на основе своего текущего восприятия. Они могут эффективно действовать, только лишь в случае полностью наблюдаемого окружения.
- Основанные на модели рефлекторные агенты. Данный тип агентов имеет внутри себя модель ненаблюдаемой в настоящий момент части окружения, например, историю предыдущих восприятий и действий агента. Такие агенты могут действовать в частично наблюдаемом окружении.
- Основанные на цели агенты. Это основанные на модели агенты, имеющие информацию о цели, т.е. перечень желательных состояний.

– Основанные на полезности агенты. Эти агенты при функционировании максимизируют функцию полезности, которая позволяет сортировать возможные состояния, в которых может оказаться агент, по степени их полезности.
– Обучающиеся агенты. Данная категория агентов может работать в изначально неизвестных средах и постепенно накапливать сведения о наиболее эффективных стратегиях поведения.

С понятием агента связаны следующие важные для дальнейшего изложения концепции [99]:

– Показатель эффективности агента – это критерии, определяющие, насколько успешным является агент.
– Восприятие (percept) – это состояние сенсоров агента в определенный момент времени.
– Поведение агента (behavior of agent) – это действие, которое агент выполняет после любой последовательности восприятий.
– Последовательность восприятий (percept sequence) – это история всего, что агент воспринимал вплоть до настоящего времени.
– Агентная функция (agent function) – это отображение из последовательности восприятий во множество возможных действий агента.

Агентная модель может быть реализована в виде компьютерной программы самым разным способом, например, агенту может соответствовать объект какого-нибудь языка программирования. В настоящей диссертации, многоагентная система реализована в виде клеточного автомата, в котором агентам соответствуют объекты языка C++, одно из свойств которых содержит адрес ячейки, а ячейки являются элементами двумерного контейнера std::vector.

Модель убеждений, желаний и намерений (belief, desire, and intention model, BDI) – популярная техника проектирования агентов [100]. В этой модели «убеждения» – это известные агенту сведения о мире, а также правила вывода новых сведений из доступных, «желания» соотносятся с планируемыми задачами агента, а «намерения» – с действиями, которые агент должен выполнить для осуществления своих задач. В эту модель включают также «события» – то, что может изменить «убеждения», «желания» и «намерения». Классические модели BDI используют темпоральную Computational Tree Logic CTL*, однако следует

понимать, что буквальное (а не концептуальное) следование таким конструкциям может существенно затруднить разработку программного обеспечения в силу трудности реализации CTL*.

Определение 6. *Однородной структурой со входами и выходами (далее – ОСВВ, клеточным автоматом) [101, с. 159], называется набор* $(\mathbb{Z}^k, \mathscr{W}, V)$*, где* $\mathscr{W}$ *– конечный автомат с* n *основными и* p *боковыми входными каналами,* m *основными и* p *боковыми выходными каналами,* l *состояниями, помещенный в каждом узле целочисленной решетки* $\mathbb{Z}^k$ *и задаваемый системой уравнений:*

$$\begin{cases} y(t) = \varphi_1(x(t), b(t), \mathfrak{s}(t)), \\ a(t) = \varphi_2(\mathfrak{s}(t)), \\ \mathfrak{s}(t+1) = \varphi(x(t), b(t), \mathfrak{s}(t)), \end{cases}$$

$$\varphi_1 : \mathscr{E}^n \times \mathscr{E}^p \times \mathscr{S} \to \mathscr{E}^m,$$

$$\varphi_2 : \mathscr{S} \to \mathscr{E}^p,$$

$$\varphi : \mathscr{E}^n \times \mathscr{E}^p \times \mathscr{S} \to \mathscr{E}^m,$$

где $x = (x_1, \ldots, x_n)$*,* $b = (b_1, \ldots, b_p)$*,* $a = (a_1, \ldots, a_p)$*,* $y = (y_1, \ldots, y_m)$ *обозначают соответственно основной вход, боковой вход, боковой выход и основной выход,* $\mathfrak{s}$ *– состояние автомата,* $t = \overline{0,N}$ *– такт дискретного времени,* $\mathscr{E}$ *– входной и выходной алфавит,* $\mathscr{S}$ *– множество состояний автомата* $\mathscr{W}$*,* $V = (\alpha_1, \ldots, \alpha_p)$*,* $\alpha_i \in \mathbb{Z}^k$*,* $i = \overline{1,p}$*, – шаблон соседства: для автомата в узле* α *каждый* $\alpha_i \in V$ *определяет конечный автомат с координатами* $\alpha + \alpha_i$*, с* $i-$*м боковым выходным каналом которого соединен* $i-$*й боковой входной канал автомата в узле* α*.*

Агентной функции соответствует здесь локальная функция перехода клеточного автомата, восприятию – боковой и основной входы, а действию агента – боковой и основной выходы КА.

Определение 7. *Назовем ландшафтом в момент времени* t *множество классов клеток определенного подмножества замощения* $\mathcal{L}(\Omega_h, l) = \{u_{ij} | u_{ij} = \mathrm{u}(t, \omega_{ij}), \omega_{ij} \in \Omega_h\}$*, такого что* u *принимает на* $\{\omega_{ij} | i = \overline{1,n}, j = \overline{1,m}\}$ *не более* l *значений, причем к классу* i *принадлежит* N_i *клеток, то есть* $\sum_{i=1}^{l} N_i = M$*,* $M = |\mathcal{L}(\Omega_h, l)|$*.*

Отметим, что для ландшафтов, генерируемых для тестирования алгоритмов обхода препятствий и поиска кратчайшего пути, клетки ландшафта будут делиться на классы по максимально возможной скорости пересечения клетки.

Введем следующие характеристики ландшафта, известные из ландшафтной экологии (см., например, [102–104]).

Определение 8. *Конфигурационная энтропия ландшафта $\mathcal{L}(\Omega_h, l)$ определяется как*

$$S(\mathcal{L}(\Omega_h, l)) = -\sum_{i=1}^{l} \frac{N_i}{M} \ln \frac{N_i}{M}.$$

Определение 9. *Total Edge (TE) определяется как общее количество соприкосновений сторон клеток в $\mathcal{L}(\Omega_h, l)$, принадлежащих к разным классам. Будем далее обозначать Total Edge ландшафта $\mathcal{L}(\Omega_h, l)$ как $TE(\mathcal{L}(\Omega_h, l))$.*

Определение 10. *Total Edge Density (TED) для ландшафта $\mathcal{L}_{n\times m}(l)$ определяется как отношение $TE(\mathcal{L}(\Omega_h, l))$ к общему количеству клеток $\mathcal{L}(\Omega_h, l)$:*

$$TED(\mathcal{L}(\Omega_h, l)) = TE(\mathcal{L}(\Omega_h, l))/M.$$

Определение 11. *Расстоянием Чебышева между ячейками $z_1 = (i,j)$, $z_2 = (k,l) \in \mathbb{Z}^2$ назовем выражение*

$$l_\infty(z_1,z_2) = \|z_1 - z_2\|_\infty = \max(|i-k|, |j-l|).$$

Определение 12. *Евклидовым расстоянием между ячейками $z_1 = (i,j)$, $z_2 = (k,l) \in \mathbb{Z}^2$ назовем выражение*

$$\|z_1 - z_2\| = \sqrt{|i-k|^2 + |j-l|^2}.$$

В работе далее будут использоваться следующие стандартные обозначения: $\mathbb{R}$, $\mathbb{Z}$ и $\mathbb{N}$ – множества вещественных, целых и натуральных чисел соответственно, $\mathbb{R}_{\geqslant 0}$, $\mathbb{Z}_{\geqslant 0}$ – множества неотрицательных вещественных и целых чисел, $\mathbb{R}^n$ – пространство n-мерных векторов с вещественными координатами, $\|\cdot\|$ – произвольная (например, евклидова) норма в $\mathbb{R}^n$.

2.2 Алгоритмы построения случайных ландшафтов с фиксированными характеристиками

2.2.1 Постановка задачи. Определения

Широко применяемыми характеристиками ландшафта (метриками) являются конфигурационная энтропия ландшафта, Total Edge и Total Edge Density [103; 104]. Конфигурационная энтропия характеризует количественное соотношение между элементами ландшафта разных классов, а метрики Total Edge и Total Edge Density характеризуют, насколько сильно квадраты разных классов перемешаны в ландшафте.

Вышеупомянутые метрики легко вычислить и существует большое количество программных продуктов для их расчета. Однако, для тестирования представляемых далее в диссертации алгоритмов обхода препятствий и получения количественных характеристик этих алгоритмов, например, зависимости эффективности алгоритма от свойств ландшафта, возникает обратная задача – построить ландшафт $\mathcal{L}(N,l)$ с заданными конфигурационной энтропией ландшафта, Total Edge и Total Edge Density. При этом построение ландшафта должно быть достаточно быстрым, так как ландшафт необходим для накопления статистики о времени прохождения по нему агентов, таким образом, необходимо за разумное время перебрать большое количество различных ландшафтов с одинаковыми характеристиками.

Разумеется, можно просто генерировать случайные ландшафты, а потом фильтровать те из них, которые подходят по метриками, но даже для ландшафта в 50 на 50 клеток для этого могут потребоваться многие дни машинного времени.

В данном разделе будут описаны алгоритмы построения ландшафтов с заданными с заранее заданными метриками, а также выяснению взаимосвязи этих метрик, так как в работе [102] было показано, что ландшафты с одинаковой конфигурационной энтропией могут выглядеть совершенно по-разному и иметь совершенно разный Total Edge.

Для генерации ландшафта с заданным Total Edge используется клеточно-автоматный подход. Клеточные автоматы применяются при создании случайных

ландшафтов в компьютерных играх. Однако, как следует из работ [105; 106], при процедурной генерации текстур и ландшафтов задача создания текстуры с заданными характеристиками типа энтропии или Total Edge обычно не ставится.

2.2.2 Генерация ландшафта с заданной конфигурационной энтропией

В настоящем разделе решим задачу эффективно построить ландшафт $\mathcal{L}(N,l)$ по заданной конфигурационной энтропии $S(\mathcal{L}(N,l))$, то есть подобрать вектор $\mathbf{V} = (N_1, \ldots, N_l)$ количеств клеток каждого класса i так, чтобы

$$\sum_{i=1}^{l} N_i = N, \tag{2.1}$$

$$S(\mathbf{V}) = -\sum_{i=1}^{l} \frac{N_i}{N} \ln \frac{N_i}{N} = S(\mathcal{L}(N,l)). \tag{2.2}$$

Простейшим способом была бы случайная генерация большого количества векторов $\mathbf{V} = (N_1, \ldots, N_l)$ случайным образом:

- сгенерировать $l-1$ случайных целых чисел $1 \leqslant \xi_i \leqslant N$, $1 \leqslant i \leqslant l-1$,
- отсортировать ξ_i по неубыванию и полагать, что $N_i = \xi_i - \xi_{i-1}$, $\xi_0 = 0$, $\xi_l = N$, в результате будет выполнено условие (2.1),
- для выполнения условия (2.2) сгенерировать как можно больше V и отобрать из них такой V_0, что $|S(V_0) - S(\mathcal{L}(N,l))| \to \min$.

Несмотря на очевидность, алгоритм абсолютно неэффективен и уже при $l = 9$ практически неприменим на обычном современном персональном компьютере.

Положим $N_i = \beta^{i-1} N_1$, $N_1 \leqslant N$, $\beta \geqslant 0$. Поскольку при перестановке компонент $\mathbf{V}$ очевидно, что $S(\mathbf{V})$ не меняется, то имеет смысл рассматривать лишь невозрастающую последовательность N_i, тогда $0 \leqslant \beta \leqslant 1$. В случае, если $\beta = 0$, будем полагать что $\beta^0 = 1$. Тогда, используя известные свойства геометрической прогрессии, перепишем (2.1) как

$$N = \sum_{i=1}^{l} N_i = \sum_{i=1}^{l} \beta^{i-1} N_1 = N_1 \frac{1-\beta^l}{1-\beta}, \tag{2.3}$$

$$\tag{2.4}$$

а условие (2.2) как

$$S(\mathbf{V}) = -\sum_{i=1}^{l} \frac{\beta^{i-1}N_1}{N} \ln \frac{\beta^{i-1}N_1}{N} = S(\mathcal{L}(N,l)). \tag{2.5}$$

Преобразуем (2.5) с учетом (2.3), получив:

$$S(\mathbf{V}) = -\sum_{i=1}^{l} \frac{\beta^{i-1}N_1}{N}(\ln \beta^{i-1} + \ln N_1 - \ln N) = -\frac{N_1 \ln \beta}{N} \sum_{i=1}^{l} \beta^{i-1}(i-1) + \ln \frac{N}{N_1}. \tag{2.6}$$

Числа $\beta^{i-1}(i-1)$ образуют арифметико-геометрическую последовательность, поэтому, справедливо соотношение

$$\sum_{i=1}^{l} \beta^{i-1}(i-1) = \frac{\beta - \beta^{1+l} - l\beta^l + l\beta^{1+l}}{(1-\beta)^2} = \frac{\beta(1-\beta^l)}{(1-\beta)^2} + \frac{l\beta^l(\beta-1)}{(1-\beta)^2}, \tag{2.7}$$

следовательно, из (2.3) следует, что:

$$N_1 \sum_{i=1}^{l} \beta^{i-1}(i-1) = \frac{\beta}{1-\beta} N_1 \frac{1-\beta^l}{1-\beta} - \frac{N_1 l \beta^l}{1-\beta} = \frac{N\beta - N_1 l \beta^l}{1-\beta}. \tag{2.8}$$

Подставляя (2.8) в (2.6), получим, что

$$S(\mathbf{V}) = -\frac{\beta(1-\beta^l) - (1-\beta)l\beta^l}{(1-\beta)(1-\beta^l)} \ln \beta + \ln \frac{1-\beta^l}{1-\beta}. \tag{2.9}$$

Из (2.3) следует, что

$$N_1 = N \frac{1-\beta}{1-\beta^l}. \tag{2.10}$$

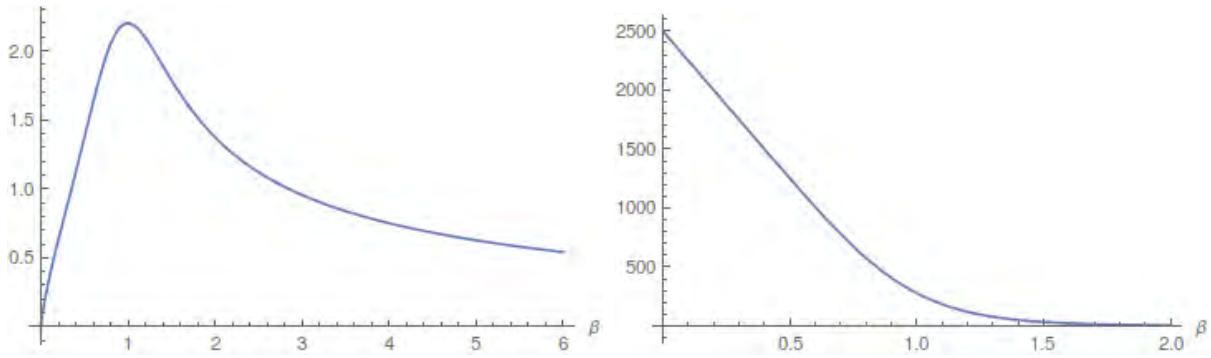
Графики вышеприведенных зависимостей показаны на рисунке 2.1.

Алгоритм построения векторов $\mathbf{V} = (N_1, \ldots, N_l)$ по энтропии S следующий

1. Решить уравнение (2.9) относительно β.

$$S = -\frac{\beta(1-\beta^l) - (1-\beta)l\beta^l}{(1-\beta)(1-\beta^l)} \ln \beta + \ln \frac{1-\beta^l}{1-\beta}, \tag{2.11}$$

2. По найденному решению $0 \leqslant \beta \leqslant 1$ определить $N_1 = N(1-\beta)(1-\beta^l)^{-1}$.
3. Построить вектор $\mathbf{V}_0 = (N_1, \beta N_1, \ldots, \beta^{l-1}N_1)$, округлить компоненты $\mathbf{V}_0$ до целых и случайно переставить их, получив вектор $\mathbf{V}$, при этом необходимо проследить, чтоб сумма всех компонент $\mathbf{V}$ равнялась N.



а) Зависимость энтропии S от β,$l = 9$

б) Зависимость N_1 от β, $l = 0$, $N = 2500$

Рисунок 2.1 — Зависимости S, N_1 и β

Алгоритм построения ландшафта $\mathcal{L}(N,l) = \{L_{ij}\}$ по вектору $\mathbf{V}$ таков:

1. Всем клеткам $\mathcal{L}(N,l)$ присваивается класс -1.
2. По равномерному закону генерируется пара целых чисел $(i,j) \in I$.
3. По равномерному закону генерируется целое число $1 \leqslant k \leqslant l$, такое что $N_k \neq 0$.
4. Если класс L_{ij} равен -1, устанавливается, что класс L_{ij} равен k и $N_k := N_k - 1$.
5. Повторяется шаг 2, пока $\mathbf{V} \neq \mathbf{0}$.

Вид ландшафтов, генерируемых по вышеуказанному алгоритму, приведен на рис. 2.2, где обозначено $S = S(\mathcal{L}(48 \times 48{,}9))$, $TED = TED(\mathcal{L}(48 \times 48{,}9))$. С помощью вычислительного эксперимента было установлено, что временная

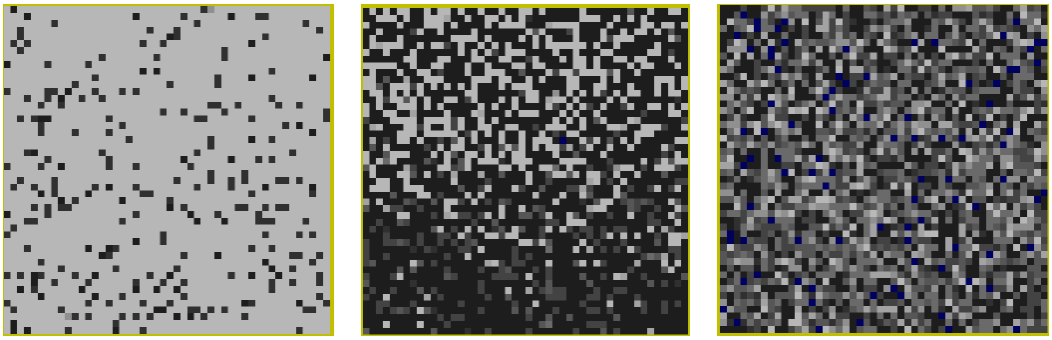


Рисунок 2.2 — Вид ландшафтов $\mathcal{L}(48 \times 48{,}10)$, сгенерированных по алгоритму раздела 2.2.2. Слева направо $S = 0.349119$, $TED = 0.40842$, $S = 1.1004$, $TED = 1.05773$, $S = 2.00013$, $TED = 1.63411$

вычислительная сложность алгоритма для прямоугольного ландшафта зависит от количества клеток N как $O(N^{3/2})$, а от количества классов клеток l – как $O(1)$.

2.2.3 Генерация ландшафта с заданным Total Edge

Предположим, что $\mathcal{L}(N,l) = \{L_{ij} | 1 \leqslant i \leqslant n, 1 \leqslant j \leqslant m\}$. Очевидно, что минимальное значение $TE(\mathcal{L}(N,l))$ будет у ландшафта $\mathcal{L}(N,l)$, в котором начиная с одного из углов последовательно размещены в одну строку клетки класса Cl_1, затем – клетки класса Cl_2 и т.д., причем если клетки одного класса заканчиваются, то строка этих клеток продолжается клетками другого класса. Если количество клеток класса k равно N_k, то при помещении клетки класса k в ландшафт устанавливается $N_k := N_k - 1$. Когда пройдена вся строка, то приступают к следующей, которую проходят в противоположном направлении.

Введем на ландшафте $\mathcal{L}(N,l)$ отношение строгого порядка «<» следующим образом

$$L_{ij} < L_{sr} = \begin{cases} i < s, \\ j < r, \quad i = s = 2k-1, 1 \leqslant k \leqslant [n/2]+1, \\ j > r, \quad i = s = 2k, 1 \leqslant k \leqslant [n/2]. \end{cases}$$

Отношение «<» естественным образом порождает операции нахождения предыдущей клетки $\mathrm{pred}(L_{ij}) = L_{sr}$, $L_{sr} < L_{ij}$ и не существует такого L_{pq}, что $L_{sr} < L_{pq} < L_{ij}$, последующей клетки $\mathrm{succ}(L_{ij}) = L_{sr}$, $L_{ij} = \mathrm{pred}(L_{sr})$. Обозначим класс клетки L_{ij} как $Cl(L_{ij})$.

Вышеупомянутый алгоритм заполнения ландшафта может быть описан как клеточный автомат, в котором начальное состояние s_{ij} любой клетки L_{ij}, $(i,j) \neq (1,1)$ равно -1, начальное состояние $s_{11} = \xi$, где $\xi \in \overline{1,l}$ – случайный номер класса клетки. Для каждого класса клеток Cl_k определено «оставшееся в запасе» количество клеток данного класса $N(Cl_k)$ в каждый такт функционирования клеточного автомата, а локальная функция перехода определяется как

$$s_{ij}(\varepsilon) = \begin{cases} Cl(\mathrm{pred}(L_{ij})), & \overset{Cl(\mathrm{pred}(L_{ij}))>0, N(Cl(\mathrm{pred}(L_{ij})))>0,}{i+j \neq 0, \zeta \leqslant 1-\varepsilon}, \\ \eta, & \eta \neq Cl(\mathrm{pred}(L_{ij})) \vee i = n \wedge j = m, \zeta > 1-\varepsilon \end{cases}$$

$$N(Cl(s_{ij})) := N(Cl(s_{ij})) - 1,$$

где $\zeta \in [0,1]$ – равномерно распределенная случайная величина, $\eta \in \overline{1,l}$ – равномерно распределенная случайная величина, $0 \leqslant \varepsilon < 1$. Заметим также, что максимальный TE у прямоугольного ландшафта $\mathcal{L}(n \times m,l)$ будет в том

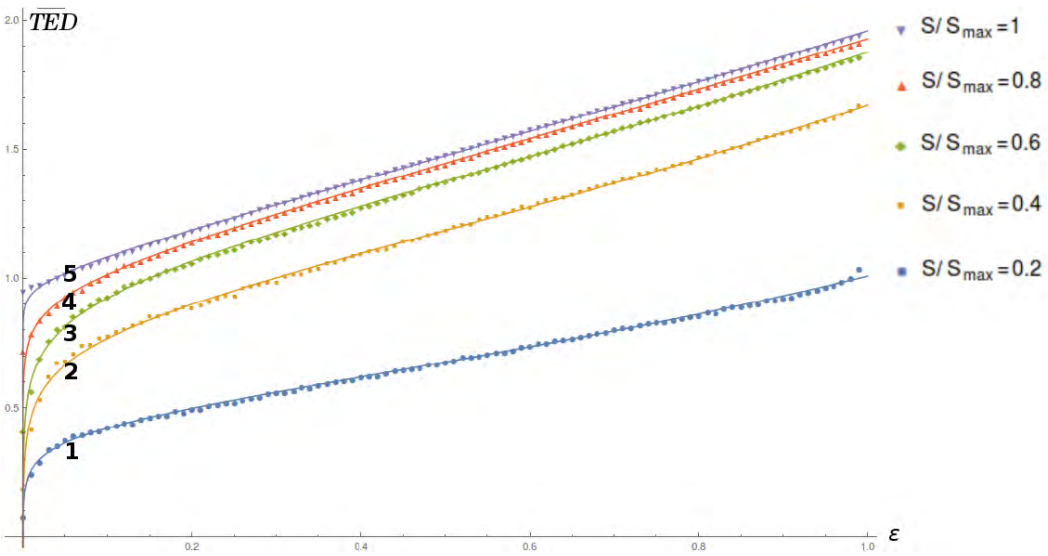


Рисунок 2.3 — Зависимость $\overline{TED}$ при разных значениях ε и S, $n = m = 48$, $l = 152$

случае, если все клетки ландшафта принадлежат к разным классам. Максимальная конфигурационная энтропия для такого ландшафта будет, очевидно, $S_{max} = S_{max}(\mathcal{L}(n \times m,l)) = \ln l$. Вычислительный эксперимент (см. рис. 2.3), повторяемый по 50 раз для каждого значения параметров, показал, что среднее значение TED зависит от ε практически линейно, причем вид такой зависимости мало изменяется при изменении числа классов l. Таким образом, возможно искать среднее значение $TED(\mathcal{L}(N,l))$ в виде

$$\overline{TED} = a_0(S/S_{max}) \ln((c_0(S/S_{max})\varepsilon^k)((1 - c_0(S/S_{max})\varepsilon))^{-1}) + b_0(S/S_{max}),$$

где a_0, b_0 – константы, зависящие от конфигурационной энтропии ландшафта. Для зависимостей, показанных на рис. 2.3, значения констант приведены в таблице 1. Созданные алгоритмом ландшафты $\mathcal{L}(48 \times 48, 9)$ приведены на рис. 2.4.

Таблица 1 — Значения констант для зависимости энтропии и TED

Номер	1	2	3	4	5
a_0	0.631706	1.04616	1.50153	3.92962	5.29811
b_0	0.902495	1.7169	2.53405	7.66127	10.9732
c_0	0.541963	0.489177	0.392038	0.188599	0.154264
k	0.0896779	0.105287	0.0763552	0.0183298	0.00576306

Временная вычислительная сложность этого алгоритма зависит от количества клеток N как $O(N)$, от количества классов клеток l – как $O(l)$.

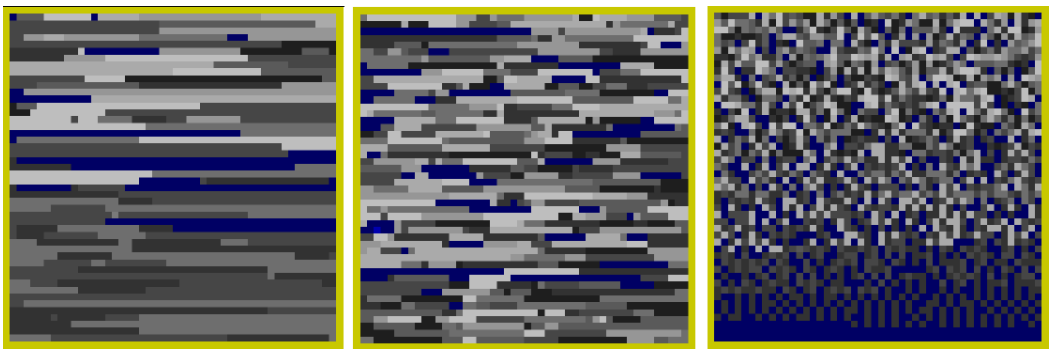


Рисунок 2.4 — Ландшафты при $S = 1.90026$, $\overline{TED} = 0.728733$, $\varepsilon = 0.05$, $S = 2.19722$, $\overline{TED} = 1.03125$, $\varepsilon = 0.2$, $S = 1.90026$, $\overline{TED} = 1.57552$, $\varepsilon = 0.9$

2.2.4 Генерация «естественных» ландшафтов

Ландшафт генерировался так, чтоб функция непроходимости $u : \mathbb{Z}^2 \to \mathbb{Z}$ имела бы локальные максимумы ровно в $N_{obst} = V_l = \beta^{l-1} N_1$ клетках. Такой способ построения ландшафта представился автору наиболее естественным в том отношении, что получалась «в основном проходимая» область с некоторыми труднопроходимыми участками. Обозначим как $N(cl)$ число не назначенных клеток класса cl, в начальный момент времени $N(cl) = V_{cl}$.

Примерные зависимости конфигурационной энтропии, TED и N_{obst} показаны на рисунке 2.5.

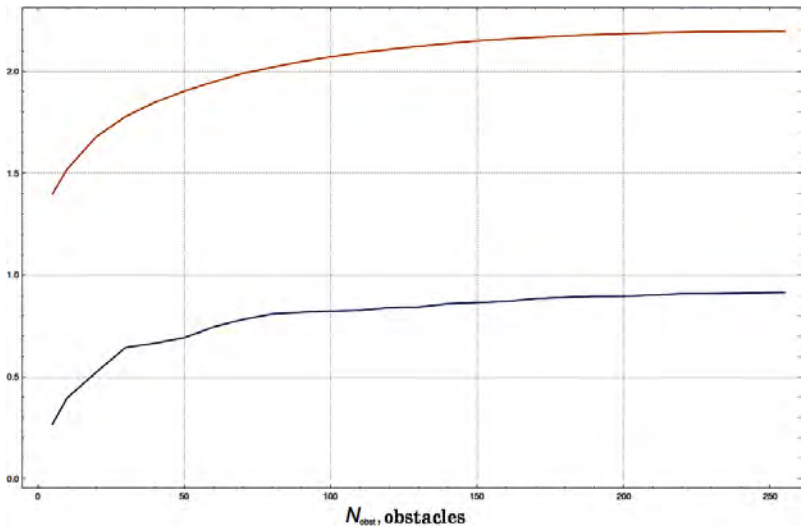


Рисунок 2.5 — Примерные зависимости конфигурационной энтропии, TED и N_{obst}.

Алгоритм построения ландшафта (также основанный на КА) вкратце был таков:

Шаг 1. Всем клеткам ландшафта присваивается класс -1.

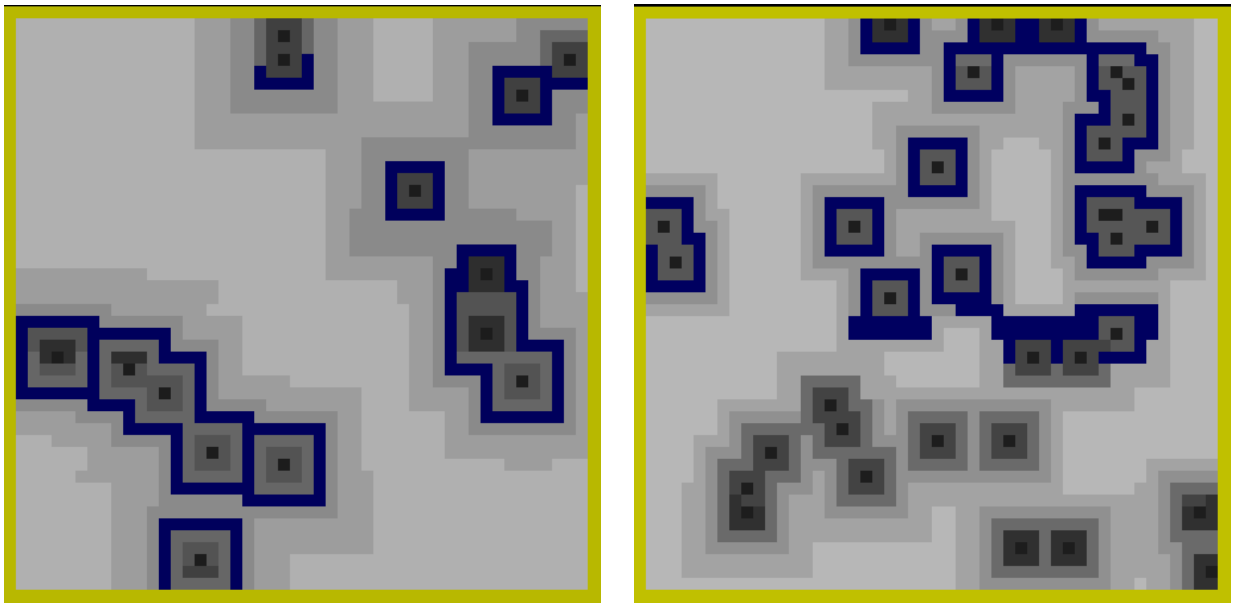
Шаг 2. По равномерному закону случайно размещается N_l клеток класса l.

Шаг 3. Определяется локальная функция перехода класса клетки (i,j) как $Cl_{ij} := cl$, $cl = Cl_{i'j'}$, где (i',j') – случайно выбранная клетка из окрестности $V_1(i,j)$, $N(cl) > 0$, если таковая существует. Если таких клеток нет, то устанавливается, что $Cl_{ij} := Cl_{ij} - 1$, пока классы не закончатся.

Шаг 4. Устанавливается $N(cl) := N(cl) - 1$ при каждом назначении клетке класса cl.

Шаг 5. Если $N(-1) > 0$ то перейти к шагу 1.

Примеры получившихся ландшафтов показаны на рисунке 2.6.



а) $S = 1.6$, $N_{obst} = 14$ б) $S = 1.8$, $N_{obst} = 33$

Рисунок 2.6 — Примеры ландшафтов, чем темнее клетка, тем труднее она проходится

2.2.5 Взаимосвязь энтропии и Total Edge

Предположим, что прямоугольный ландшафт из $n \times m$ клеток генерируется по алгоритму, изложенному в разделе 2.2.2. В ходе 50 проведенных вычислительных экспериментов выяснилось, что практически среднее значение TED не зависит от количества клеток в ландшафте, а зависит лишь от количества

классов клеток. Из эмпирических соображений (см. рис. 2.7) будем искать зависимость среднего значения $TED(\mathcal{L}(n \times m, l))$ от l и конфигурационной энтропии S в форме

$$\overline{TED} = a_1(l)\sigma(Sb_1(l)/\ln l) + a_1(l)/2,$$

где $\sigma(x) = (1 + e^{-x})^{-1}$, a_1, b_1 – неизвестные функции. Подбирая параметры по

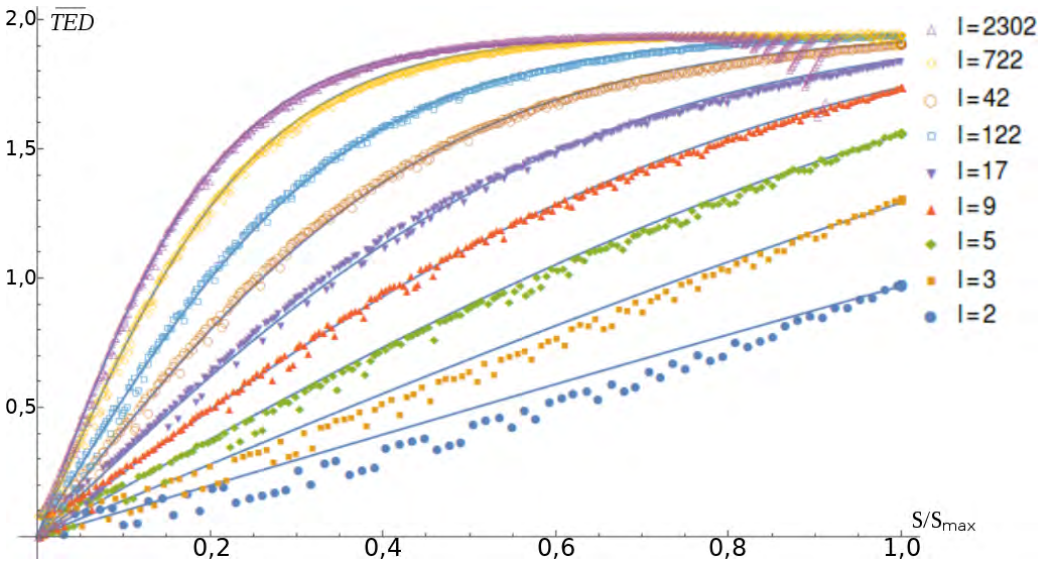


Рисунок 2.7 — Зависимость $\overline{TED}$ от S/S_{max} и количества классов клеток l ($n = m = 48$)

экспериментальным данным, можно получить (с коэффициентами детерминации $r^2 = 0.999999$ для a_1 и $r^2 = 1.0$ для b_1) аппроксимации

$$a_1(l) = 3.90302 + 5.25218(x - 10.9641)^{-2.19235},$$
$$b_1(l) = 1.15228 \ln(0.990096x - 1.18397).$$

Для решения уравнения (2.11) методом Галлея применялась функция `boost::math::tools::halley_iterate` из библиотеки Math Toolkit 2.5.0 из собрания библиотек для C++ Boost 1.62.0 [107]. Для получения случайных чисел с заданным распределением применялись шаблоны (`std::uniform_int_distribution` и т.п.) из Standard Template Library C++. В связи с использованием Boost применялся, разумеется C++ стандарта ISO/IEC 9899:2011, разработка алгоритмов генерации ландшафтов и визуализация ландшафтов велась в среде Qt 5.5.1 на основе свободного аналога разработанной автором программы моделирования движения и боевых действий иерархически организованных агентов «Бокоход» (регистрационный

номер Роспатента 2016615934, правообладатель АО «Концерн «Созвездие»). Статистический анализ данных вычислительных экспериментов и нахождение зависимостей между характеристиками ландшафта и различными параметрами проводился в среде Wolfram Mathematica 10.3 с помощью функции `NonlinearModelFit` и других.

2.3 Метод формального описания группового движения и конфликта агентов в содержащей препятствия среде

2.3.1 Непрерывная постановка задачи

Пусть $\Omega \subseteq \mathbb{R}^2$ и движение агента, ищущего кратчайший по времени маршрут $\mathbf{r}$ из точки $A \in \Omega$ в точку $B \in \Omega$ описывается уравнениями

$$\|\dot{\mathbf{r}}(t)\| = \mathrm{u}^c(t, \mathbf{r}(t)), \tag{2.12}$$

$$\mathbf{r}(0) = \mathbf{A}, \quad \mathbf{r}(t) = \mathbf{B}, \quad t \in [0,T], \tag{2.13}$$

$$T \to \min, 0 \leqslant T \leqslant T_{\max} \tag{2.14}$$

где $\mathbb{R}_{\geqslant 0}$ – множество вещественных неотрицательных чисел, $\mathrm{u}^c : \mathbb{R}_{\geqslant 0} \times \Omega \to \mathbb{R}_{\geqslant 0}$ – функция проходимости области Ω, определяющая максимально возможную скорость $\mathrm{u}^c(t, \boldsymbol{\rho})$ в точке $\boldsymbol{\rho} \in \Omega$ в момент времени t, $\|\cdot\|$ – евклидова норма,

$$T(\mathbf{r}) = \min\{t \in \mathbb{R}_{\geqslant 0} | \mathbf{r}(t) = B\}.$$

Из (2.12)–(2.14) можно получить выражение для $T : C([0,T_{\max}]; \mathbb{R}^2) \to \mathbb{R}_{\geqslant 0}$

$$T(\mathbf{r}) = \int_0^T \frac{\|\dot{\mathbf{r}}(t)\| \mathrm{d}t}{\mathrm{u}^c(t,\mathbf{r}(t))} = \int_{\widehat{\mathbf{AB}}} \frac{\mathrm{d}l}{\mathrm{u}^c(t(l),\mathbf{r}(t(l)))}, \tag{2.15}$$

где $\dot{\mathbf{r}}$ – производная $\mathbf{r}$.

Очевидно, из выражения (2.15) выразить явно T в общем случае невозможно (исключение составляет, например, случай, когда $\mathbf{r}$ является графиком некоторой функции $y \in C^2[a,b]$, т.е.

$$\mathbf{r} = \{(x,y(x)) | x \in [a,b]\},$$

а u^c не зависит от времени). Однако, можно получить различные свойства функционала T, доказать разрешимость задачи минимизации (2.18)–(2.20) в классе $\mathcal{Y} \subset W_p^1([0,T_{\max}];\mathbb{R}^2)$, построить алгоритм приближенного решения и дать оценки для него, что и будет сделано в главе 4.

Обозначим $\mathrm{u}^c_{\max} = \max\limits_{(t,\boldsymbol{\rho})\in[0,T_{\max}]\times\Omega} \mathrm{u}^c(t,\boldsymbol{\rho})$. Введем функцию непроходимости

$$\mathrm{u}(t,\boldsymbol{\rho}) = \frac{\mathrm{u}^c_{\max}}{\mathrm{u}^c(t,\boldsymbol{\rho})}.$$

Пусть $\Omega_h \supseteq \Omega$ – замощение Ω, $\Omega_h = \{\omega_{ij}|(i,j) \in \mathbb{Z}^2\}$, $\mathbf{r}_h \supseteq \mathbf{r}$ – клеточный маршрут, т.е. минимальное покрытие маршрута $\mathbf{r}$ клетками из Ω_h. Множество всех клеточных маршрутов, приближающих маршруты из $\mathcal{Y}$, обозначим как $\mathcal{Y}_h$.

Обозначим $\mathbf{r}_h[k]$ $k-$ю по порядку прохождения клетку $\mathbf{r}_h$. Функционал T имеет дискретный аналог

$$T_h(\mathbf{r}_h) = \sum_{(i,j)\in\mathbf{r}_h} u_{ij}(t)\|\mathbf{d}_{ij}\|, \tag{2.16}$$

где $u_{ij}(t) = \max\limits_{\boldsymbol{\rho}\in\omega_{ij}} \mathrm{u}(t,\boldsymbol{\rho})$ – непроходимость клетки (i,j) в момент времени t,

$$\mathbf{d}_{ij} = \mathbf{d}_k = \mathbf{r}_h[k+1] - \mathbf{r}_h[k], \quad \mathbf{r}_h[k] = (i,j).$$

При измельчении шага сетки h последовательность минимизирующих функционал (2.16) маршрутов $\{\mathbf{r}_h\}$ будет в некотором смысле сходиться к маршруту $\mathbf{r}$, минимизирующему функционал (2.15). Далее мы будем использовать обозначение

$$u_{ij}(t) = \mathrm{u}(t,(i,j)) = \max_{\boldsymbol{\rho}\in\omega_{ij}} \mathrm{u}(t,\boldsymbol{\rho}).$$

Обозначим $\mathbf{r}_h(t)$ ту клетку маршрута, в которой агент находится в момент времени t. Обратим внимание, что при $\chi \leqslant \kappa d(t)\mathrm{u}(t,\mathbf{r}_h(t))$ получается дискретный аналог (2.18)

$$\|\mathbf{r}_h(t+\chi) - \mathbf{r}_h(t)\| = \|\mathbf{d}(t)\| \left\lfloor \frac{\chi}{\kappa\|d(t)\|\mathrm{u}(t,\mathbf{r}_h(t))} \right\rfloor, \tag{2.17}$$

где $\mathbf{d}(t) \in \mathcal{D} = \{(i,j)|i,j = \overline{-1,1}\}$ – вектор направления движения агента, κ – количество секунд в одном кванте дискретного времени, $\lfloor x \rfloor$ – целая часть x.

Автором был сконструирован клеточный автомат, в котором агенты, движущиеся в соответствии с (2.17) искали кратчайший путь к точки назначения, минимизируя функционал (2.16) в некоторой своей окрестности, а

затем склеивали из получившихся фрагментов квазиоптимальный маршрут. Как альтернативный вариант, вероятность выбора агентом локально оптимального маршрута максимальна, и чем менее маршрут оптимальный, тем меньше вероятность его выбора.

При этом, в каждой окрестности агента, находящегося в клетке ω_{ij}, $U_o(i,j) = \{\omega_{ij} | \|(i,j)\| \leqslant o, \omega_{ij} \in \Omega_h\}$, $B \notin U_o(i,j)$, минимизировалось время до точки B_i пересечения границы $U_o(i,j)$ с отрезком, соединяющим местоположение агента и B.

Пусть у нас наличествует несколько агентов и заданы

1. Область $\Omega \subset \mathbb{R}^2$,
2. Множество агентов $Ag = \{ag_k | k = \overline{1,n}\}$.
3. Точки начала и конца движения агента $ag_k \in Ag$ A_k, $B_k \in \Omega$, $k = \overline{1,n}$

Траектория $k-$го агента $r_k : [0,T] \to \Omega$ отвечает условиям

$$\|\dot{r}_k(t)\| = \mathrm{u}^c(t, r_k(t)), \tag{2.18}$$

$$r_k(0) = A_k, \quad r_k(t) = B_k, \quad t \geqslant T_k, \quad T_k \leqslant T, \tag{2.19}$$

$$r_k(t) \neq r_l(t), \quad k \neq l, t \in [0,T], \tag{2.20}$$

где

$$T_k(r_k) = \min\{t \in \mathbb{R}_{\geqslant 0} | r_k(t) = B_k\},$$

$\|\dot{r}_k(t)\|$ – норма скорости $k-$го агента в момент времени t, $\mathcal{Y} \subset C([0,T];\Omega)$ – множество допустимых траекторий. В явном виде выписать функционалы T_k в общем случае невозможно. Отметим, что уравнение (2.18) означает, что агент движется в каждой точке области с максимальной (по норме) возможной в этот момент времени скоростью.

Необходимо найти такие траектории агентов, чтоб время $T = \max_{k=\overline{1,n}} T_k$ было минимальным, т.е. надо минимизировать все функционалы T_k, сопоставляющие траектории r_k агента ag_k продолжительность прохождения по этой траектории. Более формально, нужно решить задачу минимизации

$$T(r_1, \ldots, r_n) = \max_{k=\overline{1,n}} T_k(r_k) \to \min, \quad r_k \in \mathcal{Y}, k = \overline{1,n}. \tag{2.21}$$

2.3.2 Постановка задачи в терминах клеточных автоматов

Предположим, что дана система Ag, состоящая из агента третьего уровня e_{00}, агентов второго уровня e_{i0}, $i = \overline{1,n}$, и агентов первого уровня e_{ij}, $i = \overline{1,n}$, $j = \overline{1,n}$, т.е.

$$Ag = \{e_{00}\} \cup \{e_{ij} | i = \overline{1,n}; j = \overline{0,n}\}. \tag{2.22}$$

Агенты $e_{i_1 j_1}$, $e_{i_2 j_2}$ будут относиться к группе совместно перемещающихся агентов, если $i_1 = i_2$. Каждому множеству $\{e_{i_1 j} | j = \overline{1,n}\}$, $i_1 > 0$ ставится в соответствие агент $e_{i_1 0}$, а множеству $\{e_{i0} | i = \overline{1,n}\}$ – агент e_{00}.

Примером такой системы Ag является рота, состоящая из трех взводов, каждый из которых состоит из трех отделений. Здесь отделения будут элементами третьего, взводы – второго, а сама рота – первого уровня.

Пусть все эти элементы Ag перемещаются в течение определенного промежутка времени $[0,T]$ в некоторой области Ω, разбитой на квадраты ω_{ij}. Предполагается такая нумерация квадратов ω_{ij}, что i увеличивается сверху вниз от 1 с шагом 1, а j – слева направо от 1 с шагом 1. Для дальнейшего моделирования будем предполагать, что элементы второго и третьего уровня находятся примерно в центре масс многоугольника, вершинами которого являются соответствующие элементы первого уровня.

Положим также, что в начальный момент времени задано положение всех элементов Ag и что все время функционирования системы $[0,T]$ можно разбить на N равных непересекающихся интервалов длиной τ_0, $T = N\tau_0$, $N \in \mathbb{N}$. В течение каждого из таких интервалов положение всех элементов Ag постоянно, так что возможно перейти к дискретному времени $t = \overline{0,N}$. Элемент перемещается за время τ_0 на один квадрат по вертикали, по горизонтали и, для упрощения (чтобы рассматривать квадратные окрестности вместо круглых), по диагонали.

Будем считать, что с агентами из Ag враждуют и стремятся уничтожить агенты из аналогично определенного множества Ag'. Для координации своих действий агенты из Ag и Ag' могут пользоваться системами связи, задаваемыми графами Γ_1 и Γ_2 соответственно (рис. 2.8), где кружками и квадратами отмечены противоборствующие агенты первого уровня, сплошными и пунктирными линиями – графы системы связи Γ_1 и Γ_2 соответственно. Моделированию боевых действий таких агентов и посвящена настоящая статья.

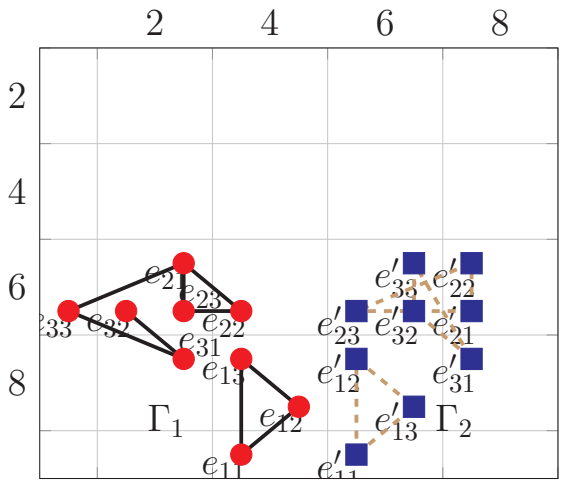


Рисунок 2.8 — Две конфликтующие группы агентов

2.3.3 Определение клеточного автомата

Положим в определении КА $k = 2$, $n = 1 + 1$, $m = 1$. Один основной вход x_1 необходим для изменения начальных состояний автоматов, что будет соответствовать получению приказов двигаться в то или иное направление от вышестоящего командования. Другой основной вход x_2 и соответствующий ему основной выход y_2 нужны для моделирования канала связи между автоматами. Для наших целей требуется несколько видоизменить вышеприведенное определение и вместо одного шаблона соседства V рассматривать вектор шаблонов соседства $(V^0, \ldots, V^5)$, причем количество боковых входов и выходов будет определяться как $p = \max(|V^0|, \ldots, |V^5|)$.

Таким образом, основные входы используются для управления всей системой и для получения автоматами информации о значительно удаленных автоматах, а боковые – для моделирования движения и воздействия на агента его непосредственной окрестности. Далее мы опишем алфавит и множество состояний автомата $\mathscr{W}$ и сконструируем функции φ_1, φ_2 и φ.

2.3.4 Состояния и алфавит

Множество ячеек разрабатываемого клеточного автомата – это $\mathbb{Z}^2$, а в качестве множества состояний будем рассматривать множество

$$\mathscr{E} = F \times Q \times S \times \mathcal{D} \times \mathbb{Z}^2 \times \mathbb{Z}^2 \times Q \times W \times U \times \mathcal{D} \times \mathcal{D} = \{(f, q, s; d, c, \tilde{c}, \tilde{q}; w, u; \zeta, \eta)\},$$

причем состояние автомата $\mathscr{W}_{ij}$ с координатами $(i,j) \in \mathbb{Z}^2$ будем обозначать как

$$\mathfrak{s}_{ij} = (f_{ij}, q_{ij}, s_{ij}; d_{ij}, c_{ij}, \tilde{c}_{ij}, \tilde{q}_{ij}; w_{ij}, u_{ij}; \zeta_{ij}, \eta_{ij}).$$

Множество $\mathscr{E}$ определено следующим образом:

$$\forall((i,j) \in \mathbb{Z}^2)\ f_{ij} \in \{0,1,2\}, \sum_{i,j \in \mathbb{Z}} f_{ij} < \infty,\ u_{ij}, w_{ij} = \overline{-1, N} \wedge w_{ij} \leqslant u_{ij}.$$

Значение $f_{ij} = 0$ соответствует отсутствию, $f_{ij} = 1$ или $f_{ij} = 2$ – наличию в квадрате ω_{ij} агента первого уровня, принадлежащего к первой или второй противоборствующей стороне. Величина $q_{ij} = (q^1_{ij}, q^2_{ij}) \in \mathbb{Z}^2$, $q^1_{ij} = \overline{1,n}$, $q^2_{ij} = \overline{0,n}$, является некоторым уникальным идентификатором агента $e_{q^1_{ij}, q^2_{ij}}$, находящегося в квадрате ω_{ij}. Иначе говоря, q^1_{ij} – порядковый номер группы, к которой принадлежит элемент, находящейся в квадрате ω_{ij}, а q^2_{ij} – порядковый номер элемента в этой группе. Если $f_{ij} = 0$, то $q_{ij} = (0,0)$, у агентов второго порядка $q^2_{ij} = 0$, $q^1_{ij} \neq 0$.

Значения u_{ij} соответствуют времени в тактах, нужному для преодоления квадрата ω_{ij}. Очевидно, что эта величина должна быть больше, если в квадрате находится труднопроходимая местность, и меньше – в противоположном случае. Будем полагать, что квадрат с ровной местностью проходится за один такт. Для совершенно непроходимого квадрата ω_{ij} устанавливается $u_{ij} = -1$. Величина w_{ij} соответствует тому, сколько времени в тактах элемент первого уровня уже проходит на текущий момент квадрат ω_{ij}. В момент попадания элемента в данный квадрат устанавливается $w_{ij} = 0$, каждый такт это значение увеличивается на единицу вплоть до $w_{ij} = u_{ij}$, когда считается, что элемент прошел квадрат и может переместиться на следующий.

Следовательно, в качестве ближайшей окрестности ячейки $(i,j) \in \mathbb{Z}^2$ мы возьмем окрестность Мура радиуса r, $V_r(i,j) = \{(i_1,j_1) | l_\infty((i,j),(i_1,j_1)) \leqslant r\}$. Восемь соседей клетки будем обозначать индексами из множества

$$\mathcal{D} = \{(-1,-1), (-1,0), (-1,1), (0,-1), (0,0), (0,1), (1,-1), (1,0), (1,1)\}.$$

Таким образом, $d_{ij} \in \mathcal{D}$ является направлением, в котором в данный момент движется агент в квадрате ω_{ij}.

Величина $\tilde{c}_{ij}$ – это полученная свыше клетка назначения агента, находящегося в квадрате ω_{ij}, $s_{ij} \in \{0,1,2,3\}$ соответствует тому, в каком состоянии (спокойном, атакующем, отступающем или идущем на помощь другим) находится агент, а также для сигнализации о своем состоянии другим агентам, $\tilde{q}_{ij} \in Q$ – для хранения идентификатора агента-цели, к которой по той или иной причине стремится данный агент. Величина $c_{ij} \in \mathbb{Z}^2$ отвечает за координату агента в том виде, в котором она известна самому агенту. В данной статье считается, что агент всегда знает свою правильную координату, в дальнейшем можно ввести погрешность определения координаты и моделировать, например, неоптимальное поведение агента из-за ошибок навигационного оборудования. Величины $(\zeta_{ij}, \eta_{ij}) \in \mathcal{D}$ используются исключительно для моделирования движения и самостоятельного смысла не имеют.

Для практической реализации будем считать, что в множестве клеток $\mathbb{Z}^2$ выделен круг (т.е. квадрат ввиду выбранной нами метрики) K из клеток с положительными координатами, не превосходящими L_K, для любой клетки (i,j) вне и на границах этого квадрата всегда $f_{ij} = 0$, $w_{ij} = 0$, $q_{ij} = (0,0)$, причем вне квадрата $u_{ij} = 0$, а на границе $u_{ij} = -1$.

Алфавит будет состоять из пустого слова ε и символов из $\mathfrak{s} \in \mathscr{E}$. Помимо этого на основных входах и выходах будут возможны коды сигналов $0,1,2,3$.

2.3.5 Входы и выходы

Так как агент может передвигаться в восьми направлениях, то боковых входов и выходов у автомата $\mathscr{W}$ должно быть минимум восемь. Пронумеруем эти входы и выходы, соответствующие направлениям, очевидным образом элементами из D (кроме $(0,0)$). К одному такому входу может быть присоединен только один выход от соседнего (находящегося на расстоянии 1) автомата. По этим входам автоматы передают своим соседям ту часть своего состояния, которая связаны с идентичностью моделируемого агента (код подразделения, атакует агент или отступает и т.п.). Фактически, передаче символов по этим «двигательным» каналам соответствует перемещение агента на одну клетку. Помимо

«двигательных» входов и выходов, необходимых для моделирования процесса перемещения, нужны также пары «информационных» входов и выходов, служащих для моделирования взаимодействия агента с ближайшей окрестностью. Нумерация этих выходов такова. Пусть $V_1(i,j)$ – некоторая окрестность (i,j). Номером выхода, соединяющего (i,j) с $(i',j') \in V(i,j)$, будет пара чисел $d_1 = (i'-i,j'-j)$, номером входа будет пара чисел $d_2 = -(i'-i,j'-j)$, d_1, $d_2 \in D$.

Для моделирования вызова подкреплений по каналу связи определены графы $\Gamma_1(t)$, вершинами которого являются клетки с координатами (i,j), такие, что $f_{ij} = 1$ в момент времени t и $\Gamma_2(t)$, вершинами которого являются клетки с координатами (i,j), такие, что $f_{ij} = 2$ в момент времени t. Ребра же графов представляют каналы связи между агентами, в случае уничтожения агента инцидентное ему ребро удаляется. Эти графы соответствуют системам связи своих войск и противника. Далее, если не принципиально, кому принадлежит система связи, будем и $\Gamma_1(t)$, и $\Gamma_2(t)$ обозначать как $\Gamma(t)$.

Более реалистичное моделирование каналов связи будет рассматриваться далее в главе 3, а здесь мы будем просто полагать, что в каждый момент времени t каждой компоненте связности $\Gamma_i^k(t) = (\mathcal{V}_i^k, \mathcal{E})$ графа $\Gamma_i(t) = (\mathcal{V}_i, \mathcal{E})$, $i = 1, 2$, поставлен в соответствие конечный автомат $\Pi_i^k(t)$ с количеством входов $\{\chi_l\}$ и выходов $\{\nu_l\}$, равным количеству вершин $l = |\mathcal{V}_i^k|$ в данной компоненте. Основные выходы y_2 и входы x_2 всех клеток $(i',j') \in \mathcal{V}_i^k$ подключены к соответствующим выходам и входам автомата $\Pi_i^k(t)$. На любом выходе конечного автомата $\Pi_i^k(t)$ всегда содержится конкатенация всех значений на его входах, т.е. $\nu_l = \chi_1\chi_2 \ldots \chi_l$ (рис. 2.9).

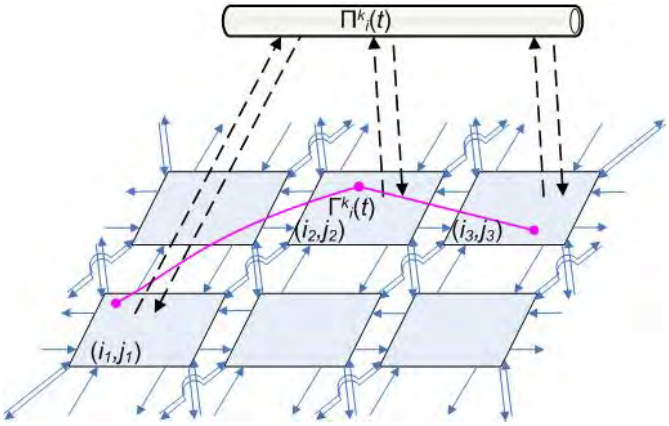


Рисунок 2.9 — Боковые (показаны сплошными стрелочками) и основные (показаны пунктиром) входы и выходы автоматов.

2.3.6 Сущности, необходимые для моделирования конфликта и системы связи

Зададим несколько функций, необходимых для моделирования конфликта. Введем обозначение

$$\theta(f) = \begin{cases} 1,\ f > 0, \\ 0,\ f \leqslant 0. \end{cases}$$

Во первых, определим вероятность уничтожения агента в зависимости от количества враждебных ему агентов в некоторой ближайшей окрестности:

$$\text{DDestroy}(\cdot; V^0) : \mathbb{Z}^2 \to [0,1],$$
$$\text{DDestroy}(i,j; V_0) = 1 - \left(\sum_{\substack{(i',j') \in V_0(i,j) \\ f_{i'j'} \neq f_{ij} \wedge f_{i'j'} \neq 0}} \theta(f_{i'j'}) + 1 \right)^{-1}.$$

Здесь $V^0 = (\alpha_1^0, \ldots, \alpha_p^0)$, $\alpha_i^0 \in \mathbb{Z}^2$, $i = \overline{1,p}$, – шаблон соседства, определяющий окрестность $V^0(i,j) = \{(i,j) + \alpha^0 | \alpha^0 \in V^0\}$, учитываемую для определения вероятности уничтожения. Можно предположить, например, что $V^0(i,j) = V_1(i,j)$.

Далее, введем функцию определения направления из D, ближайшего к направлению от $z_1 \in \mathbb{Z}^2$ до $z_2 \in \mathbb{Z}^2$ по прямой линии:

$$\text{DDirection} : \mathbb{Z}^2 \times \mathbb{Z}^2 \to D,$$
$$\text{DDirection}(z_1,z_2) = d \in D,\ d = \arg\min_{d' \in D} \left\| \frac{z_1 - z_2}{\|z_1 - z_2\|_\infty} - d' \right\|_\infty.$$

Также определим функции обнаружения противников и союзников в ближайшей окрестности. Функция, возвращающая список «своих», определяется как

$$\text{DSelf}(\cdot; V^1) : \mathbb{Z}^2 \to 2^{\mathbb{Z}^2},$$
$$\text{DSelf}(i,j; V^1) = \{(i',j') | f_{i'j'} = f_{i,j} \wedge (i',j') \in V^1(i,j)\}.$$

Здесь $V^1 = (\alpha_1^1, \ldots, \alpha_p^1)$, $\alpha_i^1 \in \mathbb{Z}^2$, $i = \overline{1,p}$, – шаблон соседства, определяющий окрестность $V^1(i,j) = \{(i,j) + \alpha^1 | \alpha^1 \in V^1\}$, например $V^1(i,j) = V_3(i,j)$.

Функция, возвращающая список противников, определяется следующим образом:

$$\mathrm{DEnemy}(\cdot; V^2) : \mathbb{Z}^2 \to 2^{\mathbb{Z}^2},$$
$$\mathrm{DEnemy}(i,j; V^2) = \{(i',j')|f_{i'j'} \neq f_{i,j} \wedge f_{i'j'} \neq 0 \wedge (i',j') \in V^2(i,j)\}.$$

Здесь $V^2 = (\alpha_1^2, \ldots, \alpha_p^2)$, $\alpha_i^2 \in \mathbb{Z}^2$, $i = \overline{1,p}$, – шаблон соседства, определяющий окрестность $V^2(i,j) = \{(i,j) + \alpha^2 | \alpha^2 \in V^2\}$. Окрестность $V^2(i,j)$ возможно задать так, чтобы зона обнаружения противников зависела от направления движения, например, чтобы из точки $(i,j) \in \mathbb{Z}^2$ были видны противники лишь в полукруге (в метрике Чебышева) радиуса r, отграниченном радиусами, которые перпендикулярны направлению движения (k,l):

$$HV(i,j; k,l) = \{(i',j')|i - |l|r + |k|(1 - |l|)\min(kr,0) \leqslant i' \leqslant i + |l|r+$$
$$+ |k|(1 - |l|)\max(kr,0), j - |k|(1 - |l|)r + |l|\min(-k(i' - i), lr) \leqslant j' \leqslant$$
$$j + |k|(1 - |l|)r + |l|\max(k(i' - i), lr)\}.$$

Зададим функцию, возвращающую список доступных в системе связи агентов, которые сигнализируют сигналом sig:

$$\mathrm{DSignal}(\cdot; sig; \Gamma) : \mathbb{Z}^2 \to 2^{\mathbb{Z}^2},$$
$$\mathrm{DSignal}(i,j; sig; \Gamma) = \{(i',j')|s_{i'j'} = sig, \text{ существует путь в графе } \Gamma \text{ из } (i,j) \text{ в } (i',j')\}.$$

Наконец, сконструируем функцию, определяющую поведение агента с координатами (i,j):

$$\mathrm{Decision}(\cdot; V^1, V^2, \Gamma) : \mathbb{Z}^2 \to Q \times D \times S,$$
$$\mathrm{Decision}(\cdot; V^1, V^2, \Gamma) : (i,j) \mapsto (id, dir, sig).$$

Эта функция проще всего записывается в форме алгоритма.

Шаг 1. Получаем список $asel = \mathrm{DSelf}(i,j; V^1)$ дружественных агентов в окрестности $V^1(i,j)$.

Шаг 2. Получаем список $aen = \mathrm{DSelf}(i,j; V^1)$ враждебных агентов в окрестности $V^2(i,j)$.

Шаг 3. Если длина $asel$ меньше длины aen не менее чем в 2 раза, то переходим к шагу 4, иначе – к шагу 5.

Шаг 4 . 1. Полагаем $id = q_{i',j'}$, где (i',j') – координаты ближайшего к (i,j) враждебного агента из aen. Устанавливаем $sig = 2$ (сигнал отступления).

Ш а г 4 . 2. Если в графе $\Gamma(t)$ нет вершин, до которых существует путь из (i,j) (местоположение других дружественных агентов неизвестно), то переходим к шагу 4.3, иначе – к шагу 4.4.

Ш а г 4 . 3. Вычисляем центр масс многоугольника, вершинами которого являются координаты враждебных агентов из aen. Полагаем (i',j') целыми частями координат центра масс. Устанавливаем $dir = -\mathrm{DDirection}(i,j;i',j')$, таким образом, наш агент будет отступать в сторону, противоположную той, где находится наибольшее скопление враждебных агентов. Переходим к шагу 8.

Ш а г 4 . 4. Вычисляем целую часть (i',j') координат центра масс многоугольника, вершинами которого являются координаты дружественных агентов, таких, что существует путь к этим координатам в $\Gamma(t)$ из (i,j). Устанавливаем $dir = \mathrm{DDirection}(i,j;i',j')$, таким образом, наш агент будет отступать в сторону, в которой находится наибольшее скопление дружественных агентов. Переходим к шагу 8.

Ш а г 5. Если враждебных агентов в $V^2(i,j)$ нет, то переходим к шагу 6.1, иначе – к шагу 7.1.

Ш а г 6 . 1. Если $s_{ij} = 3$ (т. е. наш агент уже идет на помощь), переходим к шагу 6.4, иначе – к шагу 6.2.

Ш а г 6 . 2. Строим список $asig = \mathrm{DSignal}(i,j;2;\Gamma)$. Если он пуст, полагаем $(i',j') = \tilde{c}_{ij}$, $id = \{0,0\}$, $sig = 0$, $dir = \mathrm{DDirection}(i,j;i',j')$, т.е. идем в предопределенную точку назначения и переходим к шагу 8, иначе – к шагу 6.3.

Ш а г 6 . 3. Полагаем, что $sig = 3$, (i',j') – координаты ближайшего к (i,j) агента из $asig$, $id = q_{i'j'}$, $dir = \mathrm{DDirection}(i,j;i',j')$.

Ш а г 6 . 4. Проверяем, существует ли в Γ вершина (i',j'), такая, что $\tilde{q}_{ij} = q_{i'j'}$, т.е. доступен ли в системе связи тот агент, к которому наш агент идет на помощь. Если да, переходим к шагу 6.5, иначе – к шагу 5.

Ш а г 6 . 5. Если расстояние до (i',j') равно 1, то переходим к шагу 6.7, иначе – к шагу 6.6.

Ш а г 6 . 6. Устанавливаем $id = q_{ij}$, $sig = s_{ij}$, $dir = \mathrm{DDirection}(i,j;i',j')$.

Ш а г 6 . 7. Переходим к шагу 5.

Ш а г 7 . 1. Полагаем (i',j') координатами ближайшего к (i,j) враждебного агента из aen. Устанавливаем $sig = 1$, $id = q_{i'j'}$. Если расстояние между (i,j) и (i',j') больше единицы, переходим к шагу 7.2, иначе – к шагу 7.3.

Ш а г 7 . 2. Устанавливаем, что $dir = \mathrm{DDirection}(i,j;i',j')$. Переходим к шагу 8.

Ш а г 7 . 3. Устанавливаем, что $dir = (0,0)$. Переходим к шагу 8.

Ш а г 8. Возвращаем (id, dir, sig). Конец функции.

2.3.7 Определение весов, необходимых для моделирования движения

Зададим правила вычисления вероятностей перемещения из данной клетки $(i,j) \in \mathbb{Z}^2$, $f_{ij} > 0$ в соседние. Сначала запретим перемещаться в направлении $d = (k,l) \in D$ уже занятых и совершенно непроходимых клеток:

$$P_1(i,j;k,l) = \begin{cases} \frac{1}{8}(1 - \theta(f_{i+k,j+l})),\ u_{ij} \geqslant 0, \\ 0,\ u_{ij} < 0, \end{cases}$$

и покидать клетки до истечения установленного в u_{ij} времени прохождения:

$$P_2(i,j;k,l) = \begin{cases} 0,\ u_{ij} - w_{ij} > 0, k \neq 0 \vee l \neq 0, \\ 1,\ u_{ij} - w_{ij} \leqslant 0. \end{cases}$$

Затем запретим членам одной группы расходиться слишком далеко друг от друга. Обозначим множество координат агентов, чьи координаты в настоящий момент времени принадлежат к одной компоненте связности графа $\Gamma_{f_{ij}}(t)$ как $V_\Gamma(i,j;t)$. Тогда вес расхождения агентов из одной группы можно определить как

$$P_3(i,j;k,l;t) = \psi_1\Big(\max_{\substack{(h,m)\in V_\Gamma(i,j;t) \\ q^1_{h,m}=q^1_{i,j}}} l_\infty((h,m),(i+k,j+l))\Big). \tag{2.23}$$

Также запретим группам агентов располагаться слишком далеко друг от друга. Определим координаты c_p агента второго порядка e_{p0} как целую часть координат центра масс многоугольников $\{(i_1,j_1),\ldots,(i_n,j_n)|q^1_{i_1,j_1} = \cdots = q^1_{i_n,j_n} = p\}$, в вершинах которого находятся агенты первого порядка, подчиненные данному агенту второго порядка. Поэтому введем функции:

$$I(i;p) = \left[\frac{1}{n}\sum_{\substack{k=1,i_k\neq i \\ q^1_{i_k,j_k}=p}}^{n} i_k + \frac{i}{n}\right], \quad J(j;p) = \left[\frac{1}{n}\sum_{\substack{k=1,j_k\neq j \\ q^1_{i_k,j_k}=p}}^{n} j_k + \frac{j}{n}\right].$$

Очевидно, что $(I(0;p), J(0;p)) = c_p$. Аналогично (2.23) можно установить

$$P_4(i,j;k,l) = \psi_2\Big(\max_{p=\overline{1,n}} l_\infty((I(0;p),J(0;p)),(I(i+k;q^1_{ij}),J(j+l;q^1_{ij})))\Big). \tag{2.24}$$

Здесь $\psi_1, \psi_2 : \mathbb{R} \to [0,1]$ – произвольные убывающие при $x \to +\infty$ функции.

Назовем маршрутом агента последовательность

$$M = \{(i_1,j_1),(i_2,j_2),\ldots,(i_s,j_s)|(i_k,j_k) \in K, k = \overline{1,s}, (i_{k+1}-i_k,j_{k+1}-j_k) \in D, k = \overline{1,s-1}$$

такую, что агент, находящийся в квадрате ω_{i_1,j_1}, будет последовательно каждый ход перемещаться в квадраты $\omega_{i_2,j_2},\ldots,\omega_{i_s,j_s}$. Введем обозначение

$$\psi_3(u,T_{\max}) = \begin{cases} u,\ u \geqslant 0, \\ T_{\max},\ u < 0. \end{cases}$$

Весом маршрута M назовем величину

$$\Lambda(M;\alpha,\beta,T_{\max}) = \alpha \sum_{(i,j)\in M} \psi_3(u_{ij},T_{\max}) + \beta \sum_{(i,j)\in M} \theta(f_{ij}).$$

Первое слагаемое отвечает за время, необходимое для прохождения клеток маршрута, а второе – за количество других агентов, которых, возможно, придется обходить на квадратах маршрута. Соответственно параметры модели α и β отвечают за важность каждого из слагаемых при построении маршрута, а $T_{\max}$ – величина штрафа за то, что маршрут содержит непроходимые клетки, которые обязательно придется обойти (например, возможно взять $\alpha = \beta = 1$ и $T_{\max} = 10$).

Предполагается, что агенты обоих противоборствующих сторон стремятся двигаться в некоторые подмножества $K'_f \subset K$, $K'_f = \{c_{ij}|f_{ij} = f\}$, $f = 1,2$, маршрутом с наименьшим весом. Идея поиска такого маршрута агентом, находящимся в квадрате ω_{ij}, заключается в том, чтоб вместо того, чтобы искать его по всей области K, ищется маршрут с наименьшим весом до ближайшей к c_{ij} стороны или вершины границы окрестности агента $V_o(i,j)$ (напомним, что окрестности в рассматриваемой модели являются квадратами), где o – параметр, отвечающий за дальность просмотра агента. В предыдущем разделе была описана функция $\mathrm{Decision}(i,j;V^1,V^2,\Gamma)$, которая возвращает именно направление $dir \in D$ к точке c_{ij}, если агент не отвлекается на что-нибудь. Отметим, что окрестность просматривается заново каждый ход, так как предполагается в дальнейшем ввести динамически возникающие препятствия, а также возможно появление других агентов на пути данного. Запишем направление, которого должен придерживаться агент на данном ходу для реализации своих целей, через $(k',l') = dir$.

Обозначим через $\mathscr{M}$ множество всех возможных маршрутов в K, а через $\mathscr{M}_o(k,l;k',l')$
$\subset \mathscr{M} \cap V_o(i,j)$ – множество всех маршрутов в $V_o(i,j)$, начинающихся в $(i_1,j_1) = (i+k,j+l)$, $(k,l) \in D$, и заканчивающихся на той стороне или в той вершине $L_o(i,j;k',l') \subset V_o(i,j)$, которая пересекается лучом с направляющим вектором (k',l'), исходящим из точки (i,j):

$$\mathscr{M}_o(k,l;k',l') = \{\{(i_1,j_1),(i_2,j_2),\ldots,(i_s,j_s)\} = M \in \mathscr{M} | \\ (i_1,j_1) = (i+k,j+l), (k,l) \in D, (i_s,j_s) \in L_o(i,j;k',l'), M \subseteq V_o(i,j)\}.$$

Далее, необходимо отобрать из $\mathscr{M}(k,l;k',l')$ самые легковесные маршруты. Множество таких маршрутов запишем как

$$\mathscr{M}'_o(k,l;k',l') = \{M' \in \mathscr{M}_o(k,l;k',l') | \Lambda(M';\alpha,\beta,T_{\max}) = \\ = \min_{M \in \mathscr{M}(k,l;k',l')} \Lambda(M;\alpha,\beta,T_{\max})\}. \quad (2.25)$$

При реализации клеточного автомата в программе построение вышеуказанных множеств может быть слишком долгим. Поэтому для практической реализации возможно определить $\mathscr{M}(k,l;L_o(i,j))$, $\mathscr{M}'(k,l;L_o(i,j))$, $o \geqslant 3$, несколько иначе, а именно рассматривать только маршруты длиной не более трех клеток без самопересечений:

$$\mathscr{M}_o(k,l;k',l') = \{\{(i_1,j_1),(i_2,j_2),(i_3,j_3)\} = M \in \mathscr{M} | \\ (i_1,j_1) = (i+k,j+l), (k,l) \in D, (i_s,j_s) \neq (i_p,j_p), s,p = \overline{1,3}, M \subset V_o(i,j)\} \cup \\ \{\{(i_1,j_1),(i_2,j_2)\} = M \in \mathscr{M} | (i_1,j_1) = (i+k,j+l), (k,l) \in D, (i_1,j_1) \neq (i_2,j_2), \\ M \subset V_o(i,j)\} \cup \{\{(i_1,j_1)\} = \\ = M \in \mathscr{M} | (i_1,j_1) = (i+k,j+l), (k,l) \in D, M \subset V_o(i,j)\}.$$

Далее, обозначим как $\gamma_o(i,j;k',l';r,p)$ вес маршрута, соединяющего квадрат ω_{rp} с границей окрестности $L_o(i,j)$ по лучу, исходящего из (r,p) с направляющим вектором (k',l'), и определим

$$\mathscr{M}'_o(k,l;k',l') = \{\{\ldots,(i'_3,j'_3)\} = M' \in \mathscr{M}(k,l;k',l') | \Lambda(M';\alpha,\beta,T_{\max}) + \\ + \gamma_o(i,j;k,l;i'_3,j'_3) = \min_{M \in \mathscr{M}(k,l;k',l')} (\Lambda(M;\alpha,\beta,T_{\max}) + \gamma_o(i,j;k',l';i_3,j_3))\}. \quad (2.26)$$

Если, например, все элементы движутся слева направо и ищут при этом маршрут с наименьшим весом к правой границе области, то $L_o(i,j;k',l')$ – это всегда

правая сторона квадрата $V_o(i,j)$ и

$$\gamma_o(i,j;0,1;r,p) = \Lambda(\{(r,p+1),(r,p+2),\ldots,(r,o-p+j)\};\alpha,\beta,T_{\max}).$$

Из (2.25), (2.26) следует, что все маршруты из $\mathscr{M}'_o(k,l;k',l')$ имеют одинаковый вес, который будем обозначать как $L(k,l;k',l')$. Очевидно, что необходимо уменьшать вероятность перемещения в направлении $(k,l) \in D$ при увеличении $L(k,l;k',l')$.

Увеличим вероятность перемещения агента в направлении маршрута с наименьшим весом, задав величину

$$P_5(i,j;k,l;k',l';\alpha,\beta,T_{\max};o) = \begin{cases} \psi_4(L(k,l;k',l')), (k,l) \neq (0,0), \\ 0, (k,l) = (0,0) \wedge (k',l') \neq (0,0), \\ 1, (k,l) = (0,0) \wedge (k',l') = (0,0). \end{cases} \tag{2.27}$$

Здесь $\psi_4 : \mathbb{R} \to [0,1]$ – произвольная убывающая при $x \to +\infty$ функция.

Введем

$$P'(i,j;k,l;k',l';\alpha,\beta,T_{\max};o;r) = \\ = P_1(i,j;k,l)P_2(i,j;k,l)P_3(i,j;k,l;r)P_4(i,j;k,l)P_5(i,j;k,l;k',l';\alpha,\beta,T_{\max};o).$$

Определим итоговую вероятность перемещения в направлении $(k,l) \in D$ из ячейки (i,j) при необходимости придерживаться направления (k',l'). Если

$$\sum_{(k,l)\in D} P'(i,j;k,l;\alpha,\beta,T_{\max};o;r) \neq 0,$$

то

$$P_{ij}(k,l;k',l';\alpha,\beta,T_{\max};o;r) = \frac{P'(i,j;k,l;k',l';\alpha,\beta,T_{\max};o;r)}{\sum\limits_{(k,l)\in\mathcal{D}} P'(i,j;k,l;k',l';\alpha,\beta,T_{\max};o;r)},$$

иначе

$$P_{ij}(k,l;k',l';\alpha,\beta,T_{\max};o;r) = \begin{cases} 1, (k,l) = (0,0), \\ 0, (k,l) \neq (0,0). \end{cases}$$

Наконец, определим функцию $\psi_5 : \mathbb{Z}^2 \to \mathcal{D}$ следующим образом. Пусть $(i,j) \in \mathbb{Z}^2$. Выберем случайно один вектор $(i',j') \in V_1(i,j)$, такой, что $f_{i'j'} \neq 0$, $(i',j') + d_{i'j'} = (i,j)$. Положим $\psi_5(i,j) = d_{i'j'}$. Если вышеупомянутого вектора не нашлось, положим $\psi_5(i,j) = (0,0)$. Данная функция впоследствии будет использоваться для разрешения ситуации нескольких претендентов на одну клетку.

2.3.8 Описание функционирования автомата

Определим отношение строгого квазипорядка в $\mathcal{D}$ простейшим возможным в данном случае способом (другой вариант рассмотрен в предыдущей работе [2]).

Положим в (2.23), (2.24), (2.27) $\varphi_1(x) = \varphi_2(x) = \psi(x) = 1/x$. Будем говорить, что $(k_1,l_1) \prec (k_2,l_2)$ в клетке (i,j), если $P_{ij}(k_1,l_1;\alpha,\beta,T_{\max};o;r) < P_{ij}(k_2,l_2;\alpha,\beta,T_{\max};o;r)$. В случае, если для некоторого множества $\mathcal{D}' \subset \mathcal{D}$ верно, что

$$\forall((k_1,l_1),(k_2,l_2) \in \mathcal{D}')\ P_{ij}(k_1,l_1;\alpha,\beta,T_{\max};o;r) = P_{ij}(k_2,l_2;\alpha,\beta,T_{\max};o;r),$$

то отношение квазипорядка для элементов D' определяется следующим образом. Пусть $\xi_1,\ldots,\xi_m$ – случайная перестановка чисел $\overline{1,m}$, $m = |\mathcal{D}'|$. Тогда $(k_i,l_i) \prec (k_s,l_s)$, если $\xi_i < \xi_s$.

Опишем теперь глобальное функционирование клеточного автомата с помощью алгоритма (формальное описание в соответствии с определением ОСВВ довольно сложно и может сильно затруднить понимание происходящего, поэтому мы его сделаем в следующем разделе).

Ш а г 1. Для всех $i,j = \overline{1,L_K}$ устанавливаем $(\zeta_{ij},\eta_{ij}) = \psi_5(i,j)$.

Ш а г 2. Для всех $i,j = \overline{1,L_K}$ проверяем значение f_{ij}. Если $f_{ij} \neq 0$, то переходим к шагу 3, иначе – к шагу 11.

Ш а г 3. Для всех $i,j = \overline{1,L_K}$ вычисляем значение случайной величины $\xi_{ij} \in \{0,1\}$, $\mathbb{P}\{\xi = 0\} = \mathrm{DDestroy}(i,j;V^0)$. Если $\xi = 0$, то агент в (i,j) уничтожен, переходим к шагу 11, иначе – к шагу 4.

Ш а г 4. Проверяем счетчик ходов, нужных для преодоления квадрата ω_{ij}. Если $u_{ij} = w_{ij}$, переходим к шагу 5, иначе – к шагу 10.

Ш а г 5. Вычисляем $\mathrm{Decision}(i,j;V^1,V^2,\Gamma_{f_{ij}}) = (id, dir, sig)$. Устанавливаем $\tilde{q}_{ij} = id$, $d_{ij} = dir$, $s_{ij} = sig$.

Ш а г 6. Полагаем $(k',l') = d_{ij}$. Получаем список

$$\mathcal{D}'' = \{(k,l) \in \mathcal{D} | P_{ij}(k,l;k',l';\alpha,\beta,T_{\max};o;r) \neq 0\}$$

и упорядочиваем его по убыванию по отношению $\prec$.

Ш а г 7. Выбираем последовательно элементы $(k,l) \in \mathcal{D}''$ от первого до последнего до тех пор, пока не станет $(\zeta_{i+k,j+l},\eta_{i+k,j+l}) = (k,l)$, после чего переходи к шагу 8. Если такого элемента не нашлось, то полагаем $(k,l) = (0,0)$ и переходим к шагу 9.

Ш а г 8. Устанавливаем $f_{ij} = 0$, $q_{ij} = 0$, $s_{ij} = 0$, d_{ij}=0, $c_{ij} = 0$, $\tilde{q}_{ij} = 0$, $w_{ij} = 0$.

Ш а г 9. Устанавливаем $f_{i+k,j+l} = f_{ij}$, где (k,l) – найденный на шаге 7 элемент. Переходим к шагу 11.

Ш а г 1 0. Увеличиваем счетчик ходов w_{ij} на единицу. Переходим к шагу 11.

Ш а г 1 1. Выбираем следующую пару (i,j). Переходим к шагу 1.

2.3.9 Определение функций локального тактового функционирования ОСВВ.

Описание полученного клеточного автомата будет неполным, если мы не приведем формального описания функций локального тактового функционирования ОСВВ в духе определения.

Пусть $\mathfrak{s} = (f, q, s; d, c, \tilde{c}, \tilde{q}; w, u; \zeta, \eta)$. Определим локальную выходную функцию φ_1:

$$\varphi_1^i(x, b, \mathfrak{s}) = \begin{cases} \varepsilon, f = 0 \vee x^i \neq s, \\ \mathfrak{s}, f \neq 0 \wedge x^i = s; \end{cases}$$

$d-$ю компоненту функции взаимодействия $\varphi_2 = (\varphi_2^{(-1,-1)}, \ldots, \varphi_2^{(1,1)}, \ldots)$, $d \in \mathcal{D} \setminus \{(0,0)\}$ запишем так

$$\varphi_2^d(\mathfrak{s}) = \begin{cases} \varepsilon, f \neq 0 \wedge w \neq u \vee f = 0 \wedge d \neq \eta, \\ \mathfrak{s}, f \neq 0 \wedge w = u \vee f = 0 \wedge d = \eta. \end{cases}$$

Зададим локальную функцию переходов φ. Пусть значение на $d'-$м боковом входе $b_{d'} = (f_{d'}, q_{d'}, s_{d'}; d_{d'}, c_{d'}, \tilde{c}_{d'}, \tilde{q}_{d'}; w_{d'}, u_{d'}; \zeta_{d'}, \eta_{d'})$, $d' \in \mathcal{D} \setminus \{(0,0)\}$. Обозначим $d''(b) = (\zeta,\eta)$ максимальный по отношению $\prec$ элемент множества $\mathcal{D}'' = \{d' \in \mathcal{D} \setminus \{(0,0)\} | d' = (\zeta_{d'},\eta_{d'})\}$ (т.е. определим множество соседних пустых ячеек, допускающих перемещение в них, это формальный аналог

шага 6 алгоритма из предыдущего раздела). Строго говоря, функция ψ_5 из предыдущего параграфа зависит от бокового входа b, из которого она узнает состояния соседних автоматов и выбирает случайным образом номер бокового входа $(\zeta,\eta) = d' \in \mathcal{D} \setminus \{(0,0)\}$, такой, что $d_{d'} = d'$.

Напомним, что в разд. 2.3.6 была введена функция

$$(id, dir, sig) = \text{Decision}(i,j; V^1, V^2, \Gamma).$$

При внимательном рассмотрении видно, что $id = id(x_2,b)$, $dir = dir(x_2,b)$ и $sig = sig(x_2,b)$. Координаты точки, в которую движется агент, могут быть изменены через один из основных входов вышестоящей инстанцией, таким образом, $\tilde{c} = \tilde{c}(x_1)$.

$$\varphi(x, b, \mathfrak{s}) = \begin{cases} (f_d, q_d, sig(x_2,b); dir(x_2,b), c_d, \tilde{c}(x_1), id(x_2,b); 0, u_d; 0, 0), f = 0, (\zeta,\eta) = d, \\ (f, q, s; d, c, \tilde{c}(x_1), \tilde{q}; w+1, u; 0, 0), f \neq 0, w < u, \\ (f, q, s; d, c, \tilde{c}, \tilde{q}; w, u; \zeta, \eta), f = 0, b_d = \varepsilon, d \in \mathcal{D} \setminus \{(0,0)\}, \\ (f, q, s; d, c, \tilde{c}(x_1), \tilde{q}; w, u; 0, 0), f \neq 0, w = u, b_d = \varepsilon, d \in \mathcal{D} \setminus \{(0,0)\}, \\ (0, (0,0), 0; (0,0), c, (0,0), (0,0); 0, u; \psi_5(b)), f \neq 0, b^d \neq \varepsilon, d \in \mathcal{D} \setminus \{(0,0)\}. \end{cases}$$

2.4 Моделирование рефлексии агента

2.4.1 Общее описание клеточного автомата, моделирующего агентов с памятью

Пусть, как и в предыдущем разделе, $Ag = \{ag_1, \ldots, ag_k\}$ – система агентов, передвигающихся по ландшафту $\mathcal{L}_l(n \times m)$, и начальные и конечные ячейки ландшафта указываны для каждого агента. Идея настоящего раздела состоит в том, что в модели в дополнение к общему для всех агентов «слою», соответствующему объективной реальности, каждый агент имел бы «слой», соответствующий информации о реальности, которая известна этому агенту.

Поведение агентов системы моделируется клеточным автоматом со множеством клеток $World = \{(i,j,id) | i,j \in \mathbb{Z}, id = \overline{0,k}\} \subset \mathbb{Z}^3$. Во множестве

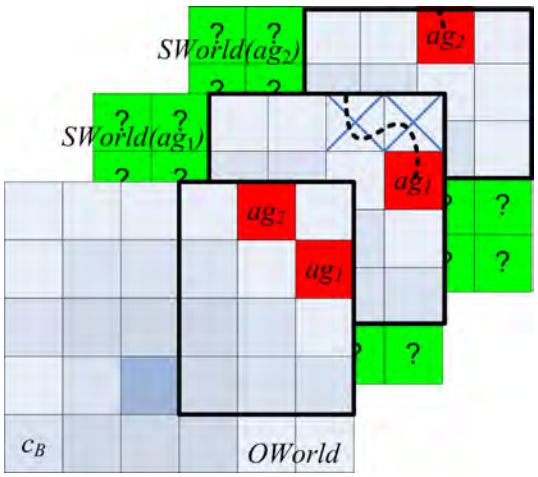


Рисунок 2.10 — Пример клеточного автомата для агентов с памятью

$World$ можно выделить $k+1$ клеточных плоскостей: слой объективной реальности $OWorld = \{(i,j,0)|i,j \in \mathbb{Z}\}$ и слои субъективной реальности агентов с идентификаторами $ag = \overline{1,k}$ $SWorld_{ag} = \{(i,j,ag)|i,j \in \mathbb{Z}\}$. Таким образом,

$$World = OWorld \cup \left(\bigcup_{ag=1}^{k} SWorld_{ag} \right).$$

Предположим, что прямоугольник $K \subset World$, $K = \{(i,j,id)|i = \overline{0,L_K}, j = \overline{0,L_K}, id = \ overline0,k\}$, и все ячейки находятся в состоянии покоя вне его.

Пусть слой объективной реальности $OWorld$ состоит из клеток $(i,j) \in \mathbb{Z}^2$ с различной непроходимостью u_{ij}. Значение u_{ij} – это число тактов времени, которое требуется для пересечения квадрата ω_{ij} с координатами (i,j). Если ω_{ij} полностью непроходим, тогда положим $u_{ij} = -1$. Кроме того, клетки могут содержать информацию о агенте в ячейке, квадрате назначения агента и т. д.

Слой субъективной реальности состоит из клеток (i,j), так что каждая его клетка (i,j) соответствует клетке (i,j) уровня объективной реальности. Клетки слоя субъективной реальности ag содержат информацию о текущей позиции агента ag, об истории движения ag и о непроходимости известных агентам ag клеток.

Итак, пусть заданы

1. Замощение Ω_h области $\Omega \subset \mathbb{R}^2$ (для простоты можно считать, что это правильный квадратный паркет $\Omega_h = \{\omega_{ij}|i,j \in Q_h \subset \mathbb{Z}\}$ и не различать клетку ω_{ij} и ее координаты (i,j)).

2. Взаимно-однозначная функция получения уникального идентификатора (УИд) агента $\mathrm{uid} : Ag \to \mathbb{N}$.
3. Дискретная функция непроходимости $\mathrm{u} : \mathbb{Z}_{\geqslant 0} \times \Omega_h \to \mathbb{Z}$, устанавливающая, сколько тактов функционирования КА необходимо для преодоления клетки (i,j). При этом предполагается, что для совершенно непроходимой в момент времени t клетки (i,j) $\mathrm{u}(t,(i,j)) = +\infty$. Эта функция получается, по своей сути, дискретизацией функции проходимости, например:

$$u_{ij}(t) = \mathrm{u}(t,(i,j)) = \left\lfloor \frac{\max_{r\in\Omega} \mathrm{u}^c(\tau t,r)}{\min_{r\in\omega_{ij}} \mathrm{u}^c(\tau t,r)} \right\rfloor,$$

где $\lfloor x \rfloor$ означает целую часть x, а τ – продолжительность одного такта. Пусть $r_h \subseteq \Omega_h$ – маршрут (на языке перечислительной комбинаторики – решеточный путь), начальная клетка маршрута $r_h[1] = (i,j) + d_1$, $d_1 \in \mathcal{D}$, k-я клетка маршрута $r_h[k] = r_h[k-1] + d_k$, $d_k \in \mathcal{D}$, $k > 1$. Обозначим $k-1$ -ю клетку маршрута r_h с координатами (i',j'), в которой агент находится в момент времени t, как $r_h(t)$. Тогда дискретным аналогом уравнения (2.18), определяющего движение агента, будет уравнение

$$\|r_h(t+k) - r_h(t)\| = \|d_k\| \left\lfloor \frac{k}{\|d_k\| u_{i'j'}(t)} \right\rfloor, \quad k \leqslant \|d_k\| u_{i'j'}(t). \tag{2.28}$$

Детальное формальное определение функций локального такового функционирования описываемого КА (без задания весов, необходимых для поддержания строя) приведено в работе [1]. Вкратце, агент на каждом такте выбирает случайный маршрут движения, удовлетворяющий уравнению (2.28) и отвечающий ряду других ограничений (например, два агента не могут одновременно находиться в одной клетке). Распределение случайного выбора маршрута таково, что вероятность выбрать минимизирующий нижеприведенные функционалы (2.29), (2.31) или (2.32) маршрут максимальна.

В связи с особенностями реализации, КА должен быть представим как в виде *robot-space cellular automaton*, так и в виде *world-space cellular automaton*, так как первый тип автомата удобен для построения строя, а второй – для работы алгоритма поиска оптимального маршрута и для построения тестовых ландшафтов. Поэтому агент (как ячейка *robot-space cellular automaton*) $ag \in Ag$ в момент времени $t \in \mathbb{Z}_{\geqslant 0}$ представляет собой объект вида

$$ag = \{selfId, leadId, lead, w, \mathcal{D}', trgtAct, trgtTmp, formTmpl, @cell\},$$

где $selfId = \mathrm{uid}(ag)$ – уникальный идентификатор агента, $leadId \in \mathbb{N}$ – уникальный идентификатор лидера данного агента, $lead \in \{true, false\}$ – указывает на то, что агент является ведущим, w – количество тактов, которое агент уже простоял в текущей клетке, $\mathcal{D}'$ – упорядоченный по убыванию желательности список возможных направлений из $\mathcal{D}$ для агента на следующий такт, $trgtAct \in \Omega_h$ – текущая целевая клетка агента, $trgtTmp \in \Omega_h$ – временная целевая клетка агента, $formTmpl = \Phi_h(ag)$ – шаблон строя, $@cell$ – указатель на ячейку, в которой находится агент ag в момент времени t. Ячейка $cell$, соответствующая клетке $(i,j) \in \Omega_h$ для *world-space cellular automaton* представляет собой

$$cell = \{u, coo, @ag\},$$

где $u = \mathrm{u}(t,coo)$ – значение функции непроходимости клетки (т.е. минимальное количество тактов, нужное для преодоления клетки), $coo = (i,j)$, $@ag$ – указатель на агента, находящегося в клетке (i,j) в момент времени t или нуль, если в (i,j) в момент времени t нет агента. Будем обозначать агента, на которого указывает $@ag$ как ag_{ij} и писать

$$ag_{ij} = \{selfId_{ij}, leadId_{ij}, w_{ij}, \mathcal{D}_{ij}, trgtAct_{ij}, trgtTmp_{ij}, formTmpl_{ij}, @cell_{ij}\}.$$

Если $@ag_{ij} = 0$, то для единообразия будем считать, что

$$ag_{ij} = \{0, 0, 0, 0, 0, 0, 0, 0\}.$$

Также будем обозначать ячейку, на которую ссылается $@cell_k$, как $cell_k$ и писать $cell_k = (u_k, coo_k, @ag_k)$. Помимо этого, будем обозначать, что $cell_{ij} = \{u_{ij}, (i,j), @ag_{ij}\}$.

В результате использования ячеек, содержащих указатели, возможно при реализации КА сэкономить объем памяти, нужной для хранения конфигурации КА, так как количество агентов обычно существенно меньше количества клеток и при хранении состояния агента в каждой клетке *world-space cellular automaton* потребовалось бы хранение большого количества нулевых значений. Также возможно увеличить быстродействие КА, так как при перемещении агента в новую клетку не нужно копировать его состояние целиком, а нужно лишь изменить значение пары указателей, и при доступе к ячейке $cell$ уже сразу известен адрес $@ag$ и наоборот. В псевдокоде алгоритмов мы также будем обозначать поля объектов $cell$ и ag принятым в языках программирования способом, т.е. в формате «имя_объекта.имя_поля», опуская числовые индексы.

Будем обозначать как $V_r(i,j)$ окрестность клетки (i,j) радиуса r

$$V_r(i,j) = \{c \in \mathbb{Z}^2 | \|c - (i,j)\| \leqslant r\}.$$

Определим функцию

$$\theta(x) = \begin{cases} 0, x \leqslant 0, \\ 1, x > 0. \end{cases}$$

Введем дискретный аналог функционалов T_k

$$T_h(r_h) = \alpha_0 \sum_{(i,j) \in r_h} \|d_{ij}\| u_{ij} + \alpha_1 \sum_{(i,j) \in r_h} \theta(agId_{ij}), \tag{2.29}$$

где α_0, $\alpha_1 \geqslant 0$ – параметры модели, отвечающие за то, предпочитает ли агент менее непроходимые или же менее населенные другими агентами маршруты, $d_{ij} = d_k$, если $r_h[k] = (i,j)$.

Обозначим множестве всех маршрутов $\mathcal{M}_d(i,j; i_r,j_r)$ агента ag_{ij}, соединяющих (i,j) и (i_r,j_r), причем (i,j) не входит в маршрут и если $(i_r,j_r) = trgtAct_{ij}$, то (i_r,j_r) не входит в маршрут, $d = r_h[1] - (i,j)$.

Общая идея состоит в том, чтоб каждый агент в клетке (i,j) искал оптимальный по времени маршрут в некоторой своей окрестности $V_r(i,j)$, а потом собирал из таких локально оптимальных маршрутов квазиоптимальный маршрут. Поскольку непроходимость области Ω_h может меняться со временем, а также не быть полностью разведанной и проходы могут блокироваться другими агентами, то поиск глобального оптимального маршрута может быть невозможен или лишен смысла. Тактовое функционирование *world-space cellular automaton*, в котором агенты ищут кратчайший маршрут до точки назначения описывается с помощью алгоритма 1, в который входят алгоритмы 3 и 4, приведенные в подразделе 2.5.4.

Следует подчеркнуть, что для агента движение в сторону скорейшего маршрута наиболее вероятно, но не обязательно. Точнее, агент в клетке (i,j) назначает каждому направлению $d \in \mathcal{D}$ вес $weights[d]$,

$$weights[d] = \min_{r_h \in \mathcal{M}_d(i,j;i_r,j_r)} T_h(r_h). \tag{2.30}$$

Далее $\mathcal{D}$ перемешивается в соответствии с весами из $weights$. Процедура взвешенного случайного перемешивания нетривиальна, автор использует предложенный в [108] алгоритм. Таким образом, получается упорядоченный список

направлений. Если движение в первом по данному списку направлении маршрута невозможно (например, путь заблокирован), то агент выбирает следующий по времени прохождения маршрут и т.д. Это требуется для разрешения коллизий между агентами и для симуляции элемента случайности в передвижении агентов.

Алгоритм 1 ((Поиска квазиоптимального по времени пути))**.**

1: **for all** $ag \in Ag$ **do**
2: $weights =$ sortDirections(ag) ▷ Сортировка направлений, алгоритм 3
3: Случайно перемешать $ag.\mathcal{D}'$ в соответствии с весами из $weights$
4: **end for**
5: **for all** $ag' \in Ag$ **do**
6: Actualize(ag') ▷ Актуализация, алгоритм 4
7: **end for**
8: t = t+1

Можно обратить внимание, что в алгоритме разделен поиск наилучшего (в смысле кратчайшего по времени маршрута) направления движения агентов (алгоритм 3) и действительное обновление конфигурации КА. Это сделано для двух целей – для возможности вставки между этими блоками алгоритма, отвечающего за построение строя и для параллельного вычисления наилучших направлений перемещения. При этом поиск маршрута является самым трудным вычислительно моментом работы КА, и занимает, при вычислении алгоритмом Дейкстры, минимум $O((2r+1)^4)$ времени каждый ход для каждого агента, так как, по сути кратчайший путь в окрестности $V_r(i,j)$ – это кратчайший путь на графе с $(2r+1)^2$ вершинами для всех агентов.

Поиск локально оптимального маршрута может повторяться каждый такт времени, как в предыдущих работах автора, альтернативно, агент ag_{ij} может рассчитать, например, алгоритмом Дейкстры или A^* локально оптимальный в $V_r(i,j)$ маршрут, следовать по нему и производить повторный пересчет лишь при достижении (i_r, j_r) или при обнаружении по ходу маршрута непроходимых препятствий или скоплений других агентов. Очевидно, в этом случае список предпочитаемых направлений движения $ag_{ij}.\mathcal{D}'$ заменяется на цепной путь $d_1, d_2, \ldots d_p$, соединяющий (i,j) и (i_r,j_r). Выбор между этими методами зависит от скорости изменения ландшафта. Также возможно производить поиск пути с

учетом знаний агента о местности и предыдущего опыта агента, как это показано в работе автора [21]. В этом случае, помимо слоя разделяемой всеми агентами реальности $Cell = \{cell\}$, для каждого агента задается свой слой «субъективной реальности». В данный слой, который также описывается как множество ячеек, по мере поступления агент сохраняет информацию об уже посещенных клетках, количестве и времени посещений клеток. При поиске локально оптимального пути необходимо, например, увеличивать «веса» переходов в уже посещенные клетки, для чего рассматривать вместо функционала (2.29) функционал

$$\tilde{T}_h(r_h) = T_h(r_h) + \alpha_2 \sum_{(i,j)\in r_h} visit_{ij}, \quad \alpha_2 \geqslant 0, \tag{2.31}$$

где $visit_{ij}$ – количество предыдущих посещений клетки (i,j), аналогично работе автора [21], моделировать конфликт агентов, как в [1] и т.д. Если агенту известно расстояние от coo_n – ближайшей к клетке назначения клетке из множества $r_h \cap V_r(i,j)$ до точки назначения coo_{target}, то для повышения качества алгоритма поиска можно рассматривать вместо (2.31) функционал

$$\hat{T}_h(r_h) = \tilde{T}_h(r_h) + \alpha_3 \|coo_n - coo_{target}\|, \quad \alpha_3 \geqslant 0. \tag{2.32}$$

Агент ag может применять ранее описанный подход для поиска маршрута в неоткрытых этими агентами областях, то есть в составе ячеек ω_{ij} с $visit_{ij}(ag) = 0$. Более стандартные подходы к поиску оптимальных маршрутов, например алгоритм Дейкстры, могут использоваться в областях, состоящих из уже посещенных ячеек. Однако, использование стандартных методов поиска оптимального маршрута ограничено скоростью изменения ландшафта с течением времени. Возможно, что информация о посещенных ячейках устарела (параметр $time_{ij}(ag)$ используется для определения актуальности информации), или даже непроходимость ячеек изменится непосредственно в процессе прохождения маршрута, выбранного как глобально оптимальный.

Таким образом, в зависимости от скорости изменения ландшафта, необходимо искать компромисс между подходом «Reacting», при котором оценивается текущая ситуация в ближайшей окрестности агента и подходом «Planning», при котором ищется глобально оптимальная траектория. Например, бессмысленно устанавливать радиус o окрестности $V_o(i,j)$, в которой ищется локально оптимальный маршрут, больше чем число тактов, в течении которого ландшафт остается неизменным.

Пример описанного клеточного автомата изображен на рис. 2.10. Увеличение непроходимости на указанном рисунке указано более темным тоном, крестики « $\times$» в слое $SWorld$ отмечают уже посещенные ячейки, метки «?» соответствуют ячейкам, статус которых неизвестен.

2.4.2 Граница

Может возникнуть закономерный вопрос – как искать локально (в окрестности $V_o(i,j)$) оптимальные маршруты, если (i,j) достаточно близко (ближе, чем на o клеток) к границе области K и, таким образом, приходится сравнивать веса маршрутов заведомо неравной длины? Эксперименты показали, что во избежание нахождения неоптимальных маршрутов необходимо продолжить K вовне симметрическим образом. В частности, необходимо продолжить матрицу непроходимостей $(u_{ij})_{i=\overline{1,L_K}}^{j=\overline{1,L_K}} = U_K$ следующим образом

$$U = \left(\begin{array}{c|c|c} Ud_K & Uv_K & Ud_K \\ \hline Uh_K & U_K & Uh_K \\ \hline Ud_K & Uv_K & Ud_K \end{array}\right),$$

где $Ud_K = (u_{ij})_{i=\overline{L_K,1}}^{j=\overline{L_K,1}}$, $Uv_K = (u_{ij})_{i=\overline{L_K,1}}^{j=\overline{1,L_K}}$, $Uh_K = (u_{ij})_{i=\overline{1,L_K}}^{j=\overline{L_K,1}}$.

Также, если маршрут содержит клетку назначения, то ее необходимо не учитывать при вычислении веса маршрута, чтобы избежать бесконечных колебаний около этой клетки, которая может иметь существенно худшую проходимость, чем все окружающие ее клетки.

2.4.3 Функция перехода субъективной реальности

Всякий такт $time$, когда устанавливается для клетки $(i,j,0)$, что $f_{ij} > 0$, для клетки (i,j,ag) увеличивается счетчик посещений $vis_{ij}(ag) := vis_{ij}(ag) + 1$.

Также устанавливается

$$su_{i+k,j+l}(ag) = u_{i+k,j+l}, \quad \|(k,l)\|_\infty \leqslant o,$$
$$time_{i+k,j+l}(ag) = time.$$

В неразведанных агентом ag областях, т.е. в состоящих из клеток ω_{ij} с $vis_{ij}(ag) = 0$, для поиска маршрута агентом применяется описанный ранее подход, в областях, состоящих из уже посещенных клеток могут применяться и более стандартные подходы поиска оптимального маршрута, например, алгоритм Дийкстры. Однако, применение стандартных методов поиска оптимального маршрута ограничено скоростью изменения ландшафта со временем – возможно, что сведения о посещенных клетках устареют (для определения новизны сведений используется параметр $time_{ij}(ag)$) или же вообще непроходимость клеток изменится прямо в процессе прохождения маршрута, выбранного как глобально оптимальный.

В слое субъективной реальности для каждого агента могут быть также заданы радиус притяжения r_a и радиус отталкивания r_r. Агент ag_1 в ячейке (i, j, ag_1), который обнаруживает внутри своей окрестности $V_{r_a}(i, j)$ другого агента ag_2, стремится к последнему, пока обнаруженный агент не попадет в окрестность $V_{r_r}(i,j)$. После этого агент ag_1, напротив, попытается уйти от агента ag_2. Таким образом, агенты могут создавать разные формации независимо от шаблона формации.

2.4.4 Автоматы и теория категорий

Отметим, что агенты, движущиеся по решетке, могут соответствовать не только реальным роботам, перемещающимся по пересеченной местности, но и моделировать некоторый процесс неклассических вычислений. Можно рассматривать конфигурации *world-space cellular automaton* $Cell = \{cell\}$ как категорию, в которой объектами являются ячейки, а морфизмами – решеточные пути между ячейками. Аналогично, можно рассмотреть конфигурации *robot-space cellular automaton* Ag как категорию, в которой объектами являются состояния агентов ag_i в момент времени t, а морфизмами – отображения $\gamma_{t_0} : ag_i(t) \mapsto ag_i(t + t_0)$.

Операция @ – порождает пару функторов между этими категориями:

$$\mathcal{F}_1 : ag \mapsto \{u, coo, @ag\},$$
$$\mathcal{F}_2 : cell \mapsto \{selfId, leadId, lead, w, \mathcal{D}', trgtAct, trgtTmp, formTmpl, @cell\}.$$

Далее, в рамках теории категорий всякую клетку $(i,j) \in \Omega_h$ можно интерпретировать как *итератор*. В этом случае $ag \in Ag$ является *коитератором* [109], который порождает поток клеток, через которые он проходит.

2.5 Алгоритмы организации строя агентов

Пусть заданы следующие динамические графы

1. Желаемый граф строя $\Phi = (Ag, E, \mathrm{coo}^c_\Phi)$, $E \subseteq Ag^2$ – множество ребер, $\mathrm{coo}^c_\Phi : [0,T] \times Ag \to \Omega$ – функция, дающая желаемые для сохранения строя координаты агента в Ω.
2. Фактический граф строя $\Gamma = (Ag, E, \mathrm{coo}^c_\Gamma)$, $\mathrm{coo}^c_\Gamma : [0,T] \times Ag \to \Omega$ – функция, дающая реальные координаты агента в Ω.

Также необходимо, чтоб в любой момент времени $t \in [0,T]$ отличие графов Φ и Γ было бы минимально

$$\mathrm{dist}(\Gamma(t), \Phi(t)) \to \min. \tag{2.33}$$

Формальное описание функции несходства графов dist будет дано ниже в разделе 2.5.1. Таким образом, необходимо найти траектории $r_k \in \mathcal{Y}, k = \overline{1,n}$, являющиеся решениями задачи оптимизации с двумя целевыми функциями (2.21), (2.33).

1. Дискретные функции, дающие желаемые для сохранения строя, предусмотренного графом Φ, координаты агента в текущий момент дискретного времени $\mathrm{coo}_\Phi : \mathbb{Z}_{\geqslant 0} \times Ag \to \Omega_h$ и реальные координаты агента $\mathrm{coo}_\Gamma : \mathbb{Z}_{\geqslant 0} \times Ag \to \Omega_h$.
2. Шаблон формации $\Phi_h(ag) = \{(i,j,agId) | (i,j) \in \mathbb{Z}^2, agId \in \mathrm{uid}(Ag) \cup \{0\}\}$, содержащий относительные координаты соседей агента по строю (строгое определение соседства дано в разделе 2.5.1), причем $agId = 0$, если нет требований к тому, какой именно сосед должен находится и

$agId > 0$, если в клетке с координатами $\mathrm{coo}(t, ag) + (i,j)$ в момент времени t должен находиться именно сосед с уникальным идентификатором $agId$. Таким образом, для элемента $(i,j,agId) \in \Phi_h(ag)$, $agId \neq 0$, справедливо соотношение

$$(i,j) + \mathrm{coo}_\Phi(ag) = \mathrm{coo}_\Phi(ag'), \quad \mathrm{uid}(ag') = agId.$$

Определение 13. *Определим расстояние* $\mathrm{gd} : Ag^2 \to \mathbb{R}_{\geqslant 0}$*, следующим образом. Если между агентами* ag_1*,* $ag_2 \in Ag$ *существует в* Φ *кратчайший путь длиной* l*, то* $\mathrm{gd}(ag_1, ag_2) = l$*, иначе* $\mathrm{gd}(ag_1, ag_2) = \infty$*,* $\mathrm{gd}(ag_1, ag_1) = 0$.

Это расстояние принято называть геодезическим расстоянием на графе. Будем считать окрестностью агента $ag \in Ag$ радиуса ρ множество $W_\rho(ag) \subseteq Ag$, такое что $\forall(ag' \in W_r(ag)) \quad \mathrm{gd}(ag, ag') \leqslant \rho$. Соответственно, будем считать соседями агента ag всех агентов $ag' \in W_\rho(ag)$.

Пусть $\Gamma_i = (Ag_i, E_i, \mathrm{coo}_i)$, $Ag_i \subseteq Ag$, $E_i \subseteq Ag^2$, $\mathrm{coo}_{\Gamma_i} : \mathbb{Z}_{\geqslant 0} \times Ag_i \to \Omega_h$ – функция соответствия вершины графа Γ_i и клетки Ω_h, $i = 1{,}2$ – два графа, описывающие строи агентов. Определим расстояние несходства между ними в момент времени t как

$$\mathrm{dist}(\Gamma_1(t), \Gamma_2(t)) = m_0 + m_1 + \kappa \sum_{ag \in Ag_c} \|\mathrm{coo}_{\Gamma_1}(t, ag) - \mathrm{coo}_{\Gamma_2}(t, ag)\|,$$

где $Ag_c \subseteq Ag$ – множество вершин наибольшего общего подграфа Γ_1 и Γ_2, $m_0 = |Ag_0 \backslash Ag_c|$, $m_1 = |Ag_1 \backslash Ag_c|$, $\kappa = (\mathrm{diam}(\Omega_h))^{-1}$ – диаметр множества Ω_h в клетках. Это расстояние в своей сущности является частным случаем ранее введенного в [11] автором расстояния графов.

2.5.1 Метрика сходства графов и метрика, порожденная графом

Определение 14. *Определим расстояние* $\mathrm{gd} : Ag^2 \to \mathbb{R}_{\geqslant 0}$*, следующим образом. Если между агентами* ag_1*,* $ag_2 \in Ag$ *существует в* Φ *кратчайший путь длиной* l*, то* $\mathrm{gd}(ag_1, ag_2) = l$*, иначе* $\mathrm{gd}(ag_1, ag_2) = \infty$*,* $\mathrm{gd}(ag_1, ag_1) = 0$.

Такое расстояние обычно называют геодезическим расстоянием на графе. Будем считать окрестностью агента $ag \in Ag$ радиуса ρ множество $W_\rho(ag) \subseteq$

Ag, такое что $\forall(ag' \in W_r(ag)) \quad \mathrm{gd}(ag, ag') \leqslant \rho$. Соответственно, будем считать соседями агента ag всех агентов $ag' \in W_\rho(ag)$.

Пусть $\Gamma_i = (Ag_i, E_i, \mathrm{coo}_i)$, $Ag_i \subseteq Ag$, $E_i \subseteq Ag^2$, $\mathrm{coo}_{\Gamma_i} : \mathbb{Z}_{\geqslant 0} \times Ag_i \to \Omega_h$ – функция соответствия вершины графа Γ_i и клетки Ω_h, $i = 1{,}2$ – два графа, описывающие строи агентов. Определим расстояние несходства между ними в момент времени t как

$$\mathrm{dist}(\Gamma_1(t), \Gamma_2(t)) = m_0 + m_1 + \kappa \sum_{ag \in Ag_c} \|\mathrm{coo}_{\Gamma_1}(t, ag) - \mathrm{coo}_{\Gamma_2}(t, ag)\|,$$

где $Ag_c \subseteq Ag$ – множество вершин наибольшего общего подграфа Γ_1 и Γ_2, $m_0 = |Ag_0 \backslash Ag_c|$, $m_1 = |Ag_1 \backslash Ag_c|$, $\kappa = (\mathrm{diam}(\Omega_h))^{-1}$ – диаметр множества Ω_h в клетках. Это расстояние в своей сущности является частным случаем ранее введенного в [11] автором расстояния графов.

2.5.2 Синтез локальной функции перехода

Обратим внимание на то, что у двумерного *robot-space cellular automaton* окрестность $V_r(i,j)$ каждой ячейки в клетке (i,j) порождена окрестностью (i,j) в обычной евклидовой метрике $\mathbb{R}^2$ и алгоритм нахождения скорейшего маршрута учитывает такую окрестность. Однако, алгоритм построения строя основан именно на *robot-space cellular automaton* и в качестве множества ячеек рассматривает Ag, а в качестве окрестности агента $ag \in Ag$ – окрестности $W_\rho(ag)$, порожденные геодезическим расстоянием на графе строя Φ.

Усовершенствуем алгоритм 1 так, чтоб агенты стремились в каждый момент времени поддерживать формацию, заданную графом Φ. Отметим, что без существенных потерь во времени прохождения поддерживать в точности заданную формацию можно лишь на ландшафтах с относительно малым количеством препятствий (иначе говоря, с малой конфигурационной энтропией).

Если предположить, что строй агентов в момент времени t описывается графом $\Gamma(t)$, то агенты стараются выбрать такое направление движения, которое было бы компромиссным между направлением, минимизирующим $\mathrm{dist}(\Phi(t), \Gamma(t))$ (точнее, минимизирующим расстояние между графом строя и той части реального графа строя, которая известна агенту) и направлением в сторону кратчайшего по времени пути.

В рамках такой логики агентов, существуют следующие виды взаимодействия:

1. $\mathrm{predict} : Ag \to \Omega_h$ – функция, с помощью которой агент предсказывает новое положение другого агента на следующем такте функционирования КА. Поскольку каждый агент каждый такт вычисляет и сортирует возможные направления движения по желательности (алгоритмы 3 и 7), то $\mathrm{predict}(ag)$ может возвращать, например, $d_0 + (i,j)$, где d_0 самое желательное для агента в клетке (i,j) направление. Проще говоря, все агенты содержат одинаковый алгоритм поведения. Поэтому каждый агент может «мысленно» подставить себя на место другого и просчитать, куда бы пошел он сам в этой ситуации. Результат такого подсчета и возвращает функция $\mathrm{predict}$. Более формально, агент вызывает функцию вычисления весов не со своими координатами на плоскости и не со своей позицией в графе строя, а с координатами и позицией в строю того агента, для которого он хочет предсказать положение.
2. Рекурсивный запрос остановки. Агент просит остановиться своего лидера, который просит остановиться своего лидера и т.п., пока запрос не дойдет до агента, не имеющего лидера.
3. Рекурсивный запрос продолжения движения. Агент просит продолжить движение своего лидера, который просит продолжить движение у своего лидера и т.п., пока запрос не дойдет до агента, не имеющего лидера.

Отметим, что поскольку $\mathrm{dist}(\Phi, \Gamma) \geqslant 0$ и $T_h(r_h) > 0$ для любых графов Φ, Γ и маршрута r_h, то для скаляризации вышеупомянутых целевых функций их проще всего сложить. Идея алгоритма состоит в том, что

1. Находящиеся в одной окрестности $V_r(i,j)$ и в одной окрестности $W_\rho(ag)$ агенты выбирают лидера (алгоритм 6), за которым следуют, причем так, чтоб лидер агентов одной окрестности обязательно бы следовал за лидером агентов в другой, кроме лидера самого высокого уровня. Множество агентов из $W_\rho(ag)$, найденных агентом ag в окрестности $V_r(i,j)$ обозначим как $found(ag)$.
2. Каждому возможному направлению $d \in \mathcal{D}$ движения агента ag в клетке (i,j) присваиваются веса, учитывающие как направление в сторону кратчайшего пути, аналогично (2.30), так и в сторону сохранения строя,

например

$$weights[d] = \min_{r_h \in \mathcal{M}_d(i,j;i_r,j_r)} T_h(r_h) + \\ + \sum_{ag' \in found(ag)} \|\mathrm{coo}_\Phi(t, ag') + d - \mathrm{predict}(ag')\|.$$

3. Множество $\mathcal{D}$ случайно перемешивается в соответствии с весами $weights$, получая множество $\mathcal{D}'$. Автор использует предложенный в [108] алгоритм случайного взвешенного перемешивания. Без перемешивания возможна ситуация, когда веса направлений движения в нескольких направлениях будут равны и неясно, какой вес выбрать. Также, во многих случаях агенту будет тогда неясно, что предпочесть – удерживать строй и проиграть в скорости, или же отклониться от строя и выиграть в скорости. Более того, перемешиванием моделируется ситуация, когда люди, например, неверно угадывают направление движения соседа и сталкиваются с ним.
4. Агент выбирает лучшее направление из перемешанного $\mathcal{D}'$, если не слишком отстал от лидера. Поскольку перемешивание не равномерно, а учитывает веса (т.е. самые оптимальные маршруты имеют самый большой шанс оказаться первыми) с помощью особого алгоритма, то поддержание строя от такого перемешивания практически не страдает. У экспоненциального распределения, которое используется для перемешивания, очень легкий хвост и вероятность выбора направления, сильно нарушающего строй, близка к нулю.
5. Если агент слишком отстал, то для движения за лидером агент в клетке (i,j) вычисляет положение лидера (i_f,j_f) согласно своему шаблону формации, и ищет в окрестности $V_\varepsilon(i_f,j_f)$ своего лидера. При обнаружении лидера ag_l в $(i_l,j_l) \in V_\varepsilon(i_f,j_f)$ агент выбирает такое направление движения d, чтоб минимизировать $\|(i,j) + d - \mathrm{predict}(ag_l)\|$.
6. Если в окрестности $V_r(i,j)$ ведущего агента ag слишком много агентов, не принадлежащих $W_\rho(ag)$ или, наоборот, не хватает агентов, принадлежащих $W_\rho(ag)$, то ag пытается выбрать другое направление (строка 24 алгоритма 2) или запросить «лишних» или «недостающих» агентов выбрать другое направление. Подробный разбор механизма обмена сообщениями в среде «Психоход» [36] выходит за рамки настоящей

работы, но, в целом, когда агент обнаруживает, что ему пришло сообщение типа «запрос другого направления», то он генерирует веса направлений заново, причем устанавливает вес ранее выбранного направления в 0, делая невозможным выбор этого направления.

Рекурсивные запросы от ведомых агентов к лидерам нужны для восстановления сильных нарушений строя или на очень неоднородных ландшафтах, когда самоорганизация для необщающихся агентов не работает. При этом, если лидер ag заметил отклонение от строя ведомых агентов из множества $found$, то он вместо остановки выбирает направление движения

$$\mathrm{def}(ag) = \frac{\sum_{ag' \in found}(\mathrm{coo}_\Gamma(ag') - \mathrm{coo}_\Phi(ag'))}{\| \sum_{ag' \in found}(\mathrm{coo}_\Gamma(ag') - \mathrm{coo}_\Phi(ag'))\|}.$$

Подробно вышеуказанное поведение агентов может быть описано как алгоритм 2, в который входят алгоритмы 5 – 7, приведенные в подразделе 2.5.4.

Алгоритм 2 ((Поддержания строя))**.**

1: **for all** $ag \in Ag$ **do**
2: $weights =$ SORTDIRECTIONS(ag) ▷ Алгоритм 3
3: $predicted$, $less$, $more$, $found =$ FINDNEIGHBOURS(ag, ε) ▷ Алгоритм 5
4: **if** t == 0 **then**
5: FINDLEADER(ag, $found$) ▷ Алгоритм 6
6: **end if**
7: $distances =$ FINDDIRECTION(ag, $found$, $predicted$) ▷ Алгоритм 7
8: Случайно перемешать $ag.\mathcal{D}'$ в соответствии с весами из $weights$ и $distances$.
9: $d_0 = ag.\mathcal{D}'[1]$ ▷ Лучшее направление
10: **if** $ag.lead \wedge distances[d_0] \gg r_0$ **then** ▷ r_0 – допустимое расхождение со строем
11: $trgtTmp = trgtAct$ ▷ Назначить временную цель
12: $trgtAct = (ag.@cell \to coo) + \mathrm{def}(ag)$
13: Рекурсивный запрос остановки агента с УИд $ag.leadId$
14: **else if** $ag.lead \wedge distances[d_0] \approx r_0 \wedge$ был рекурсивный запрос остановки **then**
15: Продолжить движение
16: Рекурсивный запрос продолжить движение агента с УИд $ag.leadId$
17: **else if** $\neg ag.lead \wedge distances[d_0] \gg r_0$ **then**

18: Рекурсивный запрос остановки агента с УИд $ag.leadId$

19: $trgtTmp = trgtAct$ ▷ Назначить временную цель

20: $trgtAct = found[leadId] - (i_s,j_s)$ ▷ $(i_s,j_s,ag.selfId) \in ag.formTmpl$ – собственное положение агента в строю

21: **else if** $distances[d_0] \approx r_0 \wedge$ была назначена временная цель **then**

22: $trgtAct = trgtTmp$

23: Рекурсивный запрос продолжить движение агента с УИд $ag.leadId$

24: **else if** длина $more$ или $less$ слишком велика **then**

25: Случайно перемешать $ag.\mathcal{D}'$ в соответствии с весами из $weights$ и $distances$.

26: **end if**

27: **end for**

28: **for all** $ag' \in Ag$ **do** ▷ Актуальное перемещение агентов

29: ACTUALIZE(ag') ▷ Алгоритм 4

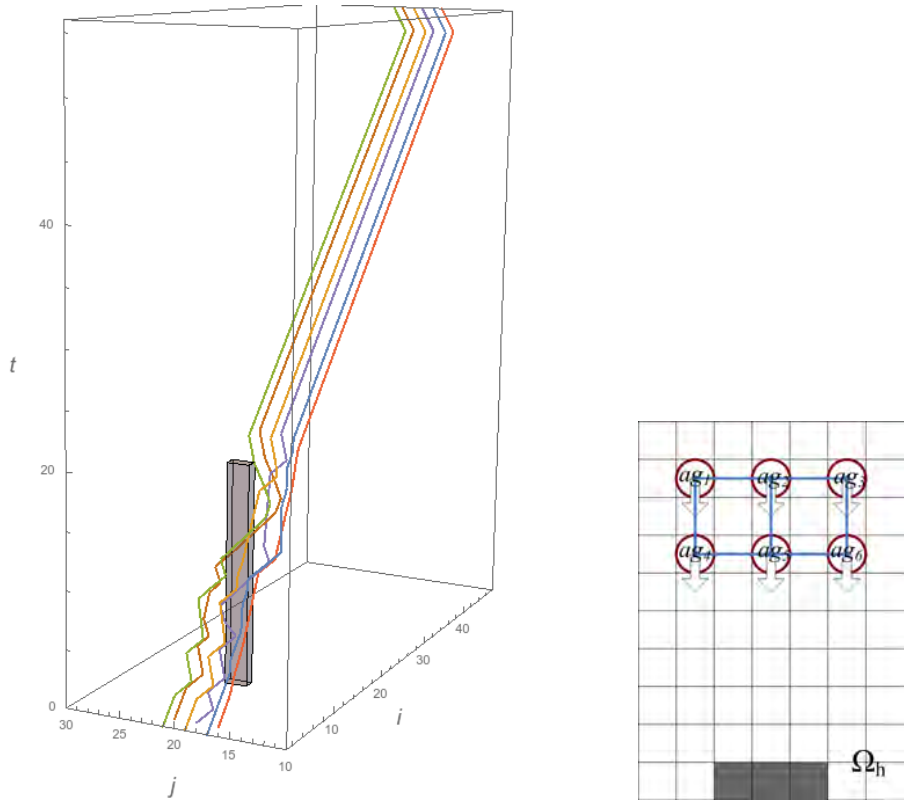
30: **end for**

31: t = t+1

2.5.3 Результаты симуляции

Предложенный алгоритм организации строя был смоделирован в программе «Психоход»[1], разработанной автором [36], и при наличии небольших одиночных препятствий агенты ведут себя так, как показано на рис. 2.11а. На указанном рисунке показаны шесть агентов ag_i, $i = \overline{1,6}$, имеющих граф строя, показанный на рис. 2.11б. Для ag_2, ag_4 лидером является агент ag_1, для ag_3, ag_5 – агент ag_2, для ag_6 – агент ag_3. Агенты пытаются наименее времязатратным образом обойти препятствие (показанное в виде прямоугольника), после чего восстанавливают строй. Однако, когда ландшафт становится более неоднородным, эффективность построения строя снижается. Это происходит потому что агенты теряют способность адекватно предсказывать направления движения друг у друга и агентам негде развернуть правильный строй после его нарушения из-за препятствия, т.к. препятствия встречаются слишком часто.

[1] `https://bitbucket.org/bokohodteam/bokohod`



а) б)

Рисунок 2.11 — Обход небольшого одиночного препятствия строем из шести агентов.

Обозначим

$$\Delta\Phi = \frac{1}{T}\sum_{ag\in Ag}\sum_{t=\overline{1,T}} \|\mathrm{coo}_\Phi(t,ag) - \mathrm{coo}_\Gamma(t,ag)\|,$$

где $\mathrm{coo}_\Phi(t,ag)$ – желаемое положение агента ag в такт $t = \overline{1,T}$ согласно графу строя Φ, $\mathrm{coo}_\Gamma(t,ag)$ – реальное положение агента ag в такт $t = \overline{1,T}$. Моделирование[2] вышеупомянутого строя из шести агентов на «естественном ландшафте», типа указанного в разделе 2.6.1 дает результат, приведенный на рис. 2.12. На этом рисунке показана зависимость $\Delta\Phi/6$ (множитель 1/6 взят, потому как агентов шесть, и смысл величины $\Delta\Phi/6$ – это среднее отклонение агента от своей позиции в строю) от количества препятствий N_{obst}, которая аппроксимируется с коэффициентом детерминации $R^2 = 0.992729$ как

$$\Delta\Phi/6 \approx 0.901073\ln(N_{obst}+1) - 0.673763.$$

Выяснилось, что уменьшить отклонение агентов от строя возможно, увеличивая

[2]Данные доступны по адресу `https://www.researchgate.net/publication/318663064_Eksperimentalnye_dannye_otnositelno_obhoda_prepatstvij_stroem_iz_sesti_agentov`

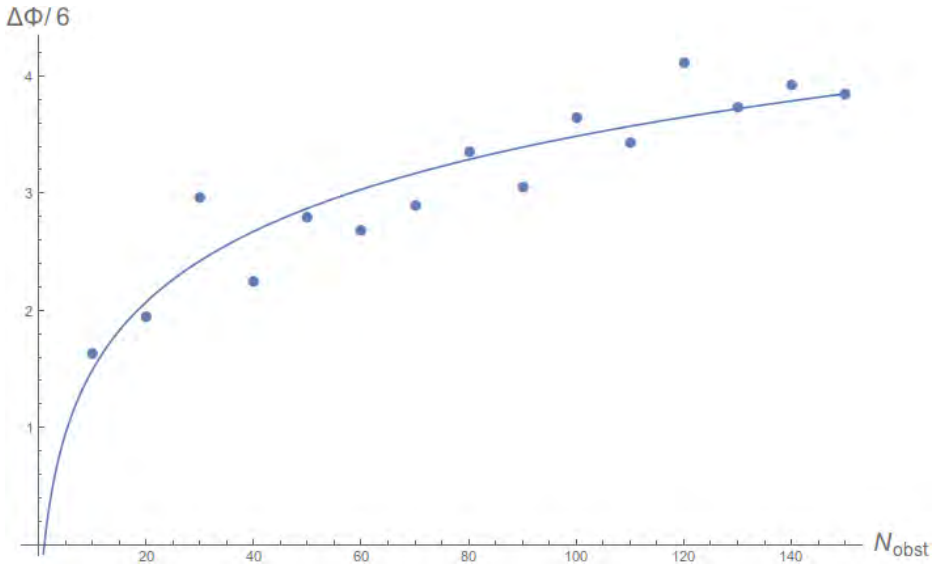


Рисунок 2.12 — Зависимость среднего отклонения позиции агентов от предусмотренной строем и количества препятствий

параметр ε, указывающий, в насколько широкой окрестности $V_\varepsilon(i_f,j_f)$ заданной шаблоном строя, $(i_f,j_f,agId)$ следует искать агента с уникальным идентификатором $agId$.

2.5.4 Псевдокод используемых алгоритмов

Продемонстрируем примеры проходимых ландшафтов различного вида. Подробное описание способов генерации ландшафтов приведено в работах [8; 21]. Также покажем псевдокод алгоритмов, необходимых для построения строя и поиска оптимального по времени маршрута.

Алгоритм 3 ((Сортировки направлений движения)).

1: **function** SORTDIRECTIONS(ag)
2: $(i,j) = ag.@cell \to coo$
3: $weights = \{\}$
4: **if** $ag.w > 0$ **then** $ag.w = ag.w - 1$
5: **end if**
6: **if** $ag.w == 0 \wedge (i,j) \neq ag.trgtAct$ **then**
7:

$$direction = \frac{(i,j) - ag.trgtAct}{\|(i,j) - ag.trgtAct\|}$$

8: (i_r,j_r) = клетка, в которой с границей $V_r(i,j)$ пересекается луч с началом в (i,j) и направленный в направлении $direction$

9: $ag.\mathcal{D}' = \mathcal{D}$

10: **for all** $d \in \mathcal{D}$ **do**

11: $weights[d] = \min_{r_h \in \mathcal{M}_d(i,j;i_r,j_r)} J(r_h)$

12: **end for**

13: **end if**

14: **return** $weights$

15: **end function**

Алгоритм 4 ((Актуализации новых положений)).

1: **function** ACTUALIZE(ag')

2: $k = 1$

3: $(i,j) = ag'.@cell \to coo$

4: **repeat**

5: $(i_{new},j_{new}) := (i,j) + ag'.\mathcal{D}'[k]$

6: $k = k + 1$

7: **until** $ag'_{i_{new},j_{new}} \neq 0 \vee (i_{new},j_{new}) \neq (i,j)$

8: $cell_{i_{new},j_{new}}.@ag = @ag'$

9: $ag'.@cell = @cell_{i_{new},j_{new}}.coo$

10: **end function**

Алгоритм 5 ((Обнаружения соседей)).

1: **function** FINDNEIGHBOURS(ag, ε)

2: $predicted = \{\}$, $less = \{\}$, $more = \{\}$, $found = \{\}$

3: $(i,j) = ag.@cell \to coo$

4: **for all** $(s,p,agId) \in ag.formTmpl$ **do** ▷ Поиск в окрестности $W_\rho(ag)$

5: Искать в окрестности $V_\varepsilon(i+s,j+p)$ агента

6: **if** агент ag' найден и $agId == 0$ **then**

7: $agId = ag'.selfId$

8: $predicted[agId] = \text{predict}(ag')$

9: Добавить в $found$ $agId$

10: **else if** агент ag' найден и $agId == ag'.selfId$ **then**

11: $predicted[agId] = \text{predict}(ag')$

12: Добавить в $found$ $agId$

13: **else if** агент ag' найден и $agId \neq ag'.selfId \wedge agId \neq 0$ **then**

14: Добавить в $more$ $ag'.selfId$ ▷ длина $more$ – это m_1 из определения метрики
15: **end if**
16: **if** Длина($ag.formTmpl$) $>$ Длина($predicted$) **then**
17: Добавить в $less$ УИд не найденных агентов ▷ длина $less$ – это m_0 из определения метрики
18: **end if**
19: **end for**
20: **return** $predicted$, $less$, $more$, $found$
21: **end function**

Алгоритм 6 ((Назначения и обнаружения лидера)).

1: **function** FINDLEADER(ag, $found$)
2: $ag.leadId = 0$
3: **for all** $id \in found$ **do** ▷ агент выбирает лидером агента с самым маленьким УИд из своих соседей
4: **if** $id < ag.selfId$ **then**
5: $ag.leadId = id$
6: **end if**
7: **end for**
8: **end function**

Алгоритм 7 ((Поиска направления для поддержания формации)).

1: **function** FINDDIRECTION(ag, $found$, $predicted$)
2: $dist = 0$, $distances = \{\}$
3: $(i,j) = ag.@cell \rightarrow coo$
4: **for all** $d \in \mathcal{D}$ **do**
5: **for all** $id \in found$ **do**
6: $dist = dist + \|d + (i,j) - predicted[id]\|$
7: **end for**
8: $distances[d] = dist$
9: **end for**
10: **return** $distances$
11: **end function**

2.6 Методы тестирования алгоритмов группового движения агентов с использованием наборов случайных ландшафтов

2.6.1 Сравнение квазиоптимальных оптимальных траекторий

Получившийся в результате работы алгоритма 1 маршрут сравнивался с глобально оптимальным, который был найден с помощью алгоритма Дейкстры. Для разных типов ландшафтов результаты сравнения получились довольно разными. Определим отклонение длины Λ маршрута $r_h(t) = (i(t), j(t))$ от оптимального по времени маршрута $r_{opt}(t) = (i_{opt}(t), j_{opt}(t))$, $t \in \mathbb{Z}_{\geqslant 0}$

$$\delta\Lambda = \frac{|\Lambda(r) - \Lambda(r_{opt})|}{\Lambda(r_{opt})} \cdot 100\%,$$

и отклонение времени T прохождения маршрута r_h от времени T_{opt} прохождения оптимального маршрута r_{opt}

$$\delta T = \frac{|T - T_{opt}|}{T_{opt}} \cdot 100\%.$$

Зададим функцию $N(S) = 0.00160518\mathrm{e}^{4.22769S}$ зависимости количества препятствий от конфигурационной энтропии ландшафта. Для ландшафтов одного типа это будет в действительности количество препятствий, для ландшафтов другого типа эта величина применяется для того, чтоб их можно было легче сравнивать с другими.

Исследовались квадратные случайные ландшафты $\mathcal{L}(n \times m, l)$ со сторонами в $n = m = 100$ клеток, с $l = 9$ различными классами клеток. Агент имел радиус обзора $r = 5$ и функционал поиска оптимального маршрута вида (2.32).

Было произведено по 1000 экспериментов для каждого значения $N(S)$ в случайном ландшафте $\mathcal{L}(100 \times 100, 9)$. В каждом эксперименте агент перемещался из клетки (1,1) в клетку (98,98)[3]. Оказалось, что при возрастании конфигурационной энтропии S в ландшафте, состоящем из равномерно распределенных клеток разной непроходимости (см. раздел 2.2.2) среднее отклонение

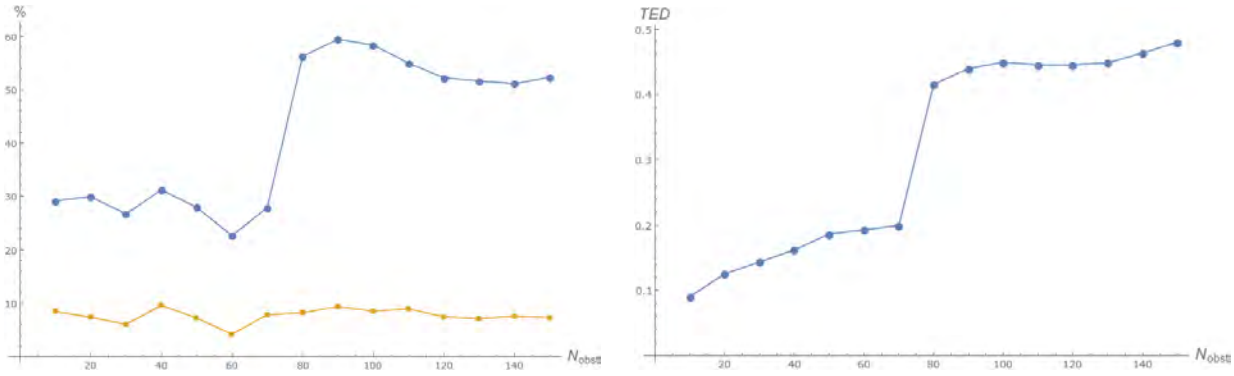
[3]Данные доступны по адресу `https://www.researchgate.net/publication/319998893_Eksperimentalnye_dannye_po_sravneniu_kvazioptimalnoj_i_optimalnoj_traektorij_agenta`

длины и времени от оптимальных (рис. 2.13а) медленно убывает, причем в районе $N(S) = 70$ наблюдается небольшая ступенька (более выраженная у ландшафтов малой площади), очевидно связанная со ступенькой в зависимости среднего значения TED от $N(S)$ (рис. 2.13б).

Однако, при возрастании количества препятствий в ландшафте, состоящем из нескольких максимально труднопроходимых клеток, которые окружены менее непроходимыми, которые, в свою очередь, окружены еще менее непроходимыми и т.д. («естественные ландшафты» из подраздела 2.2.4), характеристики алгоритма поиска несколько иные. При росте количества максимально непроходимых клеток N_{obst} и соответствующем росте конфигурационной энтропии (для такого ландшафта справедливо соотношение $N_{obst} = N(S)$), среднее отклонение длины и времени от оптимальных медленно убывает, причем в районе $N_{obst} = 70$ наблюдается резкий скачок (рис. 2.14а и 2.14б), после которой опять наблюдается медленное убывание среднего отклонения. Автор полагает, что именно такие ландшафты, а не полностью равномерно случайные, более соответствуют встречающимся в приложениях и что отклонение от оптимального маршрута не более 10% по длине и порядка 30% по времени вполне достаточно для многих практических нужд. Следует учесть, что при поиске локально оптимального маршрута алгоритмом Дейкстры требуется всего примерно $O((2r+1)^4)$ времени. При «склейке» квазиоптимального маршрута длиной L клеток из локально оптимальных временная сложность будет, таким образом, примерно $O((2r+1)^4 L)$ при пересчете направления маршрута каждый ход, при котором изменяется положение агента, или $O((2r+1)^3 L)$ при пересчете маршрута лишь по достижении границы окрестности вместо $O((nm)^2)$ при поиске глобально оптимального маршрута, что дает определенный выигрыш при $r^2 \ll nm$.

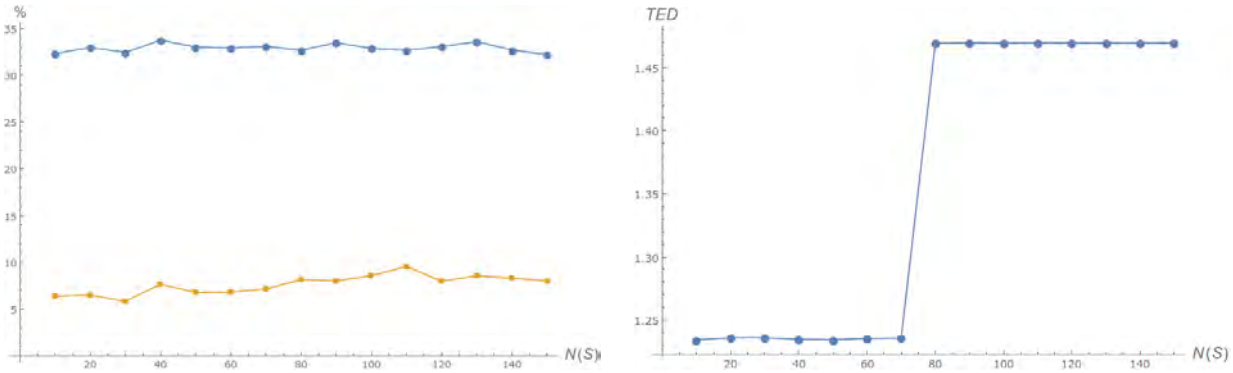
Положим $l = 9$, $n = m = 48$, $o = 6$, выберем $N_{obst} \in \{5\} \cup \{10i | i = \overline{1,25}\} \cup \{255\}$. Создадим ландшафт для каждого значения N_{obst} и выполним следующий эксперимент[4]. Пусть агент перемещается из ячейки ω_{11} в ячейку ω_{nn} 100 раз в соответствии с ранее описанным алгоритмом. Затем мы вычислим время, необходимое для завершения движения агента T^i_{bok} и время перехода от ω_{11} к ω_{nn} по прямой линии T^i_{tup}. Затем мы вычислим среднее значение и стандартное

[4]Исходные данные и программа обработки данных для всех экспериментов хранятся в https://www.researchgate.net/publication/316747096_The_experimental_data_for_gro



а) б)

Рисунок 2.13 — Проигрыш времени (круглые маркеры) и отклонение от траектории (квадратные) в процентах по сравнению с оптимальным маршрутом для равномерного случайного ландшафта



а) б)

Рисунок 2.14 — Проигрыш времени (круглые маркеры) и отклонение от траектории (квадратные) в процентах по сравнению с оптимальным маршрутом для «естественного» случайного ландшафта

отклонение выигрыша времени для всех этих серий:

$$\overline{win} = \frac{1}{50}\sum_{i=1}^{50}\frac{T^i_{tup}}{T^i_{bok}}.$$

$$\sigma = \left(\sum_{i=1}^{50}(\overline{win} - \frac{T^i_{tup}}{T^i_{bok}})^2\right)^{1/2}.$$

Было обнаружено, что среднее значение выигрыша во времени прохождения $\overline{win}$ и конфигурационной энтропии ландшафта S коррелирует с коэффициентом корреляции 0.959556 (см. рис. 2.15). $\overline{win}$ и TED ландшафта коррелируют с коэффициентом корреляции 0.964763. Оранжевая линия на рисунке соответствует кривой $y = (S(N_{obst}) + 1)\ln 9$, коричневая линия соответствует

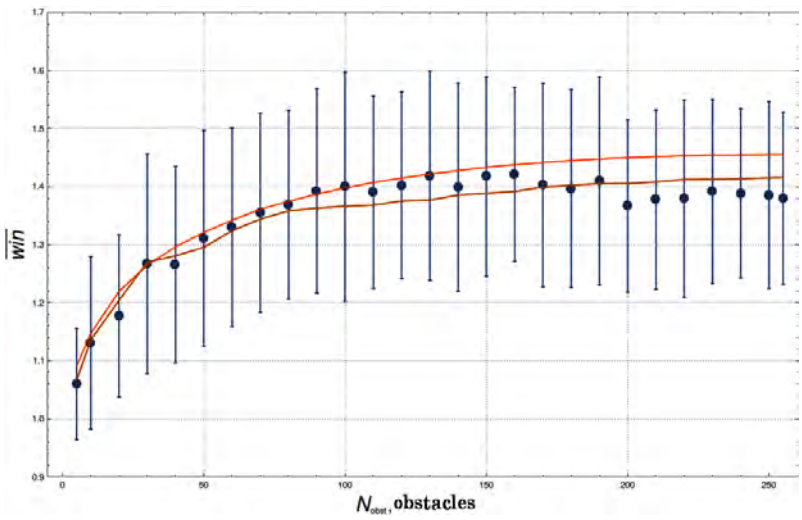


Рисунок 2.15 — Относительный выигрыш времени

кривой

$$y = 0.922178 + 0.539383TED(N_{obst}),$$

где $S(N_{obst})$ и $TED(N_{obst})$ – средняя энтропия и среднее значение плотности краевой total edge density ландшафта с N_{obst} препятствиями. Вертикальные линии соответствуют стандартным отклонениям выигрыша времени.

2.6.2 Движение агента в точку назначения как реализация броуновского моста

Движение агента по случайному ландшафту можно рассматривать как реализацию броуновского моста. В работе [110] рассматривалась модель движения броуновского моста (BBMM), которая обеспечивает биологически обоснованное приближение пути движения животного на основе дискретных данных местоположения и является мощным методом количественной оценки распределения использования (utilization distribution, т.е. распределения, сконструированного из данных о местоположении животного в пространстве в различные моменты времени). В данном разделе исследуются статистические свойства квазиоптимальных траекторий агента, получаемых с помощью предложенного в диссертации клеточного автомата. Эти траектории также интерпретируются как реализации броуновского моста и используются предположения о распределении скорости агента, аналогичные сделанным в вышеупомянутой статье.

На рис. 2.16а приводятся гистограммы распределения времени прибытия в конечную точку маршрута для равномерного ландшафта ($N_{obst} = 130$). Слева на рисунке изображена гистограмма распределения времени прибытия в конечную точку для оптимального маршрута, найденного алгоритмом Дейкстры, справа – для квазиоптимального маршрута. Данные гистограммы примерно соответствуют (с p-значениями около 0,7) распределению Нортона-Райса с функцией плотности

$$p(m,\alpha,\beta;x) = \begin{cases} \dfrac{\alpha m\left(\frac{x}{\alpha}\right)^m e^{-\frac{m\left(\alpha^2+x^2\right)}{2\beta^2}} I_{m-1}\left(\frac{mx\alpha}{\beta^2}\right)}{\beta^2}, & x > 0, \\ 0, & x \leqslant 0, \end{cases}$$

где I_k – модифицированная функция Бесселя первого рода.

Аналогичные гистограммы приведены на рис. 2.16б для распределения времени прибытия в конечную точку маршрута для «естественного» ландшафта ($N_{obst} = 130$).

2.6.3 Выявление зависимости статистических свойств квазиоптимального маршрута от характеристик проходимой местности

В данном подразделе будут получены эмпирические функции плотности распределения для времени выхода на границу и распределения использования для модели движения агента в виде броуновского моста, причем случайное блуждания приближалось с помощью дискретного случайного процесса, моделируемого с помощью клеточного автомата. Из вида аналитической модели движения будут сделаны предположения относительно теоретических функций плотности распределения и получены оценки p-значений для данных гипотез.

Если ландшафт $\mathcal{L}(\Omega_h, l)$ состоит из клеток, непроходимости которых распределены случайно, то каждый найденный агентом квазиоптимальный маршрут на определенном случайном ландшафте можно рассматривать как реализацию случайного блуждания на решетке, точнее, как броуновский мост. В этом случае

$$\tau = \underset{\mathbf{r}\in\mathcal{Y}}{\overset{\approx}{\min}} T(\mathbf{r}), \quad \tau_h = \underset{\mathbf{r}_h\in\mathcal{Y}_h}{\overset{\approx}{\min}} T_h(\mathbf{r}), \quad \tau_{\min} \leqslant \tau_h \leqslant \tau_{\max}$$

будут соответствовать времени выхода агента в конечную точку в непрерывном и дискретном случаях соответственно. Далее мы попытаемся определить закон распределения для случайных величин τ, τ_h и

$$\boldsymbol{\varrho} = \arg \underset{\mathbf{r}\in\mathcal{Y}}{\overset{\approx}{\min}} T(\mathbf{r}), \quad \boldsymbol{\varrho}_h = \arg \underset{\mathbf{r}_h\in\mathcal{Y}_h}{\overset{\approx}{\min}} T(\mathbf{r}_h)$$

экспериментально и из теоретических соображений и провести сравнение результатов. Здесь знак «$\overset{\approx}{\min}$» означает «является квазиминимумом». В качестве случайных ландшафтов мы будем брать ранее введенные автором «натуральные» и «равномерные» ландшафты [8].

В случае «равномерного» ландшафта $\mathrm{u}(t, \boldsymbol{\rho}) \in [1, u_{\max}]$,

$$u_{\max} = \max_{(t,\boldsymbol{\rho})\in[0,T]\times\Omega} \mathrm{u}(t,\boldsymbol{\rho}),$$

близко к равномерно распределенной случайной величине для всех $(t,\boldsymbol{\rho}) \in \mathbb{R}_{\geqslant 0} \times \Omega$. Таким образом,

$$\mathrm{u}^c(t, \boldsymbol{\rho}) = \frac{\mathrm{u}^c_{\max}}{\mathrm{u}(t,\boldsymbol{\rho})}$$

имеет обратное равномерное распределение с функцией плотности

$$invu(x) = \begin{cases} \frac{\mathrm{u}^c_{\max}}{u_{\max}-1} x^{-2} & , x \in [\frac{\mathrm{u}^c_{\max}}{u_{\max}}, \mathrm{u}^c_{\max}] \\ 0 & , x \notin [\frac{\mathrm{u}^c_{\max}}{u_{\max}}, \mathrm{u}^c_{\max}] \end{cases}$$

Далее, случайная величина $\|\dot{\mathbf{r}}(t)\|$ имеет для $t \in [0,T_{\max}]$ распределение Райса с функцией плотности (при $x \geqslant 0$)

$$rice(\nu,\sigma;x) = \frac{x}{\sigma^2(t)} \exp\left(\frac{-(x^2+\nu^2(t))}{2\sigma^2(t)}\right) I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right),$$

если исходить из того, что скорость агента имеет 2-мерное нормальное распределение, что является обычным предположением [110]. Обозначим функцию под интегралом (2.15) как

$$\zeta(t) = \frac{\|\dot{\mathbf{r}}(t)\|}{\mathrm{u}^c(t,\mathbf{r}(t))} = \frac{1}{\mathrm{u}^c_{\max}} \|\dot{\mathbf{r}}(t)\| \mathrm{u}(t, \mathbf{r}(t)).$$

Исходя из вышеуказанного, функция распределения подынтегральной функции $\zeta(t)$ имеет вид

$$\mathbb{P}\{\zeta(t) < z\} = \iint_{D_z} rice(\nu(t),\sigma(t);x)\Pi(y)\mathrm{d}x\mathrm{d}y, \tag{2.34}$$

где $D_z = \{(x,y) \in \mathbb{R}^2 | xy < z\}$,

$$\Pi(y) = \begin{cases} 1, y \in [\frac{1}{\mathrm{u}^c_{\max}}, \frac{u_{\max}}{\mathrm{u}^c_{\max}}], \\ 0, y \notin [\frac{1}{\mathrm{u}^c_{\max}}, \frac{u_{\max}}{\mathrm{u}^c_{\max}}]. \end{cases} .$$

2.6.4 Аналитический подход к нахождению распределения времени выхода на границу

В данном подразделе аналитически ищется функция распределения (2.34) для подынтегральной функции $\zeta(t)$ и ее функция плотности. Отметим, что работа по непосредственному вычислению интегралов была выполнена Э.Л. Шишкиной и автору диссертации принадлежит только постановка задачи и сравнение качества аппроксимаций найденных функций. Пусть $\frac{1}{u^c_{\max}} = 1$ и $\frac{u_{\max}}{u^c_{\max}} = b$. Мы имеем

$$CDF_\zeta(z) = \mathbb{P}\{\zeta(t) \leqslant z\} = \iint\limits_{D_z} rice(\nu(t), \sigma(t); x)\Pi(y)\mathrm{d}x\mathrm{d}y =$$

$$= \frac{1}{\sigma^2(t)} \iint\limits_{D_z} x \exp\left(\frac{-(x^2 + \nu^2(t))}{2\sigma^2(t)}\right) I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \Pi(y)\mathrm{d}x\mathrm{d}y,$$

где

$$\Pi(y) = \begin{cases} 1, y \in [1, b], \\ 0, y \notin [1, b]. \end{cases}$$

Таким образом,

$$CDF_\zeta(z) = \frac{1}{\sigma^2(t)} e^{-\frac{\nu^2(t)}{2\sigma^2(t)}} \int\limits_1^b \mathrm{d}y \int\limits_0^{\frac{z}{y}} x e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x =$$

$$= \frac{1}{\sigma^2(t)} e^{-\frac{\nu^2(t)}{2\sigma^2(t)}} \left(\int\limits_0^{\frac{z}{b}} x e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x \int\limits_1^b \mathrm{d}y + \right.$$

$$\left. + \int\limits_{\frac{z}{b}}^{z} x e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x \int\limits_1^{\frac{z}{x}} \mathrm{d}y \right) =$$

$$= \frac{1}{\sigma^2(t)} e^{-\frac{\nu^2(t)}{2\sigma^2(t)}} \left((b-1) \int\limits_0^{\frac{z}{b}} x e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x + \right.$$

$$\left. + \int\limits_{\frac{z}{b}}^{z} (z-x) e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x \right). \tag{2.35}$$

Используя разложение в ряд $I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) = \sum\limits_{n=0}^{\infty} \frac{1}{(n!)^2} \left(\frac{\nu(t)}{\sigma^2(t)}\right)^{2n} x^{2n}$ мы получим

$$CDF_\zeta(z) = e^{-\frac{\nu^2}{2\sigma^2}} \sum_{n=0}^{\infty} \frac{\nu^{2n}(t)}{2^n (n!)^2 \sigma^{2n}(t)} \times$$

$$\times \left[\frac{z}{\sqrt{2}\sigma(t)} \left(\Gamma\left(n+\frac{1}{2}, \frac{z^2}{2b^2\sigma^2}\right) - \Gamma\left(n+\frac{1}{2}, \frac{z^2}{2\sigma^2}\right) \right) + \right.$$

$$+ \Gamma\left(n+1, \frac{z^2}{2\sigma^2}\right) - \Gamma\left(n+1, \frac{z^2}{2b^2\sigma^2}\right) +$$

$$\left. + (b-1) \left(\Gamma(n+1) - \Gamma\left(n+1, \frac{z^2}{2b^2\sigma^2}\right) \right) \right]. \tag{2.36}$$

Далее найдем функцию плотности распределения. Используя (2.35) мы получим

$$PDF_\zeta(z) = \frac{\mathrm{d}}{\mathrm{d}z} \mathbb{P}\{\zeta(t) \leqslant z\} =$$

$$= \frac{1}{\sigma^2(t)} e^{-\frac{\nu^2(t)}{2\sigma^2(t)}} \left((b-1) \frac{\mathrm{d}}{\mathrm{d}z} \int\limits_0^{\frac{z}{b}} x e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x + \right.$$

$$\left. + \frac{d}{dz} z \int\limits_{\frac{z}{b}}^{z} e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x - \frac{d}{dz} \int\limits_{\frac{z}{b}}^{z} x e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x \right) =$$

$$= \frac{1}{\sigma^2(t)} e^{-\frac{\nu^2(t)}{2\sigma^2(t)}} \left[\left(1 - \frac{1}{b}\right) \frac{z}{b} e^{-\frac{z^2}{2b^2\sigma^2(t)}} I_0\left(\frac{z\nu(t)}{b\sigma^2(t)}\right) + \right.$$

$$+ z e^{-\frac{z^2}{2\sigma^2(t)}} I_0\left(\frac{z\nu(t)}{\sigma^2(t)}\right) - \frac{z}{b} e^{-\frac{z^2}{2b^2\sigma^2(t)}} I_0\left(\frac{z\nu(t)}{b\sigma^2(t)}\right) -$$

$$- z e^{-\frac{z^2}{2\sigma^2(t)}} I_0\left(\frac{z\nu(t)}{\sigma^2(t)}\right) + \frac{z}{b^2} e^{-\frac{z^2}{2b^2\sigma^2(t)}} I_0\left(\frac{z\nu(t)}{b\sigma^2(t)}\right) +$$

$$\left. + \int\limits_{\frac{z}{b}}^{z} e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x \right] = \frac{1}{\sigma^2(t)} \int\limits_{\frac{z}{b}}^{z} e^{-\frac{x^2+\nu^2(t)}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x.$$

Therefore

$$PDF_\zeta(z) = \frac{1}{\sigma^2(t)} \int\limits_{\frac{z}{b}}^{z} e^{-\frac{x^2+\nu^2(t)}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x =$$

$$= \int\limits_{\frac{z}{b}}^{z} \frac{1}{x} rice(\nu(t),\sigma(t);x)\mathrm{d}x. \quad (2.37)$$

Затем представим функцию плотности $PDF_\zeta(z)$ в виде ряда. Используя разложение в ряд $e^{-\frac{x^2}{2\sigma^2(t)}} = \sum\limits_{n=0}^{\infty} \frac{(-1)^n}{(2\sigma^2(t))^n n!} x^{2n}$, мы получим

$$PDF_\zeta(z) = \frac{e^{-\frac{\nu^2(t)}{2\sigma^2(t)}}}{\sigma^2(t)} \int\limits_{\frac{z}{b}}^{z} e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x =$$

$$= \frac{e^{-\frac{\nu^2(t)}{2\sigma^2(t)}}}{\sigma^2(t)} \sum_{n=0}^{\infty} \frac{(-1)^n}{(2\sigma^2(t))^n n!} \int\limits_{\frac{z}{b}}^{z} x^{2n} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x =$$

$$= \frac{e^{-\frac{\nu^2(t)}{2\sigma^2(t)}}}{\sigma^2(t)} \sum_{n=0}^{\infty} \frac{(-1)^n z^{2n+1}}{(2\sigma^2(t))^n n!(2n+1)} \left({}_1F_2\left(n+\frac{1}{2};1,n+\frac{3}{2};\frac{z^2\nu^2}{4\sigma^4}\right) - \right.$$

$$\left. -\frac{1}{b^{2n+1}}\, {}_1F_2\left(n+\frac{1}{2};1,n+\frac{3}{2};\frac{z^2\nu^2}{4b^2\sigma^4}\right)\right). \quad (2.38)$$

Далее, используя разложение в ряд I_0

$$I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) = \sum_{k=0}^{\infty} \frac{1}{(k!)^2} \left(\frac{\nu(t)}{2\sigma^2(t)}\right)^{2k} x^{2k}$$

мы получим

$$PDF_\zeta(z) = \frac{e^{-\frac{\nu^2(t)}{2\sigma^2(t)}}}{\sigma^2(t)} \int\limits_{\frac{z}{b}}^{z} e^{-\frac{x^2}{2\sigma^2(t)}} I_0\left(\frac{x\nu(t)}{\sigma^2(t)}\right) \mathrm{d}x =$$

$$= \frac{e^{-\frac{\nu^2(t)}{2\sigma^2(t)}}}{\sigma^2(t)} \sum_{k=0}^{\infty} \frac{1}{(k!)^2} \left(\frac{\nu(t)}{2\sigma^2(t)}\right)^{2k} \int\limits_{\frac{z}{b}}^{z} x^{2k} e^{-\frac{x^2}{2\sigma^2(t)}} \mathrm{d}x =$$

$$= \frac{e^{-\frac{\nu^2(t)}{2\sigma^2(t)}}}{\sqrt{2}} \sum_{k=0}^{\infty} \frac{\nu^{2k}(t)}{2^k (k!)^2 \sigma^{2k+1}(t)} \left(\Gamma\left(k+\frac{1}{2},\frac{z^2}{2b^2\sigma^2(t)}\right) - \right.$$

$$\left. -\Gamma\left(k+\frac{1}{2},\frac{z^2}{2\sigma^2(t)}\right)\right). \quad (2.39)$$

Пусть $\nu(t) > 0$, $\sigma(t) > 0$ – константы. Из (2.37) и (2.15) мы можем заключить, что функция плотности распределения случайной величины τ приближенно имеет форму

$$PDF_{\tau}(t) = T_0 PDF_{\zeta}(t), \tag{2.40}$$

где T_0 – положительная константа.

2.6.5 Вычислительный эксперимент для времени выхода в конечную точку

Конфигурационная энтропия ландшафта $\mathcal{L}(\Omega_h, l)$ определяется как

$$S(\mathcal{L}(\Omega_h, l)) = -\sum_{i=1}^{l} \frac{N_i}{M} \ln \frac{N_i}{M}.$$

и характеризует неоднородность состава ландшафта в целом.

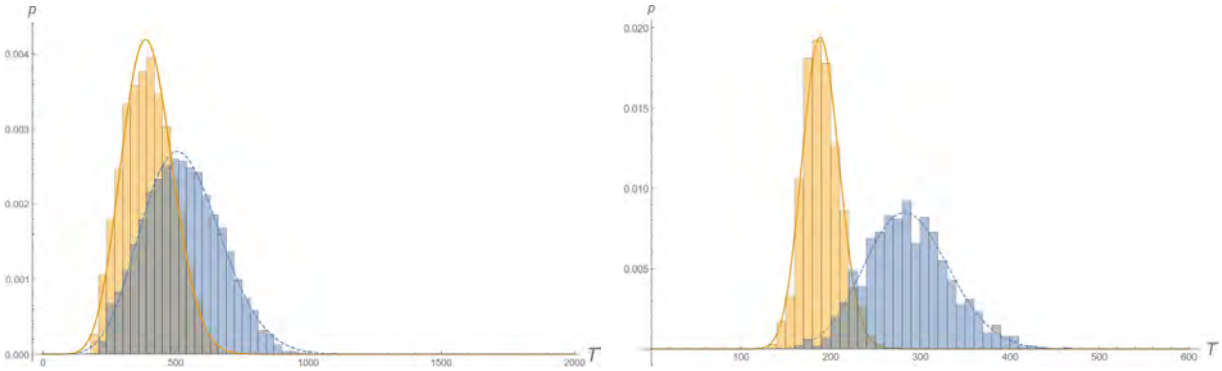
Было произведено по 10000 экспериментов в случайном ландшафте $\mathcal{L}(100\times100, 9)$ ($S = 2.125$) для оценки параметров распределения и дополнительно по 1000 экспериментов для проверки гипотезы о распределении. В каждом эксперименте агент перемещался из клетки $(1,1)$ в клетку $(98,98)$. Получились результаты, показанные в виде гистограмм на рисунке 1. На рисунке 1a показаны гистограммы распределения времени выхода в конечную точку для оптимальных (слева) и квазиоптимальных (справа) маршрутов для равномерных случайных ландшафтов, на рисунке 1b – то же самое для «естественных» ландшафтов. Далее координаты (x,y) будет совпадать с номерами клеток (i,j), если x, $y \in \mathbb{Z}$.

Для случайных равномерных ландшафтов для 10000 экспериментов получилось что

$$\tau_{\min} = 204, \tau_{\max} = 1999, \overline{\tau} = 527.923.$$

Было выяснено, что данным гистограммам наиболее соответствует (с $p-$значением около 0,7) распределение Нортона-Райса с функцией плотности

$$nrice(m,\nu,\sigma;x) = \begin{cases} \dfrac{\nu m \left(\frac{x}{\nu}\right)^m e^{-\frac{m\left(\nu^2+x^2\right)}{2\sigma^2}} I_{m-1}\left(\frac{mx\nu}{\sigma^2}\right)}{\sigma^2}, & x > 0, \\ 0, & x \leqslant 0, \end{cases} \tag{2.41}$$



а) б)

Рисунок 2.16 — Гистограммы распределения времени прибытия в конечную точку маршрута τ для равномерного (а) и «естественного» (б) ландшафтов

где I_k – модифицированная функция Бесселя первого рода. Поиск производился среди распределений самых разных типов, и гипотеза о распределении Нортона-Райса возникла именно в силу вида теоретической функции плотности (2.34) и того факта, что сумма случайных величин, имеющих распределение Райса, . В действительности, из (2.34) следует, что распределение должно выражаться в виде ряда из обобщенных гипергеометрических функций с определенными параметрами.

Особенно хорошо это соответствие проявляется для «равномерных» ландшафтов. Для проверки гипотезы о распределении применялся критерий Крамера – фон Мизеса ω^2 со статистикой вида

$$n\omega_n^2 = \frac{1}{12n} + \sum_{i=1}^{n} \left(F(x_i, \theta) - \frac{2i-1}{2n} \right)^2,$$

где $F(x, \theta)$ – теоретическая функция распределения с вектором параметров θ.

В соответствии с критерием Крамера – фон Мизеса было получено значение статистики 0.0802693, что для 1000 экспериментов позволяет сделать вывод, что гипотеза о распределении не отвергается с 15% уровнем значимости. Для равномерных ландшафтов были получены следующие значения параметров

$$m = 3.33579, \quad \nu = 11.4566, \quad \sigma = 387.896, \tag{2.42}$$

для «естественных» –

$$m = 9.47719, \quad \nu = 13.3365, \quad \sigma = 205.047.$$

2.6.6 Сравнение аналитически полученной функции распределения и экспериментальных данных

Сравним функцию плотности распределения (2.40), полученную аналитически, с функцией плотности (2.41), полученной в результате численного эксперимента. Поскольку параметры (2.40) достаточно сложны для вычисления, вместо этого мы будем использовать отрезок ряда (2.39). Заметим, что отрезок ряда (2.38) плохо аппроксимирует изучаемую функцию при больших значениях аргумента. Из рис. 2.17 видно, что функция плотности (2.40) с параметрами

$$\nu = 419.25, \sigma = 121.544, b = 1.41866, T_0 = 2.4386, \tag{2.43}$$

которая обозначена сплошной толстой линией и функция плотности (2.41) с параметрами (2.42), которая обозначена пунктирной линией, достаточно близки друг к другу и к эмпирической функции плотности. Точнее, значение статистики Крамера-Фон-Мизеса для функции плотности (2.40) с параметрами (2.43) равно $n\omega_n^2 = 0.0540313$, $n = 1000$, что намного меньше, чем значение статистики для функции плотности (2.41).

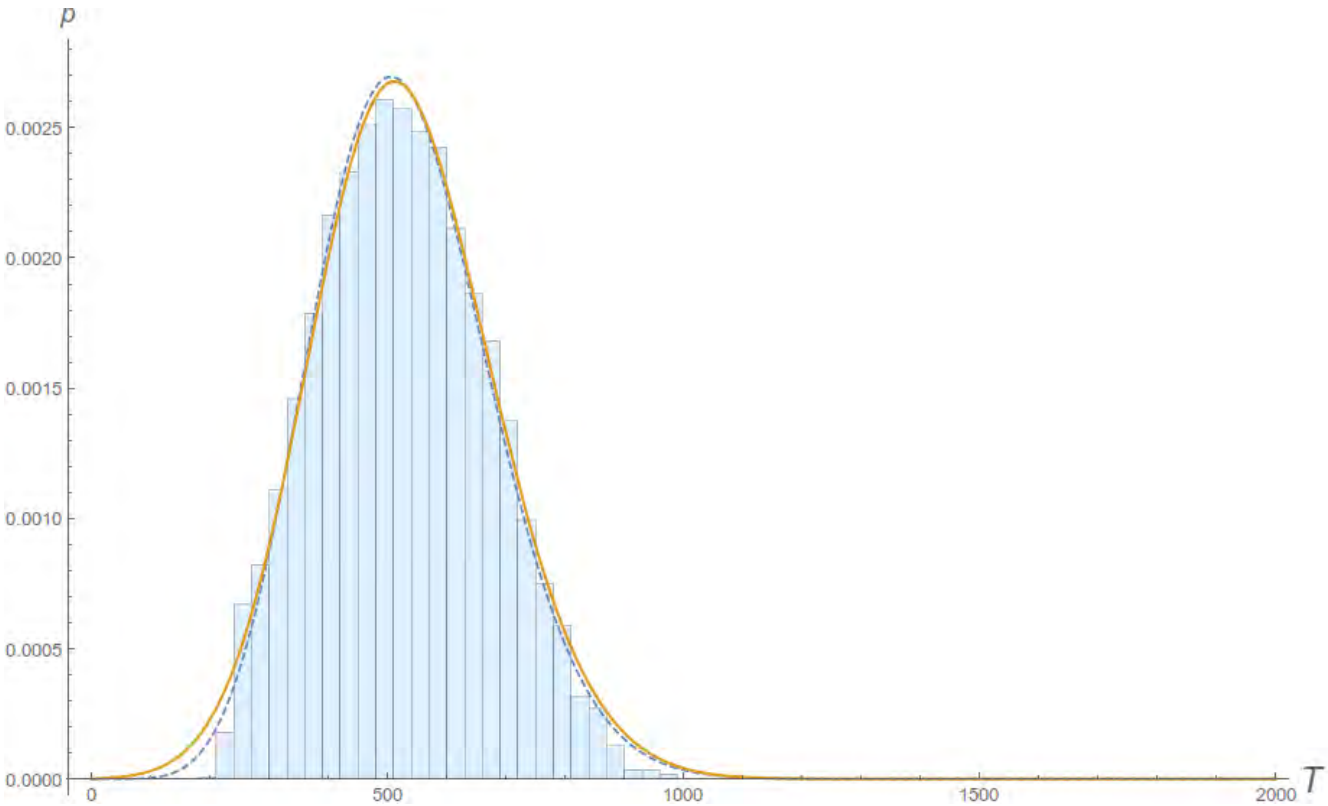


Рисунок 2.17 — Функции плотности, полученные эмпирически и аналитически

2.6.7 Вычислительный эксперимент для поиска распределения координат местоположения агента

Для распределения координат агента $\boldsymbol{\varrho}_h(t) = (x_h(t), y_h(t))$ были получены гистограммы, приведенные на рисунке 2.18.

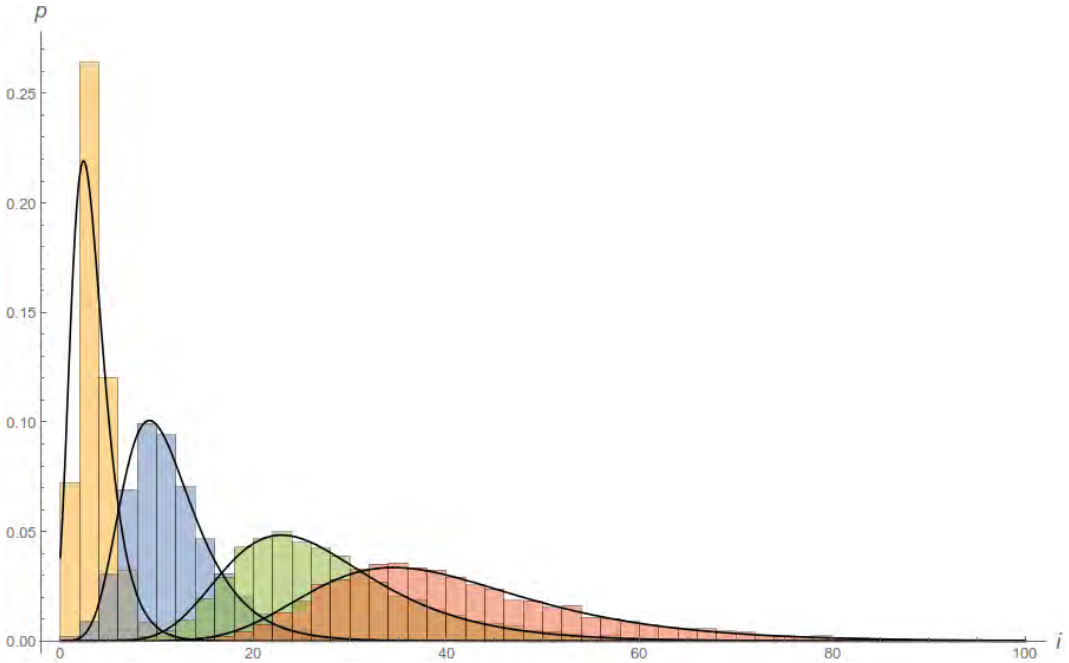


Рисунок 2.18 — Распределение координаты i агента для $t = 12, 52, 132, 200$

Подбор распределения показал, что получившиеся гистограммы (одинаковые, как для x, так и для y) хорошо приближает Extreme Value Distribution (type I max extreme value distribution, т.е. асимптотическим распределением максимума из набора одинаково распределенных случайных величин) с функцией плотности распределения

$$evd(\alpha(t),\beta(t);x) = \frac{e^{\frac{\alpha(t)-x}{\beta(t)} - e^{\frac{\alpha(t)-x}{\beta(t)}}}}{\beta(t)}.$$

Сами же функции $\alpha(t)$ и $\beta(t)$ при $t > 20$ приближаются линейными

$$\alpha(t) + \gamma\beta(t) = \mathbb{E}x(t) = 0.979616 + 0.199427t, r^2 = 0.999997,$$

где γ – постоянная Эйлера,

$$\alpha(t) = 0.352148 + 0.170925t, r^2 = 0.999965.$$

При этом качество аппроксимации падает исключительно вблизи от границы области, где происходит остановка или начало движения агента. График

зависимости $p-$значения гипотезы от времени приведен на рисунке 2.20, пунктиром обозначено $\tau_{\min}$.

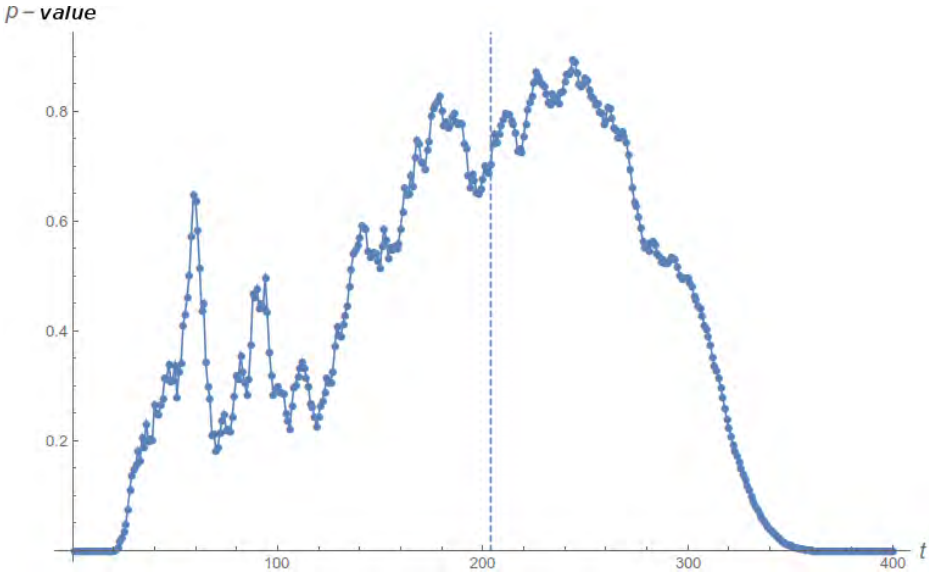


Рисунок 2.19 — Зависимость p-значения для различных времен

Вблизи конечной точки маршрута гистограммы имеют вид, указанный на рисунке 2.20.

Можно предположить, что искомая функция плотности распределения имеет вид

$$rd(t,x) = \Theta(98 - x)evd(\alpha(t),\beta(t); x) + \varphi(t,x),$$

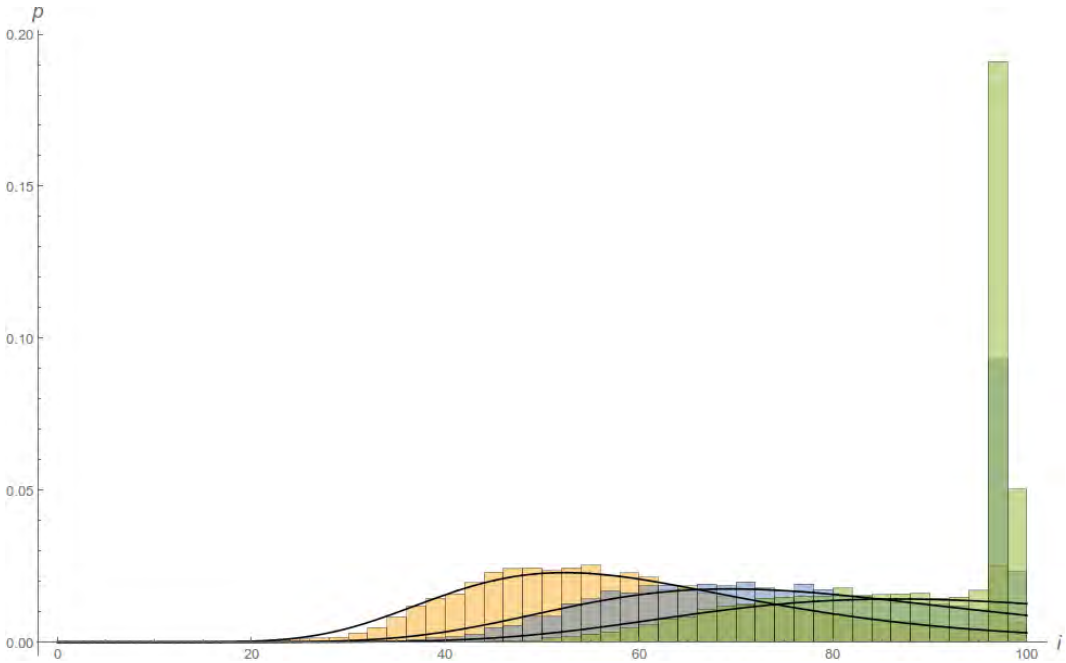


Рисунок 2.20 — Вид гистограмм вблизи конца маршрута, $t = 304, 404, 504$

$$\Theta(x) = \begin{cases} 0, & x < 0, \\ 1, & x \geqslant 0. \end{cases}$$

$\varphi(t,x) = 0$, $x < 94$, $t < \tau_{\min}$. Поведение φ в соответствии с данными вычислительного эксперимента показано на рисунке 2.21, пунктирные линии слева направо соответствуют $\tau_{\min}$ и $\overline{\tau}$.

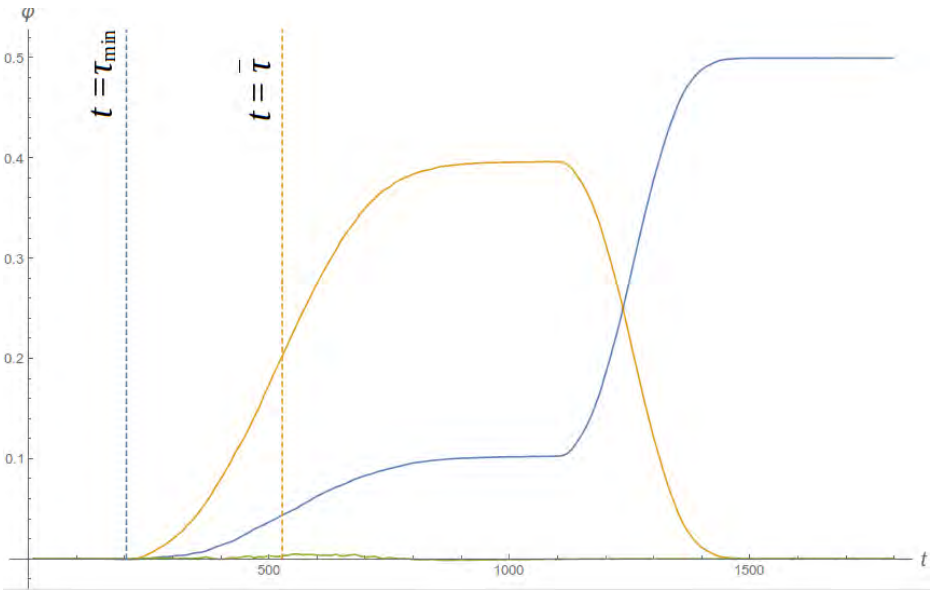


Рисунок 2.21 — Поведение функции $\varphi(t,x)$, $x = 94{,}96{,}98$

Очевидно, что функция φ должна отвечать следующим условиям:

$$1 = \int_0^{\infty} rd(t,x)\mathrm{d}x = \int_0^{98} evd(\alpha(t),\beta(t);x)\mathrm{d}x + \int_0^{98} \varphi(t,x)\mathrm{d}x.$$

$$\varphi(x,0) = 0,$$

$$\lim_{t\to\infty} \varphi(x,t) = \begin{cases} 1/98, & x = 98, \\ 0, & x \neq 98. \end{cases}$$

Из раздела 1 следует, что агентом в клетке (i_0,j_0) выбирается упорядоченный набор $\{u_{ij}^k|(i,j) = \mathbf{r}_h[k]\}$, такой что

$$T_h(\mathbf{r}_h) = \min_{\mathbf{r}_h \in \mathcal{Y}_h \cap U_o(i_0,j_0)} \sum_{(i,j)\in\mathbf{r}_h} u_{ij}^k \|\mathbf{d}_{ij}\|.$$

Можно предположить, что распределение $\varrho(t)$ близко к Extreme Value Distribution потому, что агент всегда выбирает для движения набор клеток $\mathbf{r}_h = \{\mathbf{r}_h[1], \ldots, \mathbf{r}_h[m+1]\}$ (или кривую $\mathbf{r}$) чтоб минимизировать $T(\mathbf{r})$.

Обозначим для маршрута $\mathbf{r}_h = \{\mathbf{r}_h[1], \ldots, \mathbf{r}_h[m+1]\}$, что $\chi_m = m\kappa$, тогда в силу (2.17)

$$\|\mathbf{r}_h(t+\chi_m) - \mathbf{r}_h(t)\| \leqslant \sum_{k=1}^{m} \|\mathbf{r}_h(t+\chi_k) - \mathbf{r}_h(t+\chi_{k-1})\| =$$
$$= \sum_{k=1}^{m} \|\mathbf{d}_k\| \left\lfloor \frac{\kappa}{\kappa\|\mathbf{d}(t)\|\mathrm{u}(t,\mathbf{r}_h(t))} \right\rfloor = \xi(t+\chi_m) = \sum_{(i,j)\in\mathbf{r}_h} \|\mathbf{d}_k\| \left\lfloor \frac{1}{\|\mathbf{d}_k\| u_{ij}^k} \right\rfloor \to \max. \tag{2.44}$$

Таким образом, на каждый такт агент выбирает такое направление движения, что

$$\xi(t+\chi_m) = \max_{r_h \in \mathcal{Y}_h \cap U_o(i_0,j_0)} \sum_{(i,j)\in r_h} \|\mathbf{d}_k\| \left\lfloor \frac{1}{\|\mathbf{d}_k\| u_{ij}^k} \right\rfloor,$$

и, следовательно, имеет распределение близкое к Extreme Value distribution (тип I, максимум). Из экспериментальных данных можно предположить, что оценка (2.44) является довольно точной и

$$\|\mathbf{r}_h(t+\chi_m) - \mathbf{A}\| = \|\mathbf{r}_h(t+\chi_m) - \mathbf{r}_h(0)\| \approx \xi(t+\chi_m).$$

2.7 Модель типа уравнения конвекции-диффузии для движения агентов

В этом подразделе мы будем генерировать «естественные» прямоугольные ландшафты $n \times m$ с клетками l различных классов и N_{obst} препятствиями радиуса o.

Изучим зависимость среднего числа агентов от времени и положения на ландшафте. Мы генерируем 150 случайных ландшафтов с заданной энтропией по алгоритму подраздела 2.2.4. Группа, образованная из $u_0 = 48$ агентов, перемещается с одной стороны квадратного ландшафта в противоположную. Вычислим зависимость среднего (по всем генерируемым ландшафтам) числа агентов u на x-й строке ландшафта от дискретного времени t, $x = \overline{1, x_{\ max}}$, $x_{\max} = 48$. Таким образом, мы обнаруживаем, что эта зависимость определяется, главным образом, не видом конкретного ландшафта, а конфигурационной энтропией ландшафта S. Мы искали зависимость в виде

$$u(x,t) = \frac{u_0}{2}(\operatorname{erf}(\xi_1(S;x,t)) - \operatorname{erf}(\xi_2(S;x,t))), \tag{2.45}$$

where

$$\xi_1(S;x,t) = \frac{a(S;x)}{\sqrt{t}} + b_1(S;x),$$
$$\xi_2(S;x,t) = \frac{a(S;x)}{\sqrt{t}} + \operatorname{sgn}(48-x)b_2(S;x),$$

$$\operatorname{sgn}(x) = \begin{cases} 1, & x > 0, \\ -1, & x \leqslant 0. \end{cases}$$

Такая форма параметров ξ_1 and ξ_2 была взята по аналогии с задачей распространения тепла по стержню с температуропроводностью a, нагретом на его отрезке $[l_1,l_2]$ до температуры u_0. Эта задача (см., например, [111]) имеет решение

$$u(x,t) = \frac{u_0}{2}\left(\operatorname{erf}\left(\frac{x-l_1}{2a\sqrt{t}}\right) - \operatorname{erf}\left(\frac{x-l_2}{2a\sqrt{t}}\right)\right).$$

Предположим, что

$$a(S;x) = a_1(S)x + a_2(S),$$

$$b_1(S;x) = b_{11}(S)\sqrt{x} + b_{12}(S) + \frac{b_{13}(S)}{x^{3/2}},$$
$$b_2(S;x) = b_{11}(S)\sqrt{x} + b_{22}(S).$$

Эти функции позволяю аппроксимировать $u = u(x,t)$ с коэффициентом детерминации $r^2 > 0.97$, средней абсолютной ошибкой $MAE < 0.1251$ и медианной абсолютной ошибкой $MedAE < 0.04366$ для каждого значения конфигурационной энтропии S. Экспериментальные и аппроксимированные значения u показаны на рис. 2.22. Например, когда $N_{obst} = 20$ ($S = 1.67909$)

$$u(x,t) = 24\left(\operatorname{erf}\left(\frac{-2.04693x - 1.03949}{\sqrt{t}} + \frac{4.66438}{x^{3/2}} + 1.08597\sqrt{x} + 0.926477\right) - \right.$$
$$\left. - \operatorname{erf}\left(\frac{-2.04693x - 1.03949}{\sqrt{t}} + 1.08597\sqrt{x} + \operatorname{sgn}(48-x)0.845118\right)\right).$$

Далее мы попытаемся найти задачу для уравнения в частных производных, которая может иметь решение в виде (2.45). Естественно предполагать, что это уравнение

$$\frac{\partial u}{\partial t} = C_1\frac{\partial^2 u}{\partial x^2} + C_2\frac{\partial u}{\partial x},$$

начальные условия – это

$$u(x,0) = u_0\delta(x-1),$$

$$\delta(x) = \begin{cases} 1, & x = 0, \\ 0, & x \neq 0, \end{cases}$$

а асимптотическое условие записывается как

$$\lim_{t\to\infty} u(x,t) = u_0\theta(x - x_{\max}),$$

$$\theta(x) = \begin{cases} 1, & x \geqslant 0, \\ 0, & x < 0. \end{cases}$$

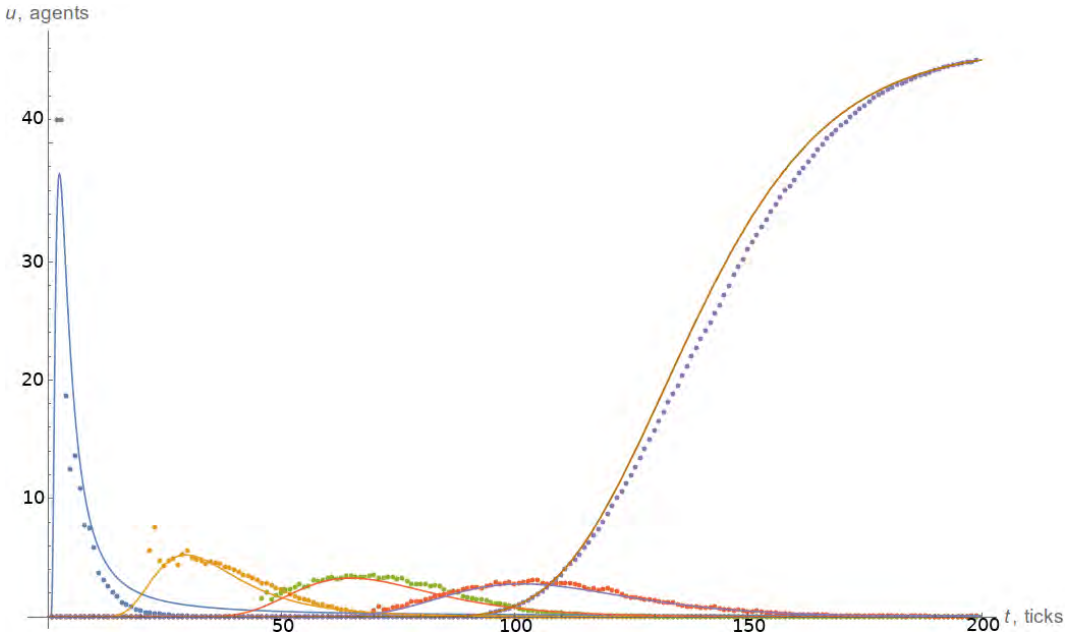


Рисунок 2.22 — Зависимость среднего количества агентов u от дискретного времени t, $N_{obst} = 20$, $x = 2$, $x = 12$, $x = 24$, $x = 36$, $x = 48$ слева направо

Обозначим

$$U_1(x,t) = \mathrm{e}^{-\left(\frac{a(S;x)}{\sqrt{t}} + b_{22}(S) + b_{11}(S)\sqrt{x}\right)^2},$$

$$U_2(x,t) = \mathrm{e}^{-\left(\frac{a(S;x)}{\sqrt{t}} + b_{11}(S)\sqrt{x} + b_{12}(S) + \frac{b_{13}(S)}{x^{3/2}}\right)^2}.$$

Получим из (2.45) что когда $x \in (1, x_{\max})$

$$\frac{\partial u}{\partial t}(x,t) = \frac{u_0(U_1(x,t) - U_2(x,t))(a_1(S)x + a_2(S))}{2\sqrt{\pi t^3}}, \tag{2.46}$$

$$\frac{\partial u}{\partial x}(x,t) = -\frac{u_0\Big(x^2(U_1(x,t)-U_2(x,t))\Big(2a_1(S)\sqrt{x}+b_{11}(S)\sqrt{t}\Big)+3b_{13}(S)\sqrt{t}U_2(x,t)\Big)}{2\sqrt{\pi t x^5}}, \tag{2.47}$$

$$\begin{aligned}
\frac{\partial^2 u}{\partial x^2}(x,t) = \frac{u_0}{4\sqrt{\pi}t^{3/2}x^{13/2}}\Bigg(b_{13}(S)\sqrt{t}U_2(x,t)x^3\Bigg(4\Bigg(4a_1(S)^2x^2+\\
+6a_1(S)a_2(S)x+7a_1(S)b_{11}(S)\sqrt{tx^3}+3a_2(S)b_{11}(S)\sqrt{tx}\Bigg)+\\
+24a_1(S)b_{12}(S)\sqrt{t}x+t\Bigg(10b_{11}(S)^2x+12b_{11}(S)b_{12}(S)\sqrt{x}+15\Bigg)\Bigg)+\\
+8a_1(S)^2\sqrt{tx^{13}}(b_{22}(S)U_1(x,t)-b_{12}(S)U_2(x,t))-\\
-6b_{13}(S)^2tU_2(x,t)x\Bigg(-a_1(S)x^{3/2}+3a_2(S)\sqrt{x}+b_{11}(S)\sqrt{t}x+\\
+3b_{12}(S)\sqrt{tx}\Bigg)+8a_1(S)b_{11}(S)tx^6(b_{22}(S)U_1(x,t)-b_{12}(S)U_2(x,t))+\\
+10a_1(S)b_{11}(S)^2tU_1(x,t)x^{13/2}-10a_1(S)b_{11}(S)^2tU_2(x,t)x^{13/2}+\\
+2b_{22}(S)b_{11}(S)^2t^{3/2}U_1(x,t)x^{11/2}-2b_{11}(S)^2b_{12}(S)t^{3/2}U_2(x,t)x^{11/2}-\\
-18b_{13}(S)^3t^{3/2}U_2(x,t)+(U_1(x,t)-U_2(x,t))\Bigg(8a_1(S)^3x^{15/2}+8a_1(S)^2a_2(S)x^{13/2}+\\
+8a_1(S)b_{11}(S)\sqrt{t}x^6(2a_1(S)x+a_2(S))+\\
+2a_2(S)b_{11}(S)^2tx^{11/2}+2b_{11}(S)^3t^{3/2}x^6+b_{11}(S)t^{3/2}x^5\Bigg).
\end{aligned} \tag{2.48}$$

Решим (2.47), (2.48) относительно U_1, U_2. Подставим в (2.46) найденные значения и получим

$$C_1(x,t) = \frac{P_1(\sqrt{x},\sqrt{t})}{Q_1(\sqrt{x},\sqrt{t})}, \quad C_2(x,t) = \frac{P_2(\sqrt{x},\sqrt{t})}{Q_2(\sqrt{x},\sqrt{t})},$$

где

$$
\begin{aligned}
Q_1(\sqrt{x},\sqrt{t}) = -3b_{11}(S)tx^2\Bigg(4a_1(S)^2x^{13/2}(b_{12}(S)-b_{22}(S))- \\
-2b_{13}(S)x^3\Big(2a_1(S)x(a_1(S)x+a_2(S))-2a_1(S)\sqrt{t}x(b_{22}(S)-2b_{12}(S))+t\Big)+ \\
+b_{13}(S)^2\Big(a_1(S)\sqrt{tx^5}+3a_2(S)\sqrt{tx^3}+3b_{12}(S)tx^{3/2}\Big)+ \\
+3b_{13}(S)^3t\Bigg)+a_1(S)\Bigg(8a_1(S)^2\sqrt{t}x^9(b_{22}(S)-b_{12}(S))+ \\
+b_{13}(S)\Bigg(4a_1(S)\Big(a_1(S)\sqrt{tx^{15}}+3a_2(S)\sqrt{tx^{13}}\Big)- \\
-12a_1(S)tx^{13/2}(b_{22}(S)-2b_{12}(S))+15t^{3/2}x^{11/2}\Bigg)- \\
-6b_{13}(S)^2tx^4\Big(-a_1(S)x+3a_2(S)+3b_{12}(S)\sqrt{t}\Big)- \\
-18b_{13}(S)^3t^{3/2}x^{5/2}\Bigg)+3b_{11}(S)^2t^{3/2}x^4\Bigg(b_{13}(S)\Big(3a_1(S)x^{5/2}+a_2(S)x^{3/2}- \\
-(b_{22}(S)-2b_{12}(S))\sqrt{tx^3}\Big)+2a_1(S)x^4(b_{22}(S)-b_{12}(S))- \\
-b_{13}(S)^2\sqrt{t}\Bigg)+b_{11}(S)^3t^2x^6\Big(x^{3/2}(b_{22}(S)-b_{12}(S))+2b_{13}(S)\Big),
\end{aligned}
$$

$$
\begin{aligned}
Q_2(\sqrt{x},\sqrt{t}) = 2\sqrt{t}\Bigg(&-3b_{11}(S)tx^2\Bigg(4a_1(S)^2x^{13/2}(b_{12}(S)-b_{22}(S))-\\
&-2b_{13}(S)x^3\Bigg(2a_1(S)x(a_1(S)x+a_2(S))-\\
&-2a_1(S)\sqrt{t}x(b_{22}(S)-2b_{12}(S))+t\Bigg)+b_{13}(S)^2\Bigg(a_1(S)\sqrt{tx^5}+\\
&+3a_2(S)\sqrt{tx^3}+3b_{12}(S)tx^{3/2}\Bigg)+3b_{13}(S)^3t\Bigg)+\\
&+a_1(S)\Bigg(8a_1(S)^2\sqrt{t}x^9(b_{22}(S)-b_{12}(S))+\\
&+b_{13}(S)\Bigg(4a_1(S)\Bigg(a_1(S)\sqrt{tx^{15}}+3a_2(S)\sqrt{tx^{13}}\Bigg)-\\
&-12a_1(S)tx^{13/2}(b_{22}(S)-2b_{12}(S))+15t^{3/2}x^{11/2}\Bigg)-\\
&-6b_{13}(S)^2tx^4\Bigg(-a_1(S)x+3a_2(S)+3b_{12}(S)\sqrt{t}\Bigg)-18b_{13}(S)^3t^{3/2}x^{5/2}\Bigg)+\\
&+3b_{11}(S)^2t^{3/2}x^4\Bigg(b_{13}(S)\Bigg(3a_1(S)x^{5/2}+a_2(S)x^{3/2}-(b_{22}(S)-\\
&-2b_{12}(S))\sqrt{tx^3}\Bigg)+2a_1(S)x^4(b_{22}(S)-b_{12}(S))-b_{13}(S)^2\sqrt{t}\Bigg)+\\
&+b_{11}(S)^3t^2x^6\Bigg(x^{3/2}(b_{22}(S)-b_{12}(S))+2b_{13}(S)\Bigg)\Bigg),
\end{aligned}
$$

$$
P_1(\sqrt{x},\sqrt{t}) = -3b_{13}(S)\sqrt{tx^{13}}(a_1(S)x+a_2(S)),
$$

$$
\begin{aligned}
P_2(\sqrt{x},\sqrt{t}) = x^{5/2}(a_1(S)x+a_2(S))\Bigg(&-b_{13}(S)\Bigg(4\Bigg(4a_1(S)^2x^5+\\
&+6a_1(S)a_2(S)x^4+7a_1(S)b_{11}(S)\sqrt{tx^9}+3a_2(S)b_{11}(S)\sqrt{tx^7}\Bigg)+\\
&+24a_1(S)b_{12}(S)\sqrt{t}x^4+tx^3\Bigg(10b_{11}(S)^2x+12b_{11}(S)b_{12}(S)\sqrt{x}+\\
&+15\Bigg)\Bigg)+6b_{13}(S)^2\Bigg(-a_1(S)\sqrt{tx^5}+3a_2(S)\sqrt{tx^3}+b_{11}(S)tx^2+\\
&+3b_{12}(S)tx^{3/2}\Bigg)-\\
&-2x^{11/2}(b_{22}(S)-b_{12}(S))\Bigg(2a_1(S)\sqrt{x}+b_{11}(S)\sqrt{t}\Bigg)^2+18b_{13}(S)^3t\Bigg).
\end{aligned}
$$

Форма коэффициентов $a_1(S)$, $a_2(S)$, $b_{11}(S)$, $b_{12}(S)$, $b_{13}(S)$, $b_{22}(S)$ зависит от метода генерации ландшафтов. Если ландшафты генерируются так, как указывается в начале подраздела, то

$$a(S) = \alpha(S)a(20),$$
$$b_1(S;x) = \beta_1(S)b_1(20;x),$$
$$b_2(S;x) = \beta_2(S)b_2(20;x),$$

где

$$\alpha(S) = 0.000344002\frac{N_{obst}(S)}{10} + 1.00018, \quad MAE < 0.001,$$
$$\beta_1(S) = \beta_2(S) = 1.0582 - 0.0581965\sqrt{\frac{N_{obst}(S)}{10} - 1}, \quad MAE < 0.005.$$

Резултаты такой аппроксимации показаны на рис. 2.23, $x = x_{max} = 48$.

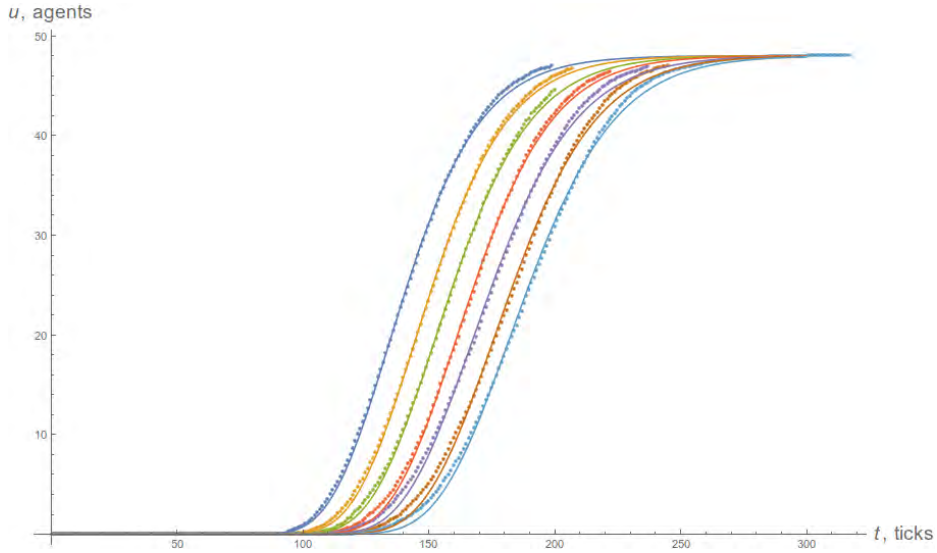


Рисунок 2.23 — Зависимость u от энтропии S, слева направо $N_{obst} = 20$, $N_{obst} = 40$, $N_{obst} = 60$, $N_{obst} = 80$, $N_{obst} = 100$, $N_{obst} = 120$, $N_{obst} = 140$

Следует обратить внимание, что мы получаем разные результаты по различным процедурам генерации ландшафта. Общий вид функции u будет по-прежнему описываться (2.45), но точные значения параметров могут быть совершенно различными. Например, можно разместить квадраты разных классов на равномерно по всему ландшафту (рис. 2.6c). Сравнение функций u для равномерно случайного ландшафта (справа) и «естественного» ландшафта (слева), приведено на рис. 2.24.

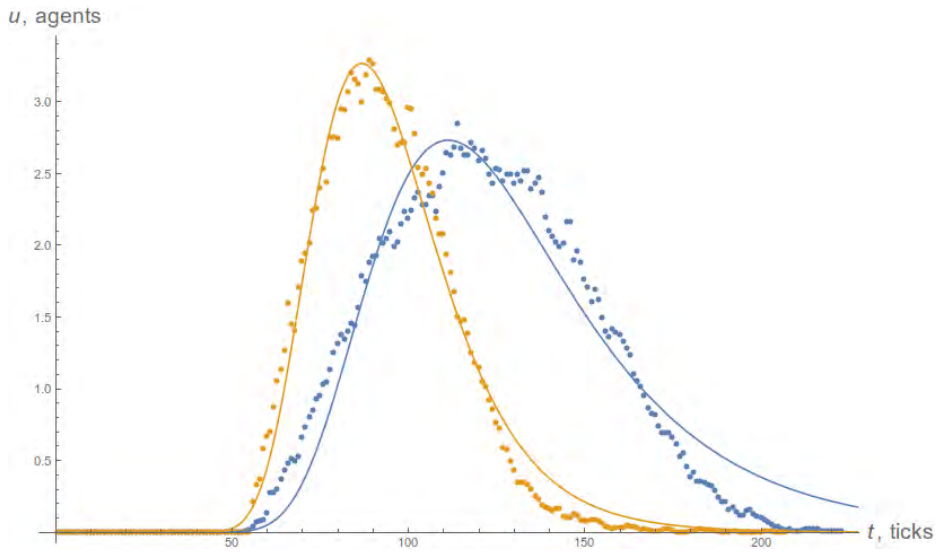


Рисунок 2.24 — Функции u для ландшафтов, сгенерированных разными способами, $x = 26$, $S = 1.84877$

2.8 Методы квзиоптимального распределения авионики на основе клеточного автомата и муравьиного алгоритма

2.8.1 Постановка задачи

Модульная система организации аппаратуры на борту самолета называется интегрированной модульной авионикой (IMA). Устройства IMA, именуемые модулями, параллельно выполняют несколько функций бортовых систем с использованием стандартизированного программного обеспечения. В системе IMA периферийные устройства соединены с единой системой интерфейсов ввода-вывода. Все модули IMA соединены единой широкополосной сетью с высокой пропускной способностью. Параллельное выполнение нескольких функций воздушного судна в одном модуле позволяет сократить количество авиационной радиоэлектронной аппаратуры и вместе с этим массу и стоимость системы авионики, а также – расходы на разработку, производство и техническое обслуживание.

Сегодня системы аппаратуры IMA внедрены на всех современных гражданских и военных самолетах, самолетах бизнес-класса и вертолетах. В самолетах Boeing 787 и A350 реализованы системы IMA второго поколения, так называемая распределенная интегрированная модульная авионика (DIMA).

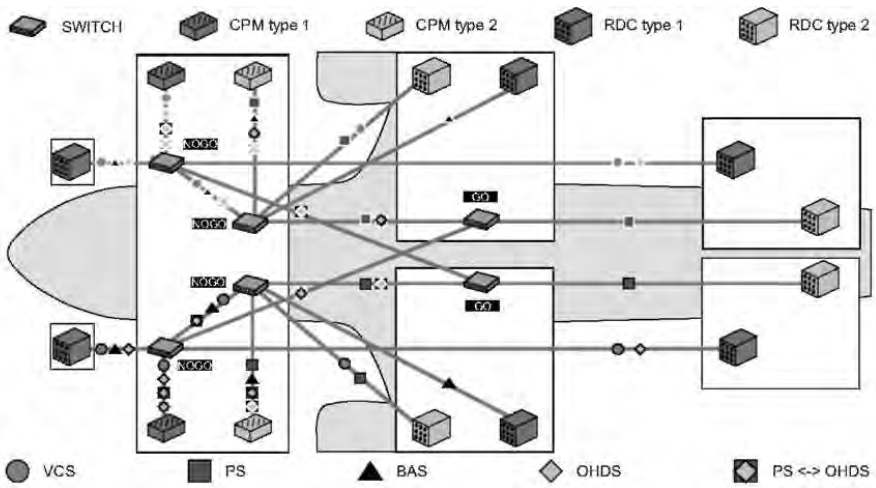


Рисунок 2.25 — Пример назначения кабельных трасс и мест установки

В настоящем разделе предлагается использование клеточного автомата в виде сетки с движущимися по ней агентами для решения двух задач DIMA: пространственное распределение устройств и присваивания периферийных проводов (см. рис. 2.25, приведен по книге [112]). В роли подвижных агентов будут выступать сами устройства, пытающиеся найти для себя наиболее «удобное» место на самолете. Несмотря на кажущееся отличие задачи от ранее изложенных, она может быть решена в рамках все той же модели движения и с помощью того же программного обеспечения.

Программы присваивания устройств вычисляют пространственное распределение устройств DIMA в самолете. Устройства присваиваются местам установки, ограниченным по объему. Кроме того, учитываются ограничения по пространственному разделению.

Предлагается алгоритм квазиоптимального распределения устройств по местам установки в пространстве с использованием модифицированного социального потенциала, аналогичного использованному в [74] для моделирования роя агентов. Сама задача оптимизации решается в рамках клеточно-автоматного формализма, идея какого применительно к задачам распределения ресурсов была почерпнута в работе [113].

Периферийные провода соединяют периферийные устройства с устройствами DIMA. Изложенная задача решается в условиях ограничения на количество проводов и на соединения между проводами в кабельных трассах. Результатом присваивания периферийных проводов является трасса каждого периферийного провода. Она сопровождается перечнем всех кабельных трасс,

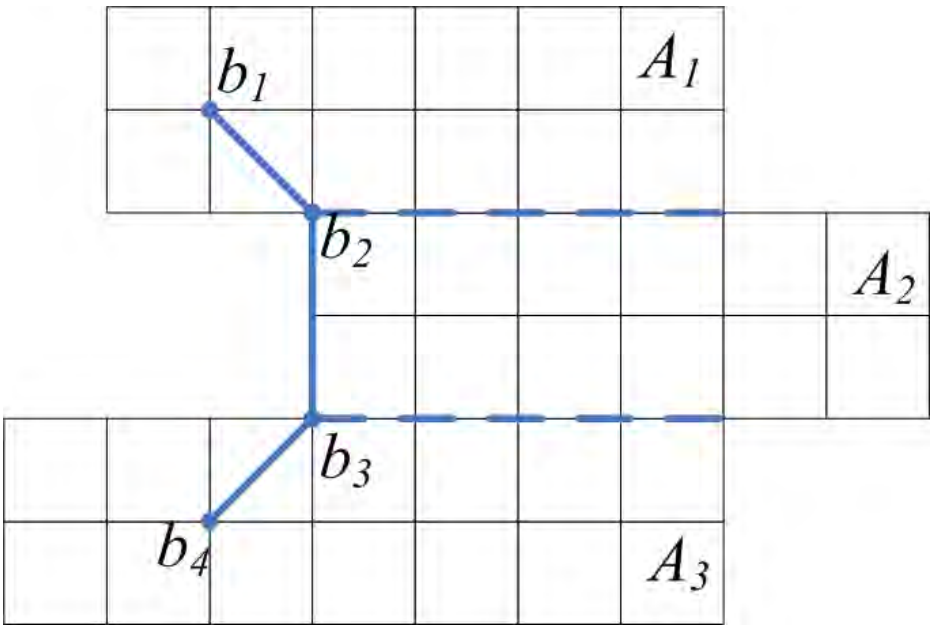


Рисунок 2.26 — Разбиение U на выпуклые подмножества

соединений и мест установки, которые она пересекает. Набор всех трасс не должен превышать количество ресурсов в каждой из кабельных трасс, соединений или мест установки, при этом никакие две трассы не должны нарушать ограничение по разделению. Кроме того, в кабельных трассах и местах установки не должно размещаться неограниченное количество проводов, при этом оно должно ограничиваться диаметром или точками крепления.

В разделе описывается алгоритм поиска квазиоптимального назначения кабельных трасс с помощью алгоритма муравьиной колонии поиска оптимального пути на графе типа описанного в [114]. Отличительной особенностью предлагаемого алгоритма является параллельное сосуществование нескольких муравьиных колоний.

2.8.2 Задача минимизации длины кабеля

Пусть задана равномерная решетка в $\mathbb{Z}^3$ и подмножество $U \subset \mathbb{Z}^3$. Имеется множество средств $V = \{v_i | i = \overline{1,n}\}$, которые нужно разместить по ячейкам U. Для каждого средства $v \in V$ задан список средств $C(v) \subset V$, которые должны быть подсоединены непосредственно к v. Необходимо разместить средства по решетке так, чтоб расстояния между всеми подсоединенными средствами было минимальным.

Поскольку $U \in \mathbb{Z}^3$ является, вообще говоря, невыпуклым и даже несвязным, определим расстояние ρ на U как геодезическое расстояние, т.е. расстояние между клетками x и y равно длине минимального решеточного пути между ними, а если такого пути нет, то расстояние будем считать бесконечным. Поскольку такое расстояние трудно вычисляется (требуется использовать, например, алгоритм Дейкстры или эвристические алгоритмы поиска кратчайшего пути типа A^*), то имеет смысл заранее выделить из U пересекающиеся лишь границами подобласти U_i^R, в которых U является замощением некоторого выпуклого связного подмножества $\mathbb{R}^3$ и полагать

$$\forall(x,y \in U_i^R) \quad \rho(x,y) = \| x - y \|_{\mathbb{R}^3} ,$$

а геодезическим расстоянием пользоваться лишь в том случае, когда $x \in U_i^R$ и $y \in U_j^R$, $i \neq j$. Можно пойти еще дальше, и приближать конечное геодезическое расстояние между клетками $y_0 \in U_0^R$ и $y_m \in U_m^R$, причем кратчайший путь $L_0 \dots L_{m-1}$ от y_0 к y_m таков что $L_i \in U_i^R$, $i = \overline{1,m-1}$, как

$$\rho(x,y) = \sum_{i=0}^{m-1} \| y_i - y_{i+1} \|_{\mathbb{R}^3} ,$$

где y_{i+1} – ближайшая к y_i клетка на пересечении границ U_i^R и U_{i+1}^R (рис. 2.26).

Альтернативно, если множество U не меняется со временем, то возможно заранее, до процесса поиска оптимального размещения, составить таблицу геодезических расстояний между каждой парой клеток из U.

Введем понятие притягивающе-отталкивающего (социального) потенциала. Пусть для каждого средства $v_k \in V$ задана функция $\varphi_k : U \to \mathbb{R}_{\geqslant 0}$ и также задана функция $\varphi_0 : U \to \mathbb{R}_{\geqslant 0}$. Эти функции отвечают следующим условиям ($C_k \geqslant 0$, $\alpha_l > 0$)

1. Пусть $d_0(i,j,l)$ – это расстояние от клетки $(i,j,l) \in U$ до границы ∂U (будем считать, что ∂U не содержится в U), тогда $\varphi_0(i,j,l)$ монотонно возрастает при убывании $d_B(i,j,l)$. Для простоты будем считать, что

$$\varphi_0(i,j,l) = -\frac{C_0}{d_0(i,j,l)^{\alpha_0}}.$$

2. Пусть в клетке (i_1,j_1,l_1) находится средство v_k. Пусть $d(i_1,j_1,l_1;i_2,j_2,l_2) = d_k(i_2,j_2,l_2)$ – расстояние между клетками (i_1,j_1,l_1) и (i_2,j_2,l_2). Тогда $\varphi_k(i,j,l)$ выражается как

$$\varphi_k(i,j,l) = \frac{C_{k1}}{d_k(i,j,l)^{\alpha_{k1}}} - \frac{C_{k2}}{d_k(i,j,l)^{\alpha_{k2}}}.$$

Будем считать, что на средство $v_s \in V$, находящееся в ячейке (i,j), действует притягивающе-отталкивающий потенциал

$$\Phi_s(i,j,l) = \sum_{k=1}^{n} \delta_s(\varphi_k(i,j,l)),$$

где

$$\delta_s \circ \varphi_k = \begin{cases} 0, s = k, \\ 0, v_s \notin C(v_k), \\ 1, v_s \in C(v_k) \wedge s \neq k. \end{cases}$$

Обозначим как $P(i,j,l)$ средство (точнее, номер средства) в клетке (i,j,l) и будем считать, что $P(i,j,l) = 0$, если средства в данной клетке нет. Обозначим как $\mathcal{D} = \{(i,j,l)|i,j,l = \overline{-1,1}\}$.

Предлагается следующий алгоритм 1.

0. Средства из V произвольно размещаются в U.
1. Каждой клетке $(i,j,l) \in U$ присваивается список $\{\varphi_k(i,j,l)|k = \overline{0,n}\}$
2. Начинается функционирование клеточного автомата со следующей локальной функцией перехода:

2.1. Средство v_s в ячейке (i,j,l) выбирает такое возможное направление смещения $d = (d_1,d_2,d_3) \in \mathcal{D}$, что $\Phi_s(i + d_1, j + d_2, l + d_3) \to max$,

2.2. Если клетка $(i + d_1, j + d_2, l + d_3)$ свободна, то v_s перемещается в эту клетку,

2.3. Если клетка $(i + d_1, j + d_2, l + d_3)$ занята, но находящееся в ней средство планирует переместиться в клетку (i,j,l), то средства меняются клетками,

2.4. Если клетка $(i + d_1, j + d_2, l + d_3)$ занята и находящееся в ней средство не планирует переместиться в клетку (i,j,l), то помечаем d как невозможное, идем к шагу 1.1.

3. Если прошло тактов больше, чем максимально возможное количество или автомат стал стабилен, оканчиваем алгоритм.

4. Возвращаемся к 1.

Начальное размещение средств можно оптимизировать с помощью алгоритма 2.

0. Случайно размещаются n_0 средств из V (предпочтителен выбор средств с наибольшим $|C(v)|$).
1. Каждой клетке $(i,j,l) \in U$ присваивается список $\{\varphi_k(i,j,l)|k = \overline{0,n}\}$
2. Случайно выбирается средство $v_k \in V$ и размещается в окрестности одного из уже размещенных средств в клетке (p,q,r) так, чтоб $\Phi_k(p,q,r) \to max$.
3. Если в V еще остались неразмещенные средства, переходим к 1, иначе к 4.
4. Конец.

Вместо направлений $d \in \mathcal{D}$ и окрестностей лишь из смежных клеток, можно рассматривать пути из m направлений и окрестности из m слоев клеток.

2.8.3 Размещение устройств по местам установки

Пусть для каждой клетки U и для каждого устройства из V задана вероятность появления устройства в данной клетке $P : U \times V \to [0,1]$. Эту функцию возможно добавить к алгоритму предыдущего раздела и сформулировать п. 2.1. алгоритма 1 следующим образом

Средство v_s в ячейке (i,j,l) сортирует направления движения $d = (d_1,d_2,d_3) \in \mathcal{D}$, присваивая им веса $P(i+d_1,j+d_2,l+d_3)\Phi_s(i+d_1,j+d_2,l+d_3)$ и выбирает направление с максимальным весом. Альтернативно, можно случайно взвешенно перемешать направления и выбрать первое из них.

2.8.4 Назначение кабельных трасс

Как было упомянуто выше, для поиска оптимальной кабельной трассы будет использован алгоритм типа муравьиной колонии. Пусть $V' \subseteq V$ множество

устройств, которые должны быть соединены кабелями, и устройств-концентраторов. Также пусть задана матрица A для устройств из V', которые должны быть соединены, $a_{ij} = 1$, если устройства v_i, v_j должны быть соединены и $a_{ij} = 0$ в противном случае. Также задана матрица B, $b_{ij} = 1$, если устройства v_i, v_j должны быть аппаратно разделены и $b_{ij} = 0$ в противном случае. Для каждого средства $v \in V$ задан список коммутационного оборудования $L(v)$, к которому оно может быть подключено.

Определим граф $\Gamma = (V,E,\psi)$, где E – множество соединений между средствами, $\psi : V \to U$ – функция соответствия средства и места установки, полученного на предыдущем этапе, и его подграф $\Gamma' = (V'',E'',\psi)$, вершинами которого являются средства из V' и коммутационное оборудование. Каждый муравей $m \in K_i$ в каждый момент t имеет список $M_i(m,t)$ уже посещенных узлов Γ', в которые он больше не возвращается. Муравьи из K_i имеют «зрение», в том смысле, что стремятся посетить вершины v_j, такие что $a_{ij} = 1$. Множество целевых вершин $m \in K_i$ обозначим как $T(t,v_i) = L(v_i) \backslash M_i(m,t)$

Каждому $v_i \in V'$ соответствует муравьиная колония K_i, которая помечает ребра Γ' своим собственным феромонам. Колонии могут «враждовать» друг с другом, так что феромон одной колонии является репеллентом для «враждебной» им, что нужно для разделения маршрутов. Для ограничения количества подключений к одному узлу примем, что феромон в концентрации выше определенной начинает действовать как репеллент. Таким образом, каждому ребру (r,s) между вершинами v_r и v_s соответствует вектор значений феромона $\tau_{rs}(t) = (\tau^1_{rs}(t), \ldots, \tau^{|V'|}_{rs}(t))$, $\tau^j_{rs}(t)$ – уровень феромона колонии K_i в момент времени t.

Определим функцию восприятия феромона муравьем из k-й колонии

$$\theta_{rs}(k,t) = \begin{cases} \sum_{i=1}^{|V'|} \tau^i_{rs}(t), \forall(i) & b_{ik} = 0, 0 < \sum_{i=1}^{|V'|} \tau^i_{rs}(t) \leqslant m_{rs}, \\ \xi, & \sum_{i=1}^{|V'|} \tau^i_{rs}(t) = 0, \\ & 0, \, otherwise. \end{cases}$$

Здесь ξ – небольшое случайное число, $m_{rs} > 0$ – предельное количество подключений через ребро (r,s).

Подытожим сказанное:

1. Муравей $m \in K_j$, переходя по ребру (r,s) в момент времени t, помечает его феромоном своей колонии K_j:

$$\tau_{rs}^{j} = \tau_{rs}^{j} + \frac{Q}{\Lambda(t,m)},$$

где $\Lambda(t,m)$ – длина пройденного пути в момент времени t муравьем m, Q – параметр системы порядка длины пути.

2. Муравьи, попав в одну из целевых вершин, помечают ее. Если оптимальный маршрут до этой вершины считается найденным, то она исключается из списка целевых для данной колонии.

3. Обозначим множество всех смежных с v_r вершин как $O(v_r)$. Вероятность перехода муравья $m \in K_i$ из вершины $v_r \in V''$ в соседнюю вершину $v_s \in V''$ определяется как

$$P_{rs}(m,t) = \min_{v_j \in T(t,v_i)} \frac{\theta_{rs}(i,t)^{\alpha}(\rho(v_s,v_j))^{-\beta}}{\sum_{l:v_l \in O(v_r)\setminus M_i(m,t)} \theta_{rl}(i,t)^{\alpha}(\rho(v_s,v_l))^{-\beta}}.$$

4. С каждым тактом функционирования системы количество феромона колонии K_k испаряется по правилу

$$\tau_{ij}^{k} = (1-p)\tau_{ij}^{k}(t) + \sum_{m \in K_k} \frac{Q}{\Lambda(t,m)},$$

где $p \geqslant 0$ – скорость испарения феромона.

Глава 3. Модели и методы построения самоорганизующейся системы связи мобильных агентов

3.1 Общая схема модели системы связи

В разделе 2.3 автором были предложены модель движения иерархически организованных агентов и модель конфликта таких агентов. При описании модели конфликта агентов была сконструирована упрощенная система связи, позволяющая агентам обмениваться сигналами о наступлении или отступлении. Данная глава посвящена моделированию системы связи, подобной тем, что используются в тактическом звене управления воинским формированием у нас в стране и за рубежом. Одной отличительной чертой моделируемой системы связи является то, что канал связи между двумя агентами может состоять из нескольких каналов, организованных совершенно различно как на физическом уровне, так и на уровне протоколов. Другой чертой является неустойчивость системы военной связи (которая относится, в целом, к классу MANET [115], хотя и может иметь некую стационарную компоненту). В один момент времени между агентами, которые, как правило, оснащены как высокоскоростными средствами с протоколами типа WiMAX или Wi-Fi, так и низкоскоростными узкополосными радиостанциями диапазона МВ или ДКМВ с нестандартизированными протоколами передачи данных, может быть организован высокоскоростной канал, а в другой – только низкоскоростной, а в некоторые промежутки времени невозможно организовать никакого канала или возможно организовать только составной канал. По очевидным причинам, в открытой печати не существует подробного описания систем военной связи, однако функции и архитектуру таких систем можно понять из многочисленных популярных статей, например [116].

Вышесказанное делает довольно затруднительным моделирование канала связи на сетевом или канальном уровне. Более того, в военных условиях интерес представляет не доставка пакета, а именно доставка сообщения в установленный срок, так как ценность доставленной со значительной задержкой информации может оказаться нулевой. Поэтому моделирование будет проводиться на уровне сообщений, которыми обмениваются агенты, с упрощением очень многих деталей.

Предлагаемая модель сети связи состоит из двух частей – из меняющегося во времени в зависимости от местоположения связывающихся агентов графа системы связи, имеющего ребра, соответствующие каналам связи разного вида, и присвоенных вершинам графа системы связи расписаний рассылки сообщений по другим вершинам, причем скорость доставки сообщения зависит от вида канала между узлом-отправителем и узлом-адресатом. Таким образом, задачи, возникающие при исследовании модели, в некоторой степени напоминают динамические задачи теории расписаний, типа задач управления транспортными системами, однако, в отличие от многих задач этого типа, допустимо одновременное обслуживание произвольного количества требований. Классические модели из теории телетрафика не вполне подходят – поступление сообщений в канал не случайно, а обусловлено расписанием и потеря сообщения исключена. С другой стороны, в разных каналах применяются различные протоколы, не всегда документированные, с разными максимальными скоростями передачи и дисциплинами обслуживания пакетов, что затрудняет имитационное моделирование в известных средах типа ns-3 или OMNeT++. Все вышеперечисленное и подтолкнуло автора к созданию предлагаемой феноменологической модели сети военной связи.

3.1.1 Ячейки и состояния

Будем исследовать клеточный автомат, моделирующий движение и конфликт агентов, ранее описанного вида [1]. Предположим, что множество ячеек нашего автомата – это $\mathbb{Z}^2$, а в качестве множества состояний будем рассматривать множество векторов типа

$$\mathscr{E} = F \times W \times Q \times U = \{(f_{ij}, w_{ij}, q_{ij}; u_{ij}) | (i,j) \in \mathbb{Z}^2\},$$

где для всех i, $j \in \mathbb{Z}^2$ $f_{ij} \in F = \{0,1,2\}$, $u_{ij} \in U \subset \mathbb{R}$ – ограниченное множество, $w_{ij} \in W \subseteq \mathbb{Z}^g$ – набор чисел, необходимый для реализации внутренней динамики клеточного автомата,

$$\sum_{i,j \in \mathbb{Z}} f_{ij} < \infty.$$

Значение $f_{ij} = 0$ соответствует отсутствию, $f_{ij} = 1$ или $f_{ij} = 2$ – наличию в клетке (i,j) агента, принадлежащему к первой или второй противоборствующей стороне соответственно. Величина $q_{ij} \in Q \subset \mathbb{Z}^p$ соответствуют некоторому уникальному идентификатору агента, находящегося в клетке с координатами (i,j). Этот идентификатор является вектором целых чисел, так как в предыдущих работах автора рассматривались модели, учитывающие внутреннюю p-уровневую иерархию агентов, компонента с номером $k = \overline{1,p-1}$ вектора q_{ij} соответствует номеру агента (группы агентов) k-го уровня иерархии внутри группы $k+1$-го уровня иерархии, p-я компонента вектора q_{ij} соответствует номеру группы агентов (агента) самого верхнего уровня.

Значения u_{ij} соответствуют некоторой характеристике местности, влияющей на распространение радиоволн – например, высоте середины квадрата с координатами (i,j) над уровнем моря, лесистости, наличию зданий и т.п.

Для целей практической реализации можно считать, что в множестве клеток $\mathbb{Z}^2$ выделен квадрат K из клеток с положительными координатами, не превосходящими длины стороны квадрата L_K, причем для любой клетки (i,j) вне и на границах этого квадрата всегда $f_{ij} = 0$, $w_{ij} = \mathbf{0}$, $q_{ij} = \mathbf{0}$, вне квадрата $u_{ij} = 0$, т.е. клетки вне квадрата K находятся всегда в состоянии покоя.

Поскольку сама модель движения и конфликта была уже рассмотрена в статьях [1; 2], то можно считать, что наборы f_{ij}, w_{ij}, q_{ij} являются исходными данными и уже заданы во всех i, $j \in \mathbb{Z}^2$ в любой момент функционирования автомата.

3.1.2 Граф системы связи

Должен быть задан граф потребностей в связи, указывающий на то, какие вообще агенты должны быть связаны друг с другом. Пусть $Q_f \subseteq Q$ – множество вершин-идентификаторов агентов, принадлежащих к стороне f, $E_f \subseteq Q_f \times Q_f$ – множество ребер, определяющее связность агентов. Определим граф потребностей в связи стороны f как $\Pi_f = (Q_f, E_f)$.

Система связи может изменяться в каждый момент $t \in \mathbb{Z}$, $t = \overline{0,N}$ функционирования клеточного автомата в зависимости от дальности агентов, особенностей местности, применения средств радиоэлектронной борьбы и т.п.

Зададим граф системы связи соответствующим образом, чтоб отразить это:

$$\Gamma_f(t) = (V_f(t), C, \varphi_f(t,\cdot)),$$

где $f = \overline{1,2}$, $V_f(t) = \{(i,j) \in \mathbb{Z}^2 | f_{ij}(t) = f\}$ – множество клеток, в которых в данный момент t находятся агенты стороны f, $C = \{c_1, \ldots, c_l\}$ – множество каналов, $\varphi_f(t,\cdot) : C \to V_f(t) \times V_f(t)$ – функция инцидентности. Каждый элемент множества C может означать, например, канал (Hotspot) некоторой модификации Wi-Fi 802.11b ($\sim$ 2,4 ГГц) в диапазоне ДМВ (300 МГц – 3 ГГц),канал определенной модуляции, например, ЧМ в диапазоне МВ (30 – 300 МГц), канал определенной модуляции, например USB, в диапазоне ДКМВ (3 – 30 МГц). Таким образом, $c \in C$ является числовым идентификатором определенного канала связи.

Будем считать, что все каналы имеют некоторый m-мерный «вес» – абстракцию свойств канала – предельной дальности, необходимости прямой видимости и т.п., определенный функцией $\psi_1 : C \to \mathbb{R}^m$, $\psi_1 = (\psi_1^1, \ldots, \psi_1^m)$. Длина вектора веса m зависит количества выделяемых у канала связи признаков, т.е. от степени желаемой детализации модели системы связи.

Также для каждого канала $c \in C$ задана предельная скорость передачи данных (пропускная способность) $v_{max}(c)$.

Обозначим через $L(i_1,j_1; i_2,j_2)$ множество всех клеток, пересекаемых в более чем одной точке отрезком прямой между точками $(i_1 + 0{,}5, j_1 + 0{,}5)$ и $(i_2 + 0{,}5, j_2 + 0{,}5)$. Будем считать, что задана функция видимости $\psi_2 : 2^{\mathbb{Z}^2} \to \mathbb{R}^m$, определяющая, каковы условия для организации канала, $\psi_2 = (\psi_2^1, \ldots, \psi_2^m)$.

Приведем пример такой функции видимости для $m = 2$. Если величина u_{ij} из вектора состояния ячейки соответствует высоте местности, то уравнение прямой, проходящей через середины клеток (i_1,j_1), (i_2,j_2) на высоте $u_{i_1 j_1}$ и $u_{i_2 j_2}$ над серединами первой и второй клетки соответственно будет (в координатах ι, κ, $\upsilon(\iota, \gamma)$).

$$\frac{\iota - i_1 - 0{,}5}{i_2 - i_1} = \frac{\kappa - j_1 - 0{,}5}{j_2 - j_1} = \frac{\upsilon - u_{i_1,j_1}}{u_{i_2,j_2} - u_{i_1,j_1}}.$$

Соответственно, можно рассчитать, насколько высота клеток посередине отрезка, соединяющего середины клеток (i_1,j_1), (i_2,j_2), превышает высоту в клетках на его концах, вычисляя величину типа

$$\psi_2^1(L(i_1,j_1; i_2,j_2)) = \max_{(\iota,\kappa) \in L(i_1,j_1;i_2,j_2)} u_{\iota,\kappa} - \upsilon(\iota, \kappa). \tag{3.1}$$

Очевидно, что если значение ψ_2^1 будет много больше нуля, то, например, каналы ДМВ между соответствующими клетками организовать будет нельзя, хотя возможно будет, до какого-то значения ψ_2^1, организовать каналы МВ и т.п. Соответствующим образом должны быть назначены и веса для каналов данных типов.

Также можно ограничить каналы некоторой предельной длиной между серединами квадратов (i_1,j_1) и (i_2,j_2) и положить

$$\psi_2^2(L(i_1,j_1;i_2,j_2)) = \sqrt{(i_2-i_1)^2+(j_2-j_1)^2}. \tag{3.2}$$

Описание так сконструированной функции видимости можно пояснить рисунком 3.1. На нем высоты незаштрихованных клеток равны 0, высоты клеток, штрихованных одинарной штриховкой, равны 1. «Вес» канала c определим как $\psi_1(c) = (1,200)$, т.е. канал может быть организован через клетки, «возвышающиеся» над связываемыми не более, чем на 1, и максимальная дальность канала – приблизительно 200 клеток (точная оценка зависит от того, как именно расположены клетки). Как показано на рисунке 3.1, $\psi_2(L(2,3;2,5)) = (\psi_2^1(L(2,3;2,5)),\psi_2^2(L(2,3;2,5))) = (1,2)$.

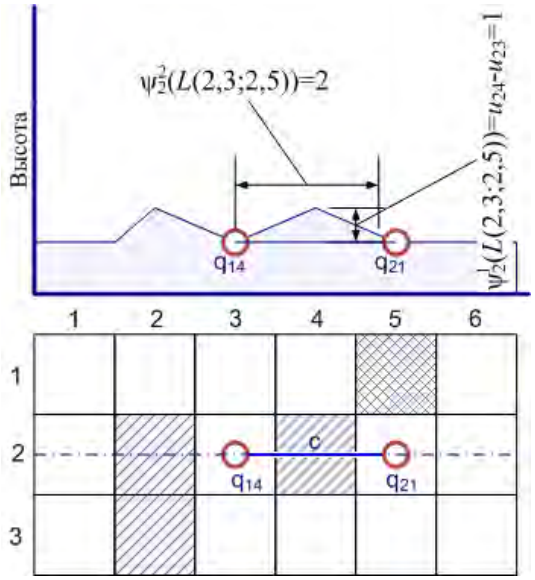


Рисунок 3.1 — Пример функции видимости.

Определим функцию совместимости каналов $com : C \times C \to \{0,1\}$. Каналы c_i, c_j будут совместимыми ($com(c_i,c_j) = 1$), если возможна одновременная работа обоих каналов без помех и несовместимыми ($com(c_i,c_j) = 0$) – в противоположном случае. Если организовано n несовместимых каналов, по которым одновременно осуществляется отправка сообщений, то скорость передачи данных по каждому из них падает в n раз.

Функция инцидентности отвечает следующим условиям. Пусть

1. $\varphi_f(t,c_k) = ((i_1,j_1),(i_2,j_2))$,
2. Клетка (i_1,j_1) имеет состояние $(f, w_{i_1j_1}, q_{i_1j_1}; u_{i_1j_1})$.
3. Клетка (i_2,j_2) имеет состояние $(f, w_{i_2j_2}, q_{i_2j_2}; u_{i_2j_2})$.

Тогда

1. $(q_{i_1j_1},q_{i_2j_2}) \in E_f$. Это значит, что в любой момент времени связаны не произвольные агенты, а лишь те, которые изначально должны быть связаны в соответствии с графом Π_f, что соответствует логике планирования связи.
2. Для всех $i = \overline{1,m}$, где m – длина вектора веса, $\psi_1^i(c_k) \geqslant \psi_2^i(L(i_1,j_1;i_2,j_2))$. Это означает, что вес канала не может быть меньше, чем допускают условия видимости.
3. Максимальная пропускная способность канала c_k – наибольшая из возможных. Это означает, что сначала пытаются назначить каналы с максимальной v_{max}, а если это не получается (каналы данной пропускной способности кончились или назначение канала запрещает функция видимости) – то с меньшей пропускной способностью.

Агенты с идентификаторами из множества Q_f могут являться адресатами или отправителями сообщений друг для друга. Сообщения рассылаются каждым агентом с идентификатором $q \in Q_f$ по определенному расписанию

$$R(q) = \{(t_i, q_i, b_i, \tilde{t}_i, \gamma_i) | i \in \mathbb{Z}\}, \tag{3.3}$$

где $(t_i, q_i, b_i, \tilde{t}_i, \gamma_i) \in R(q)$ имеет следующий смысл: t_i – момент времени, в который рассылается сообщение длиной b_i, которое должно быть доставлено за время $\tilde{t}_i$ адресату с идентификатором q_i, $\gamma_i \in \mathbb{N}$ – важность сообщения, определяющая, какая часть канала достанется на передачу сообщения, если через один и тот же канал одновременно пересылается несколько сообщений. При некоторых состояниях клеточного автомата (соответствующим обнаружению противника, попаданию в окружение и т.п.) сообщения рассылаются внепланово и имеют зависящую от состояния клеток в окрестности агента длину и срок доставки.

Идентификатор агента, находящегося в момент времени в клетке с координатами z в момент времени t, будем обозначать $q(t,z)$. Клетку, в которой в момент времени t находится агент с идентификатором q, будем обозначать $z(t,q)$.

Канал, соединяющий z и z' в момент времени t обозначим как $c(t,z,z') = c(t,q,q')$, $q = q(t,z)$, $q' = q(t,z')$. Если вершины z, z' не смежные в графе $\Gamma_f(t)$ и существует некоторый путь, их соединяющий (кратчайший или иной обусловленный выбранной моделью передачи информации), то $c(t,z,z')$ – множество всех каналов-ребер данного пути.

Количество каналов несовместимых с $c = c(t,q,q')$, по которым происходит передача информации в момент времени t обозначим как $n(t,c)$.

Заметим, что в статье [117] была вкратце описана подобная конструкция коммуникационного графа, не связанная, однако, с клеточным автоматом.

3.1.3 Модель передачи информации

Длину сообщения вместе с накладными расходами на пересылку, отправленного агентом с идентификатором q агенту с идентификатором q' в момент времени t будем обозначать как $b(t,q,q')$. Если $t = t_i$ из расписания $R(q)$ и q' входит в число адресатов в момент t_i, то объем пересылаемого сообщения в «чистом виде» будет, согласно (3.3), b_i, а вместе с накладными расходами станет

$$b(t,q,q') = K_1(t,c(t,q,q'))b_i + K_2, \tag{3.4}$$

иначе $b(t,q,q') = 0$. Здесь $K_1(t,c(t,q,q')) > 0$, $K_2 > 0$ – коэффициенты накладных расходов на пересылку. Далее будем полагать, что обозначения типа b, b' обозначают уже длины сообщений с учетом накладных расходов.

Будем обозначать скорость в канале c в момент времени t как $v(t,c)$. Скорость зависит от пропускной способности канала $v_{max}(c)$ и от некоторых нижеперечисленных случаев. Если по каналу c передается в некоторый интервал времени только одно сообщение, то в этот интервал $v(t,c) = v_{max}(c)$, иначе скорость падает прямо пропорционально количеству n одновременно передаваемых в один интервал сообщений как $v(t,c) = v_{max}(c)/n$ (см. далее).

Если $c(t_0,q_0,q_r)$ содержит в себе некий путь из более чем одного ребра (который обозначим $q_0 \dots q_i \dots q_r$) и передача сообщения длиной b_0 в момент времени t_0 осуществляется именно по этому пути, то в узле $z_i = z(t,q_i)$ может устанавливаться задержка $d_i(t)$ перед отправкой сообщения из транзитного узла далее по маршруту. Будем считать, что сообщение начнет исходить из i-й

вершины маршрута, $i = \overline{1,r-1}$, в момент времени

$$t_i = t_{i-1} + d_i(t_{i-1}) + \frac{b\alpha}{v(t_{i-1},c(t_{i-1},q_{i-1},q_i))},$$

где t_{i-1} – время, в которое сообщение начало передаваться из предыдущего узла маршрута, α – параметр системы, отвечающий за то, после приема какой части сообщения транзитный узел начинает ретрансляцию, $0 < \alpha \leqslant 1$, $b = K_1(t_{i-1},c(t_{i-1},q_{i-1},q_i))b_0 + K_2$. Скорость передачи сообщения по $i-$му каналу-ребру маршрута в момент времени t_i, $i = \overline{1,r}$, будем считать равной наименьшей из скоростей передачи сообщения по всем предыдущим ребрам маршрута и самого i-го ребра:

$$v(t_i,c(t_i,q_{i-1},q_i)) = \min_{j=\overline{1,i}} v(t_j,c(t_j,q_{j-1},q_j)).$$

Скоростью всего маршрута будем считать скорость в его последнем ребре.

Пусть в канале $c = c(t,q,q')$ начиная с момента t передается сообщение β длиной в b с приоритетом γ. В процессе передачи в момент времени t' возможны три «элементарных случая», влияющих на время доставки сообщения (пока предполагается, что в достаточно большой интервал времени происходит только один случай из нижеперечисленных):

1. В момент времени t' сменяется канал: $c' = c(t',q,q')$, тогда продолжительность доставки сообщения становится

$$\hat{T}_1(\beta) = t' + \frac{b - (t'-t)v(t,c)}{v(t',c')}. \tag{3.5}$$

2. В момент времени $t' \geqslant t$ в канал поступает сообщение β' длиной b' и приоритетом γ', таким что $\gamma \neq 0$ или $\gamma' \neq 0$. Тогда продолжительность доставки сообщения β составит

$$\hat{T}_2(\beta) = t' + \frac{(\gamma+\gamma')(b-(t-t')v(t,c))}{\gamma v(t,c)}, \tag{3.6}$$

а продолжительность доставки сообщения β' составит

$$\hat{T}_2(\beta') = \frac{(\gamma+\gamma')b'}{\gamma' v(t,c)}. \tag{3.7}$$

При этом, если сообщение β будет доставлено до окончания доставки сообщения β', т.е. если $\hat{T}_2(\beta) < t' + \hat{T}_2(\beta')$, то выражение (3.7) изменится

как

$$\hat{T}_2(\beta') = \hat{T}_2(\beta) - t' + \frac{b' - \frac{(\hat{T}_2(\beta)-t')v(t,c)\gamma'}{\gamma+\gamma'}}{v(t,c)} = \hat{T}_2(\beta) - t' +$$
$$+\frac{b'(\gamma+\gamma') - (\hat{T}_2(\beta)-t')v(t,c)\gamma'}{(\gamma+\gamma')v(t,c)} = \frac{b+b'-v(t,c)(t-t')}{v(t,c)} = \frac{b+b'}{v(t,c)} + t' - t, \tag{3.8}$$

а если сообщение β' будет доставлено до окончания доставки сообщения β, т.е. $\hat{T}_2(\beta) > t' + \hat{T}_2(\beta')$, то выражение (3.6) изменится как

$$\hat{T}_2(\beta) = t' + \hat{T}_2(\beta') + \frac{b - t'v(t,c) - \frac{\hat{T}_2(\beta')v(t,c)\gamma}{(\gamma+\gamma')}}{v(t,c)} = \frac{b+b'}{v(t,c)}. \tag{3.9}$$

3. В несовместимый с c канал начинают в момент времени t' передавать сообщение β' длиной b'. Тогда продолжительность доставки сообщения β становится

$$\hat{T}_3(\beta) = t' + \frac{b - (t'-t)v(t,c)}{v(t',c)}, \tag{3.10}$$

где $v(t',c) = v(t,c)/2$ (несложно видеть, что если сообщение начинает передаваться по $n = n(t,c)$ несовместимым каналам, то $v(t',c) = v(t,c)/n$). Если сообщение β' будет доставлено до окончания доставки сообщения β (тогда продолжительность доставки сообщения β' будет $t'' = b'/v(t',c)$), то продолжительность доставки сообщения β станет

$$\hat{T}_3(\beta) = t' + t'' + \frac{b - (t'-t)v(t,c) - (t''-t')v(t',c)}{v(t,c)}. \tag{3.11}$$

Алгоритм исследования задержек в системе связи вкратце следующий:

Шаг 1. Разбить граф $\Gamma_f(t)$ на подграфы $\Gamma_f^{(i)}(t)$, такие что для любой вершины $z = z(t,q)$ графа $\Gamma_f^{(i)}(t)$ справедливо, что, если q' является адресатом в расписании $R(q(t,z))$, то $z' = z(t,q')$ является вершиной графа $\Gamma_f^{(i)}(t)$. Очевидно, что каналы в таких подграфах освобождаются и занимаются независимо и вместо изучения всего графа целиком можно изучать передачу информации по меньшим по размеру подграфам.

Шаг 2. В каждом графе $\Gamma_f^{(i)}(t)$ для каждого t построить упорядоченный список $TC1_f^{(i)} = \{\tau_1, \ldots, \tau_m\}$ моментов, в которых происходит «элементарный случай» 1 (смена канала).

Ш а г 3 . Для каждой вершины q графа $\Gamma_f^{(i)}(t)$ просмотреть расписание $R(q)$ и построить упорядоченный по возрастанию t список

$$TTr_f^{(i)} = \{(t_j; q_j, q'_j, b_j, \tilde{t}_j, \gamma_j) | j = \overline{1,l}\}$$

моментов, в которых происходит рассылка сообщений с адресатом q' в $\Gamma_f^{(i)}(t)$.

Ш а г 4 . Для каждого элемента $(t_j; q_j, q'_j, b, \tilde{t}_j, \gamma_j)$ из $TTr_f^{(i)}$ последовательно вычислить время прохождения сообщения T_j, учитывая по формуле (3.4) накладные расходы и увеличивая или уменьшая время прохождения в соответствии с формулами (3.5) - (3.11), если на время передачи информации по расписанию придется момент из $TC_f^{(i)}$ или же возникнут «элементарные случаи» 1 или 2, моменты возникновения которых последовательно рассчитываются.

С этой целью полагаем $j = 1$, множество моментов случая 2 $TC2_f^{(i)} = \{\}$, множество моментов случая 3 $TC3_f^{(i)} = \{\}$, множество времен прохода сообщений через транзитные узлы $TM_f^{(i)} = \{\}$ и совершаем:

Ш а г 4 . 1 . Упорядочиваем $TTr_f^{(i)} \cup TM_f^{(i)}$ по возрастанию t. Выбираем $t_j \in TTr_f^{(i)} \cup TM_f^{(i)}$. Вычисляем $T_j = t_j + (b(t_j, q_j, q'_j)/v(t_j, c(t_j, q_j, q'_j)))$.

Ш а г 4 . 2 . Полагаем $TC_f^{(i)} = TC1_f^{(i)} \cup TC2_f^{(i)} \cup TC3_f^{(i)}$, упорядочиваем $TC^{(i)}$ по возрастанию. Если существуют t_k^c, $t_{k+1}^c \in TC^{(i)}$, такие что

$$t_k^c < t_j < t_{k+1}^c \text{ и } T_j > t_{k+1}^c \text{ или}$$

$$0 < t_j < t_1^c = \min TC_f^{(i)} \text{ и } T_j > t_1^c,$$

то изменяем T_j в соответствии с формулами (3.6)–(3.11), причем нужную формулу выберем в зависимости от того, моментом какого «элементарного случая» является t_{k+1}^c или t_1^c.

Ш а г 4 . 3 . Если $T_j > t_{j+1}$ и q_{j+1}, q'_{j+1} таковы, что при пересылке между ними информации в момент времени t_{j+1}

– задействуется канал между q_j, q'_j, то добавляем t_{j+1} в $TC2_f^{(i)}$,
– задействуется зависимый от $c(T_j, q_j, q'_j)$ канал, то добавляем t_{j+1} в $TC3_f^{(i)}$,
– $c(t_j, q_j, q'_j)$ – путь из более, чем одного ребра и q_s, q'_s – соседние узлы в этом пути, между которыми в момент t_{j+1} передается сообщение размером b (включая накладные расходы), то добавляем $t_s + b\alpha/v(t_s, c(t_s, q_s, q'_s)) + d_s(t_s)$, где t_s – время попадания сообщения в узел q_s, $t_j < t_s < t_{j+1}$, $d_s(t_s)$ – задержка сообщения в узле q_s, в $TM_f^{(i)}$,

– переходим к шагу 4.2.

Иначе, если $j \leqslant l$ – увеличиваем j на единицу, переходим к шагу 4.1, если $j > l$ – к шагу 5.

Шаг 5. Конец.

Можно заметить, что основные элементы модели – сообщения, каналы и скорость передачи аналогичны, соответственно, требованиям, машинам и ресурсам задачи построения расписания выполнения работ проекта с учетом отношений предшествования и ограничения на ресурсы (RCPSP, [118]).

В разделе 3.2 покажем работу алгоритма на конкретном примере.

3.2 Демонстрация работы алгоритма

Для проверки модели выберем типичный фрагмент военной сети связи $\Gamma_1(t)$, в котором агенты q_{11}, q_{12}, q_{13} – командиры отделений связаны с командиром взвода q_{14}, который, в свою очередь, соединен с командиром роты q_{21}, а последний – с командиром батальона q_{31}. Нижестоящие агенты периодически докладывают вышестоящим об изменениях в обстановке и об исполнении приказов, а вышестоящие направляют нижестоящим различные указания. Агенты обычно пользуются для обмена сообщениями высокоскоростными каналами одного типа, а в случае невозможности их применения из-за помех или особенностей рельефа – используют низкоскоростные каналы, также одного типа (рисунок 3.2).

Для того чтобы избежать излишних вопросов, длины сообщений, время рассылки и максимальную пропускную способность каналов будем исчислять в условных безразмерных единицах.

Здесь $C = \{c_1, \ldots, c_9\}$, $Q = Q_1 = \{q_{11}, q_{12}, q_{13}, q_{14}, q_{21}, q_{31}\}$,

$$v_{max}(c_i) = \begin{cases} 400 \text{ единиц в такт}, & i = \overline{1,5}, \\ 4 \text{ единицы в такт}, & i = \overline{6,9}. \end{cases}$$

Функция совместимости каналов com задается таблицей 2. Моменты $\tau_i \in \mathbb{Z}$, $i = \overline{0,3}$, изменения графа таковы: $\tau_0 = 0$, $\tau_1 = 350$, $\tau_2 = 801$, $\tau_3 = 821$.

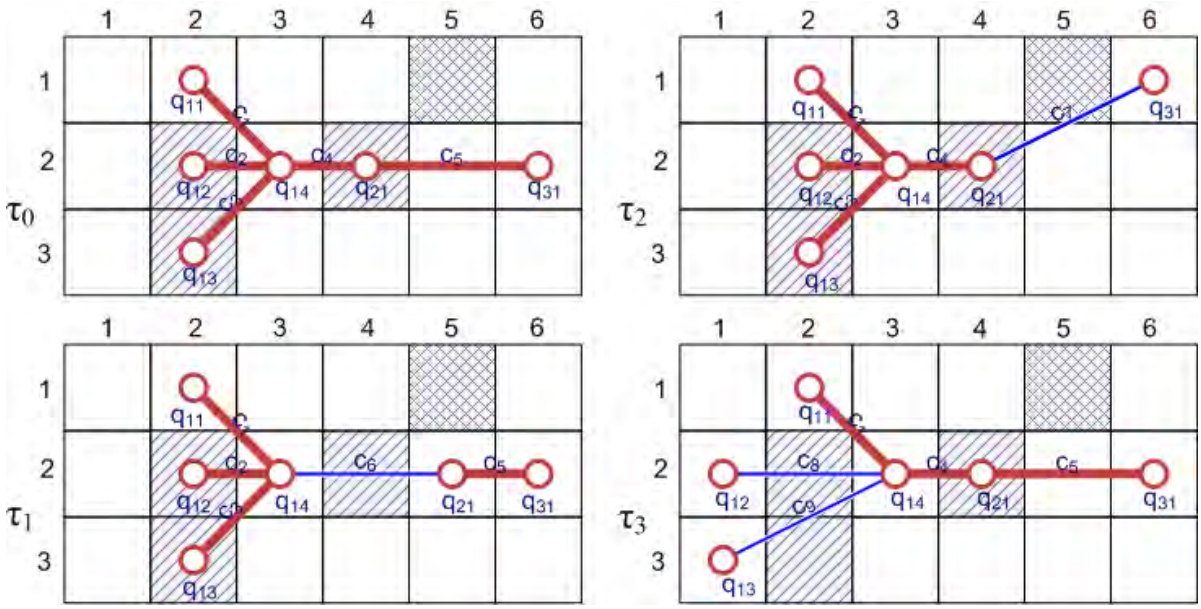


Рисунок 3.2 — Поведение модельной системы связи. Штриховкой обозначена степень проницаемости клетки для радиоволн, чем гуще штриховка, тем хуже проницаемость

Таблица 2 — Значения функции com

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9
c_1	1	0	0	1	1	1	1	1	1
c_2	0	1	0	1	1	1	1	1	1
c_3	0	0	1	1	1	1	1	1	1
c_4	1	1	1	1	1	1	1	1	1
c_5	1	1	1	1	1	1	1	1	1
c_6	1	1	1	1	1	1	1	1	1
c_7	1	1	1	1	1	1	1	1	1
c_8	1	1	1	1	1	1	1	1	0
c_9	1	1	1	1	1	1	1	0	1

Функция весов каналов задается как

$$\psi_1(c_i) = \begin{cases} (0{,}100), & i = \overline{1{,}5}, \\ (1{,}200), & i = \overline{6{,}9}. \end{cases} \tag{3.12}$$

Первая компонента $\psi_1(c_i)$ соответствует способности канала проницать клетки, вторая – максимальной дальности (в клетках), на которой может работать канал.

Клетки (i_1,j_1) и (i_2,j_2) могут быть соединены каналом c только если

$$\psi_1^k(c) \geqslant \psi_2^k(L(i_1,j_1;i_2,j_2)), \quad k=\overline{1,2}.$$

где ψ_2 – функция, определяемая соотношениями (3.1), (3.2).

Предполагается, что накладные расходы несущественны. Расписание агентов пусть имеет вид:

– для q_{ij}, $i=1$, $j=\overline{1,3}$, $R(q_{ij})=\{(600+100p,q_{14},300,30,1)|p=\overline{0,4}\}$;
– $R(q_{14}) = \{(400+100p,q_{1k},300,30,1)|p=\overline{0,6},k=\overline{1,3}\} \cup \{(800+150p,q_{21},900,30,1)|p=\overline{0,2}\}$;
– $R(q_{21}) = \{(200+150p,q_{14},900,30,1)|p=\overline{0,6}\} \cup \{(1000+200p,q_{31},850,30,1)|p=\overline{0,1}\}$;
– $R(q_{31})=\{(200p,q_{21},900,30,1)|p=\overline{0,6}\} \cup \{(800,q_{14},950,30,1)\}$.

Функция q задается таблицей 3. Свойства местности, изображенной на рисунке

Таблица 3 — Значения функции q

	$\tau_0 \leqslant t < \tau_1$	$\tau_1 \leqslant t < \tau_2$	$\tau_2 \leqslant t < \tau_3$	$t \geqslant \tau_3$
q_{11}	(1,2)	(1,2)	(1,2)	(1,2)
q_{12}	(2,2)	(2,2)	(2,2)	(2,1)
q_{13}	(3,2)	(3,2)	(3,2)	(3,1)
q_{14}	(2,3)	(2,3)	(2,3)	(2,3)
q_{21}	(2,4)	(2,5)	(2,4)	(2,4)
q_{31}	(2,6)	(2,6)	(1,6)	(2,6)

3.2, будут описываться следующей матрицей

$$\mathcal{U}=(u_{ij})_{j=\overline{1,6}}^{i=\overline{1,3}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{3.13}$$

Множество ребер графа потребностей в связи $\Pi_1=(Q,E_1)$ будет иметь вид

$$E_1=((q_{11},q_{14}),(q_{12},q_{14}),(q_{13},q_{14}),(q_{14},q_{21}),(q_{21},q_{31})).$$

На основе графа Π_1, матрицы $\mathcal{U}$ (3.13) и функции весов каналов ψ_1 (3.12) конструируется функция инцидентности φ_1 графа связи Γ_1, вид которой показан в таблице 4. Например, в момент времени τ_0 клетки $(1,2)$ и $(2,3)$ соединены

каналом c_1, так как в этих клетках, согласно таблице 3, находятся агенты q_{11} и q_{14}, $(q_{11},q_{14}) \in E_1$, $\psi_1^1(c_1) = 0 \geqslant \psi_2^1(L(1,2;2,3)) = 0$ и $\psi_1^2(c_1) = 100 \geqslant \psi_2^2(L(1,2;2,3)) = \sqrt{2}$ и

$$c_1 \in \arg\max_{c \in C} v_{max}(c).$$

Таблица 4 — Значения функции φ_1

	τ_0	τ_1	τ_2	τ_3
c_1	((1,2),(2,3))	((1,2),(2,3))	((1,2),(2,3))	((1,2),(2,3))
c_2	((2,2),(2,3))	((2,2),(2,3))	((2,2),(2,3))	-
c_3	((3,2),(2,3))	((3,2),(2,3))	((3,2),(2,3))	-
c_4	((2,3),(2,4))	((2,3),(2,4))	-	((2,3),(2,4))
c_5	((2,4),(2,6))	-	-	((2,4),(2,6))
c_6	-	((2,3),(2,5))	-	-
c_7	-	-	((2,4),(1,6))	-
c_8	-	-	-	((2,1),(2,3))
c_9	-	-	-	((3,1),(2,3))

В соответствии с шагом 1 алгоритма, граф $\Gamma_1(t)$ может быть в любой такт разбит на два подграфа

$$\Gamma_1^{(0)}(t) = (\{z(t,q_{11}), z(t,q_{12}), z(t,q_{13}), z(t,q_{14}), z(t,q_{21})\}, C, \varphi_1),$$

$$\Gamma_1^{(1)}(t) = (\{z(t,q_{21}), z(t,q_{31}), z(t,q_{14})\}, C, \varphi_1).$$

Легко видеть, что обмен начнется в каналах между (q_{31}, q_{21}) и сообщения, отправленные от q_{31} к q_{21} в моменты времени $t_{2p+1} = 200p$, $p = \overline{0,3}$, будут приходить в срок

$$T_{2p+1} = 200p + \frac{900}{400} = 202{,}25.$$

В момент $t_8 = 800$ произойдет «элементарный случай» 2, скорость передачи упадет в 2 раза и сообщение от q_{31} к q_{21} должно было бы прийти за

$$T_8 = 800 + 2 \cdot \frac{900}{400} = 804{,}5.$$

Так как в момент $\tau_2 = 801 > T_8$ происходит «элементарный случай» 1 – смена канала между (q_{31}, q_{21}), то сообщение от q_{31} к q_{21} придет за время

$$T_8 = 801 + \frac{900 - (801 - 800)200}{4} = 976.$$

В момент $\tau_3 = 821$ опять произойдет «элементарный случай» 1, так как $T_8 > \tau_3$ и поэтому необходимо изменить значение T_8 так, что

$$T_8 = 821 + \frac{900 - (801 - 800)200 - (821 - 801)2}{200} = 824{,}3.$$

Аналогично, время доставки сообщения от q_{31} к q_{14} составит (при сделанном для упрощения предположении, что сообщение было задержано в узле q_{21} до момента $t_{11} = 811$ в виду перегрузки канала)

$$T_{11} = T_8 + \frac{950 - 900}{400} = 824{,}425.$$

В последующие моменты времени t_{14}, $t_{15} = 1000$, t_{18}, $t_{19} = 1200$ будет происходить только «элементарный случай» 2. Поэтому время доставки

$$T_i = t_i + 2 \cdot \frac{850}{400} + \frac{900 - 850}{400} = t_i + 4{,}375, \quad i = \overline{14, 15},$$

$$T_i = t_i + 2 \cdot \frac{850}{400} = t_i + 4{,}25, \quad i = \overline{18, 19}.$$

Обмен по каналам, связывающим пару (q_{14}, q_{21}) в соответствующие моменты времени будет (стрелочка означает вычисление нового времени доставки сообщения после тех или иных событий в соответствии с алгоритмом):

– $t_2 = 200$, $T_2 = 202{,}25$.
– $t_4 = 350$, $T_4 = 350 + \frac{900}{4} > 500 \to T_4 = 500 + \frac{900-(500-350)4}{2} = 650$.
– $t_6 = 500$, $T_6 = 800 + \frac{900-(800-650)2-(650-650)4-(650-500)2}{4/5} = 1175 > 801 \to$ $T_i = 801 + \frac{900-(801-800)4/5-(800-650)2-(650-500)2}{400/5} = 804{,}74$.
– $t_7 = 650$, $T_7 = 804{,}74 + \frac{900-(804{,}74-801)400/5-(801-800)4/5-(800-650)2}{400/4} = 807{,}74$
– $t_9 = 800$, $T_9 = 807{,}74 + \frac{900-(807{,}74-804{,}74)400/4-(804{,}74-801)400/5-(801-800)4/5}{400/3} =$ $809{,}99$.
– $t_{10} = 800$,

$$T_{10} = 809{,}99 + \frac{950 - (809{,}99 - 807{,}74)400/3}{400} -$$
$$- \frac{(807{,}74 - 804.74)400/4 + (804{,}74 - 801)400/5 + (801 - 800)4/5}{400} = 810{,}115.$$

– $t_{12} = t_{13} = 950$, $T_{12} = T_{13} = 950 + 2 \cdot 900/400 = 954{,}5$.

– $t_{16} = t_{17} = 1100$, $T_{16} = T_{17} = 1104{,}5$.

В графе $\Gamma_1^{(0)}(t)$ в канале (q_{1j},q_{14}) обмен будет таков:

– $t_i \in \{400{,}500\}$, «элементарный случай» 3, $T_i = t_i + 3/4$.

– $t_i \in \{600{,}600{,}700{,}700{,}800{,}800\}$, «элементарный случай» 3, $T_i = t_i + \frac{300}{400/4} = t_i + 3$.

Далее, в канале (q_{11},q_{14}):

$$t_i \geqslant 900, T_i = t_i + \frac{300}{400/2} = t_i + 1{,}5.$$

В каналах (q_{12},q_{14}), (q_{13},q_{14}), после $\tau_3 = 821$:

– $t_{10} = 900$, «элементарные случаи» 2 и 3, $T_{10} = 900 + \frac{300}{4/3} = 1125$.

– $t_{11} = 1000$, «элементарные случаи» 2 и 3, $T_{11} = 1000 + \frac{300-(1000-900)4/3}{4/6} = 1250$.

Видно, что, например, в моменты t_{10}, t_{11} сообщения не доставляются в установленный срок, в точности таким же образом и по той же причине, что и в многократно повторенных автором наблюдениях реальной сети военной связи.

Результаты, полученные в настоящем разделе, хорошо согласуются с экспериментами на стенде. Самый простой способ смоделировать фрагмент системы связи с помощью стенда, не прибегая к крайне дорогостоящей и неоднозначно функционирующей военной технике – соединить 6 компьютеров с двумя сетевыми платами, одна из которых обеспечивает передачу данных со скоростью 100 Мбит/с, а вторая – 10 Мбит/с, причем нужно пересылать данные между компьютерами и включать и отключать сетевые платы по определенному расписанию. Разумеется, необходимо скорректировать объем пересылаемых данных пропорционально столь высоким скоростям передачи. При этом следует иметь, однако, в виду, что в реальности стандартный протокол TCP в глобальной сети (wide area network) военной связи может не употребляться, а вместо него могут использовать различные специальные протоколы разной степени продуманности и документированности на основе UDP.

3.2.1 Задачи, для решения которых проводится моделирование.

Модель, предложенная в данной главе, позволяет дать ответы на следующие вопросы, представляющие большой практический интерес:

1. В любой ли момент времени t заданный граф системы связи $\Gamma_f(t)$ обеспечивает плановую передачу сообщений от всех источников за заданное время?
2. Какое количество информации передается по заданному ребру $\Gamma_f(t)$ в любой заданный момент времени? Не выбраны ли каналы так, что это количество всегда много меньше или больше того, что может пропустить ребро, то есть не будет ли время доставки сообщения по данному ребру всегда много больше или много меньше максимально допустимого? Иначе говоря, адекватен ли подбор средств связи у агента его потребностям?
3. Возможно ли подобрать некую общую характеристику местности (т.е. значений компоненты состояния u_{ij}) и графа потребностей в связи Π_f так, чтобы от них зависела вероятность отказа системы связи (т.е. невозможности обеспечить передачу данных за заданное время) в одном или нескольких ребрах?
4. Сколько внеплановых сообщений можно одновременно разослать без отказа системы связи? Возможно ли подобрать множество каналов (т.е. оснастить соответствующим образом агентов средствами связи) так, чтоб плановая и внеплановая рассылка сообщений осуществлялась бы без отказов системы связи или с минимальным количеством отказов?
5. Как оптимально установить задержки в транзитных узлах и скорректировать расписание, так, чтобы при выдерживании установленного времени доставки минимизировать возникновение задержек из-за «элементарных случаев»?

Для ответов на эти вопросы необходимо знать, во-первых, существует ли путь в графе системы связи между клетками, занимаемыми теми или иными агентами, а во-вторых – сколько в действительности займет передача информации по данному пути. Отметим, что в используемых в Вооруженных силах РФ телекоммуникационных сетях, являвшихся основным предметом исследований автора (ОКР «Созвездие М2» [119]), применяются протоколы, исключающие

отказ в передаче данных из-за сильного превышения объема этих данных пропускной способности канала, однако остро стоят проблемы устаревания информации и наличия в сложном канале узких мест, через которые в разумный срок можно передать только очень ограниченный объем данных.

При этом, предлагаемая модель не требует знания используемых протоколов передачи данных, что может оказаться немаловажным как в случае исследования системы связи вероятного противника, так и в случае моделирования своей собственной сети связи, когда число используемых протоколов может исчисляться десятками и все они сильно нестандартны.

Дальнейшим направлением развития будет являться исследование случаев, когда можно полагать, что время доставки и размер сообщений будут являться случайными величинами с заданным распределением. Также планируется изучать системы, в которых время передачи в расписании одного агента будет увязано с временем получения этим агентом сообщения от другого агента.

3.3 Алгоритмы автоматической организации сети связи мобильных агентов

3.3.1 Введение

Задача распределения ограниченных ресурсов и построения оптимального расписания в сетевых структурах, в том числе и иерархических структурах транспортного типа, достаточно хорошо известна, и каждый год появляется большое количество статей по этой теме. Например, можно упомянуть работу [120], посвященную планированию железнодорожных перевозок, в [121] исследуются задачи распределения ресурсов в иерархических системах, в [122] рассматривается задача согласования входных и выходных параметров участка газотранспортной системы и др. Обычно в таких задачах необходимо распределить ресурсы между вершинами некоторого графа, причем поток ресурсов через ребра графа так или иначе ограничен. Решением таких задач будет, например, оптимальное в каком-либо смысле расписание или распределение ресурсов (не являющихся элементами самого графа) по вершинам графа.

В данном разделе, однако, будет описан принципиально иной тип задач распределения ресурсов, часто встречающийся в задачах планирования военной связи (и, в меньшей степени, в гражданской связи), точнее, в задаче распределения и назначения частот. В описываемой задаче ресурсом являются сами ребра некоторого графа (соответствующие каналам связи), которые необходимо распределить между парами вершин в соответствии с заданным шаблоном. Сам шаблон, который далее будет именоваться графом потребностей, аналогичен сети потребностей и возможностей из работы [123]. Вершинами – потребителями ресурсов являются пользователи системы связи или/и их технические средства.

При этом характер взаимоотношений между агентами определенных типов в системе намного более постоянен, чем конкретный состав агентов. Решением такой задачи является вариант распределения каналов, отвечающий некоторым требованиям. Будет предложен подход к решению таких задач, ориентированный на работу не с конкретными агентами системы, а с абстрактной структурой системы агентов, «отвязанной» от самих агентов.

Также стоит отметить, что несмотря на большое количество литературы по частотному планированию (см., например, монографию [124]), в ней основное внимание уделяется вопросу физической совместимости частот и оптимального расположения радиостанций. Вопросы распределения и перераспределения уже заведомо электромагнитно-совместимых частот по нескольким тысячам мобильных радиостанций (а это обычная ситуация для воинского формирования тактического уровня) затрагиваются довольно мало. Обычно принято формулировать задачу оптимального назначения частотных каналов как задачу минимальной раскраски графа, в которой, в отличие от классической постановки задачи, смежность вершин определяют ограничения на использование не только одного цвета, но и некоторых их комбинаций (см., например, [125; 126]). Однако такая постановка обычно не учитывает ни иерархии агентов, которым надо назначить частоты, ни возможных изменений в составе агентов, ни случая недостатка частот. При этом подготовка списка электромагнитно-совместимых частот – довольно трудная вычислительно задача и решать ее каждый раз заново при каждом изменении состава радиостанций, как иногда предлагается в более специализированных изданиях, – дело крайне непростое.

Возможны и другие подходы к управлению спектром, применяемые в основном в сетях когнитивного радио и основанные, например, на теории игр

[127], но ориентированные также на существенно более простую систему связи (фактически состоящую из равноправных средств, соединяющихся по типу «точка–точка»), чем рассматриваемая в настоящем разделе.

Повышению эффективности настройки систем специального назначения посвящено достаточно работ [128–130]. В [128; 130] представлены принципы функционирования автоматизированной частотно-адаптивной радиолинии (ЧАРЛ) специального назначения в условиях априорной неопределенности относительно изменяющихся условий ведения связи. ЧАРЛ использует группу радиочастот. Данные передаются на рабочей частоте до тех пор, пока качество связи остается удовлетворительным. Если в результате анализа радиоканала выясняется, что качество связи снижается, например в результате воздействия помех, происходит переход на одну из резервных частот, на которой качество связи наилучшее с точки зрения используемых критериев. Выбор наилучших частот и установление связи между передатчиком и приемником осуществляются с помощью обмена командами управления.

В [130] показана адаптация радиостанции «Пирс» по частоте для улучшения качества работы (но не для настройки канала связи) в ДКМВ-диапазоне. Это позволило в соответствии с текущими условиями ведения связи автоматически изменять скорость передачи данных при возможном одновременном изменении количества субчастот сигнала и их расстановки в полосе частот.

В [33] описывается сеть, содержащая несколько головных и подчиненных радиостанций. Функционирование сети начинается с того, что подчиненные радиостанции приступают к рассылке маячков настройки на не заданных заранее частотах. Маячки обнаруживаются головными радиостанциями с помощью сканирующего приемника. Далее головные радиостанции определяют настроечные данные, соответствующие обнаруженным маячкам, настраиваются на основе этих данных и рассылают на обнаруженных частотах подчиненных радиостанций маячки ответа, после получения которых подчиненные радиостанции переходят в сеть головных радиостанций.

3.3.2 Постановка задачи

Для дальнейшего изложения нам необходимо дать математическое описание системы связи и «настройки системы связи» (распределения канального ресурса), воспользовавшись понятиями, предложенными автором в работе [13]. Задача распределения ресурсов, упомянутая во введении, в самом простом виде обычно формулируется следующим образом.

Объекты Даны такие объекты:

1. задано множество классов средств
$$CComm = \{ccomm_1, \ldots, ccomm_s\};$$
2. задано множество средств
$$Comm = \{(comm_1; ccomm_{i_1}), \ldots, (comm_n; ccomm_{i_n})\},$$
где $comm_i$ – идентификатор i-го средства, а $ccomm_{i_j}$ – класс i_j-го средства, $1 \leqslant i \leqslant s$;
3. задано множество агентов
$$Ag = \{ag_1, \ldots, ag_m\};$$
4. между множествами Ag и $Comm$ задано отношение принадлежности
$$f_{have} : Comm \to Ag,$$
и если для $comm \in Comm$ $f_{have}(comm) = ag$, то будем говорить, что агент ag обладает средством $comm$;
5. задано множество классов каналов
$$CF = \{cf_1, \ldots, cf_l\}.$$
В реальных задачах класс соответствует частоте или пропускной способности или подобным характеристикам канала;
6. задана функция возможностей средств
$$f_{comm} : CComm \times CF \to \mathbb{Z},$$
определяющая, сколько максимально каналов заданного типа может образовать средство заданного типа одномоментно;

7. задано множество каналов

$$F = \{(f_1; cf_{i_1}), \ldots, (f_p; cf_{i_p})\},$$

где f_i – идентификатор i-го канала, а cf_{i_j} – класс i_j-го канала, $1 \leqslant i \leqslant p$;

8. задан не содержащий петель маркированный мультиграф потребностей (пример такого графа показан на рис. 3.3)

$$\mathrm{Req} = (Ag, E_{\mathrm{Req}}, \varphi_{\mathrm{Req}}), \quad E_{\mathrm{Req}} \subseteq Ag^2 \times CF,$$

$\varphi_{\mathrm{Req}} : Ag \times E_{\mathrm{Req}} \to \mathbb{Z}$ – функция инцидентности, которая показывает, какие именно агенты каким количеством каналов какого класса должны быть соединены между собой; в качестве меток ребер выступают классы каналов;

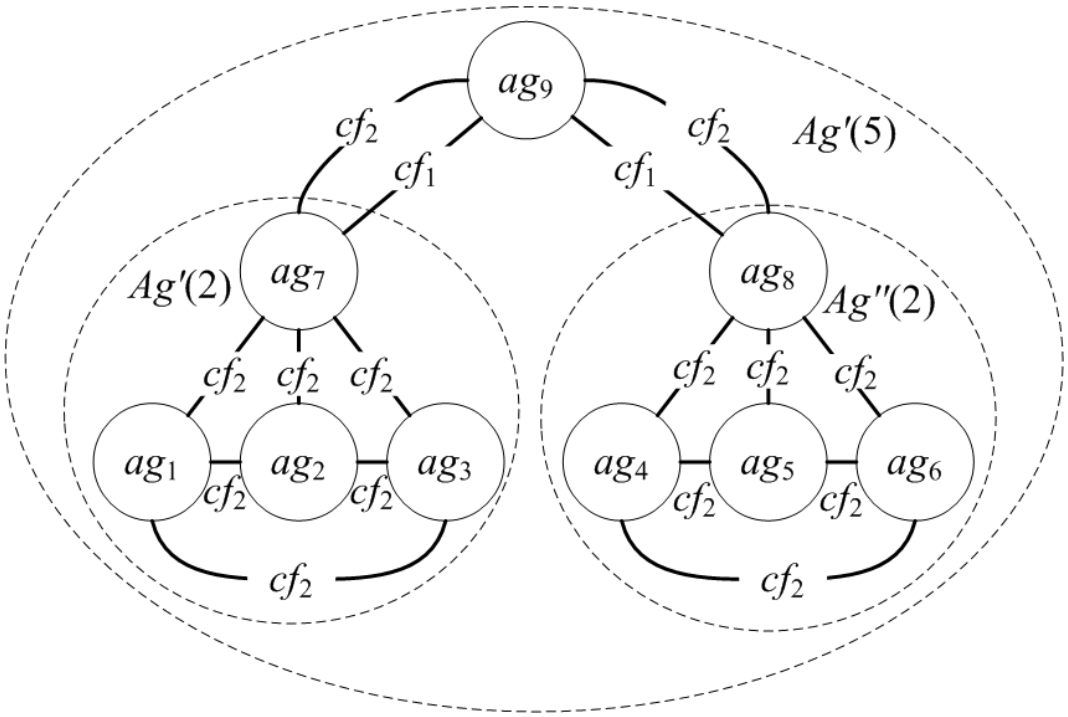


Рисунок 3.3 — Граф потребностей Req

9. задана функция емкости класса канала

$$f_e : CF \to \mathbb{N},$$

определяющая, сколько максимально агентов могут одновременно использовать канал данного класса.

Условия Необходимо распределить каналы из F по всем средствам из $Comm$, получив в результате граф связи

$$\Gamma = (Comm, E, \varphi), \quad E \subseteq F,$$

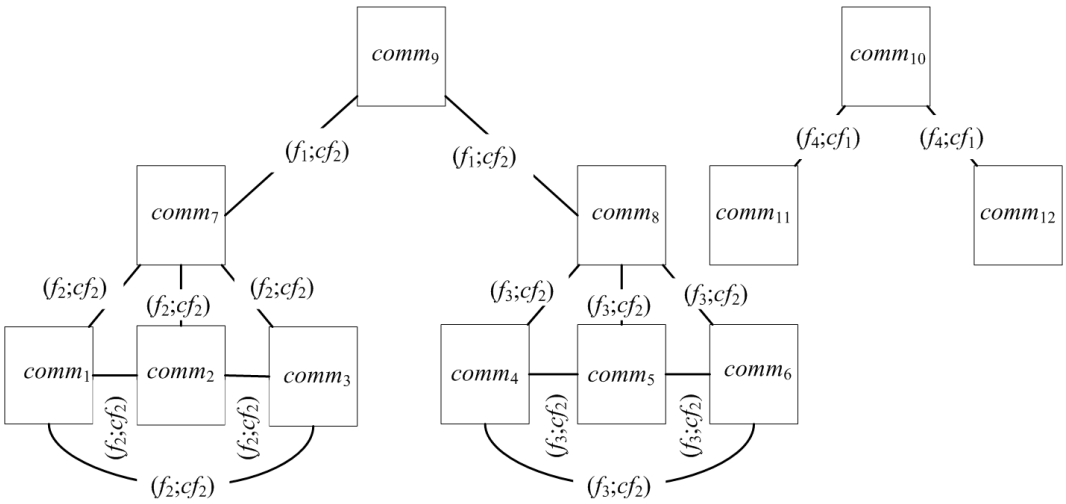


Рисунок 3.4 — Граф связи Г

$\varphi : Comm \times E \to \{0,1\}$ – функция инцидентности, так чтобы

1. каждому ребру графа Req соответствовало одно ребро графа Г или непрерывный путь в графе Г (случай ретрансляции);
2. каждой вершине ag графа Req соответствовала бы вершина $comm$ графа Г, такая что $f_{have}(comm) = ag$;
3. количество ребер e графа Г, маркированных одним и тем же каналом $(f, cf) \in F$, не превышает емкости канала данного класса $f_e(cf)$;
4. количество ребер e графа Г, маркированных каналом (f, cf) и инцидентных вершине $comm$, не превзойдет возможности средства $f_{comm}(ccomm, cf)$.

Пример графа Г, соответствующего графу Req, показанному на рис. 3.3, изображен на рис. 3.4.

Подытожим, что в исследуемой модели искомой переменной является матрица инцидентности графа Г, неформально иногда называемая «вариант распределения радиочастот». Известными параметрами модели являются каналы F, средства $Comm$, агенты Ag, отношения f_{have}, f_e, f_{comm} и граф Req.

Процесс распределения каналов из множества F между агентами и принадлежащими им средствами часто неформально называется «настройкой системы связи». Для упрощения мы будем считать, что все каналы из F заведомо совместимы электромагнитно, иначе необходимо будет добавлять еще метки длин ребер Г и ограничения на возможные назначения каналов для близких друг к другу средств, после чего применять уже известные методы типа метода координационных колец.

3.3.3 Недостатки модели

Хотя автору и не приходилось видеть статьи с описанием задачи распределения ресурсов такого типа, но она вполне типична в определенных сферах деятельности и решается обычным перебором. Применительно к военной связи (в которой агентами являются должностные лица Вооруженных сил) такой алгоритм был реализован автором в составе большого коллектива в виде разработанного в АО «Концерн «Созвездие» программного обеспечения ЕСВП [14]. При традиционном подходе к обсуждаемой задаче распределения наблюдаются следующие недостатки:

1. Граф Req в буквальном смысле рисуется оператором в течение достаточно продолжительного времени, и при каждом изменении состава множеств $Comm$, Ag необходимо перерисовывать его заново, что при мощности множеств $Comm$, Ag в сотни элементов может затянуться на неопределенный срок.

2. Более того, в реальности множество каналов F не постоянно: обычно каналы выбывают из него со временем непредсказуемым образом в силу преднамеренных или непреднамеренных помех. При существенном изменении множества каналов граф Г, разумеется, необходимо конструировать заново. При этом отношения между новыми агентами будут иметь примерно такую же структуру, что и между их предшественниками.

3. В реальности агенты из Ag, как правило, неравноправны, и для каждого $ag_i \in Ag$ может быть задан вес α_i, определяющий, какую долю имеющихся в F каналов следует выделить данному агенту в ситуации нехватки каналов на всех агентов.

4. Граф Г доводится до средств (радиостанций) с помощью запоминающих устройств разного типа. Эта методология восходит к временам, когда средств связи было относительно немного и сгруппировать их было несложно.

5. Граф Г может быть послан и по каналу связи, но возникает вопрос: ведь именно Г и задает настройку канала связи (частоту, как минимум) и как послать Г, когда каналы вообще еще никак не настроены?

6. Наконец, при случайном выходе из строя центра планирования, способного рассчитывать графы Г, вся система вообще постепенно выходит из строя.

Цель настоящего раздела двояка:

1. Дать расширенную формулировку ранее сформулированной задачи, учитывающую вышеупомянутые недостатки.

2. Описать систему связи (безразлично, гражданскую ли, военную ли, хотя первоначально автор работал именно с военными системами связи), в которой каналы назначаются не централизовано, а выбираются каждым средством исходя из характеристик самого средства и положения этого средства в общей системе.

Отметим, что в разрабатываемых сейчас за рубежом перспективных системах военной связи [131] проблемы 1–6 активно решаются с помощью технологии «когнитивной радиосети» (cognitive network), в которой все узлы обладают «осознанием» как изменений частотного спектра (spectrum sensing), как и в предлагавшейся ранее методологии «когнитивного радио», так и общей «плоскостью знаний» (knowledge plane) сети, в которой содержатся, в том числе, сведения о виде, предназначении и состоянии узлов сети. В числе прочих технологий применяется динамическое управление спектром, в ходе которого узлы согласованно занимают или освобождают те или иные частотные каналы в зависимости от своего положения в иерархии узлов, изменений помеховой обстановки или других обстоятельств. В настоящей работе, по сути, рассматривается частная задача динамического управления спектром в когнитивной радиосети – начальное выделение частот в соответствии с ролями агентов.

Стоит отметить, что алгоритмы автоматической кластеризации известны достаточно давно (см., например [132]), однако они применяются в основном для организации сенсорных сетей, в которых основным принципом построения кластеров является энергетическая эффективность, а не положение средства в какой-либо иерархии.

Рассмотрим вкратце вопросы обеспечения безопасности организация настройки радиосети по заранее не организованному каналу связи посредством обмена маячками настройки (МН) и маячками ответа (МО) между центром управления связью (ЦУС) и настраиваемыми радиостанциями. В этом алгоритме все радиостанции снабжались одинаковой сеткой технических каналов, ЦУС рассылал на одном из выбранных из предустановленной сетки каналов МН, тогда как радиостанции сканировали вышеупомянутую сетку каналов и при обнаружении МН отправляли МО. После получения МО ЦУС направлял радиостанции сгенерированные заблаговременно настроечные данные, включающие рабочую частоту и другие параметры. Предполагалось, что настроечные

данные рассчитываются исходя из сведений о составе и структуре настраиваемой системы связи. Вопросы закрытия передаваемых данных решались с помощью асимметричной криптосистемы типа RSA, исключающей необходимость предварительного распространения ключей (автор сознательно не будет касаться возможности использования RSA в России для такой цели, поскольку это вопрос более политический, чем технический).

При организации мобильных беспроводных сетей связи, работающих в условиях изменяющейся помеховой обстановки и содержащих большое количество радиосредств, которые используют разные по составу настраиваемые параметры (например, частоты), необходимо осуществлять их автоматическую или автоматизированную настройку, чтобы время перестройки сети при изменении условий ее функционирования позволяло сохранить эффективность сетевого управления. Это особенно касается распространения ключей шифрования, которые на начальном этапе функционирования систем мобильной связи специального назначения при отсутствии защищенных каналов связи могут быть доставлены только ручным способом на флеш-накопителях, что существенно увеличивает время настройки сети связи и ухудшает адаптивность всей системы управления. Так, согласно Дж. Бойду [133, P. 42], действия по боевому управлению должны осуществляться в течение нескольких часов или даже минут, тогда как ввод ключей экспедицией может затянуться на более длительное время.

В разделе предлагается доставка настроечных радиоданных радиостанциям организовывать автоматически – с помощью маячков по специальным каналам настройки, выбираемым на подходящих по отношению сигнал/шум (ОСШ; signal-to-noise ratio, SNR) частотах, защищенным с помощью асимметричного шифрования.

Идея управления доступом (media access control, MAC) к среде передачи данных посредством рассылки маячков в различных частотных участках и динамического формирования списка рабочих каналов изложена в [131], однако вопросы защиты информации на уровне MAC в цитируемой статье не рассматривались, как и вопросы первоначальной настройки системы связи.

В [134] описана технология Wake-on-LAN, предназначенная для удаленного включения/выключения устройства. Управляющие данные (так назывваемые магические пакеты) передаются широковещательно по уже организованному каналу. Управляемое устройство находится в дежурном режиме и выдает питание

на сетевой адаптер, который находится в режиме пониженного энергопотребления и просматривает все пакеты, приходящие на его MAC-адрес, но ничего на них не отвечает. Если один из пакетов окажется «магическим», сетевой адаптер выдаст сигнал на включение питания устройства.

Проблемы контроля целостности и защиты информации тоже достаточно широко освещены в литературе. В [135] представлен протокол IPSec, основанный на способе организации защищенной сети связи, в которой последовательно осуществляется обмен видоизмененными IP-пакетами для взаимной аутентификации между узлами сети, а затем происходит обмен шифрованными пакетами, включающими вставку для проверки целостности сообщений.

В [136; 137] представлены протоколы SSL и TLS, с помощью которых может быть организована защищенная сеть связи, где между сервером и другими узлами на транспортном уровне происходят: 1) отправка от сервера сообщения, содержащего открытый ключ сервера, 2) отправка ответного сообщения от узла сети, в котором содержится симметричный сеансовый ключ, зашифрованный открытым ключом сервера, и 3) последующий обмен между сервером и узлом сети информационными сообщениями, зашифрованными сеансовым ключом.

Недостатком указанных криптографических протоколов является то, что они могут работать только при уже действующем транспортном или сетевом уровне сети, что требует организации указанных уровней в открытом режиме. Для разработанного автором метода автоматического развертывания радиосети в разделе 3.5 предлагается отличный от криптографических протоколов SSL, TLS или IPSec способ одновременной организации и защиты каналов связи.

3.3.4 Распределение ресурсов в зависимости от роли агента и в условиях недостатка ресурсов

Ранее уже упоминалась неравноправность агентов из Ag. Вопросы конкретной организации агентов, которая может определяться графом воинской иерархии или иной социальной структурой, выходят далеко за рамки диссертации. Достаточно сказать, что каждому агенту $ag \in Ag$ однозначно соответствует вектор признаков агента

$$\mathbf{I} : ag \leftrightarrow (ag^1, \ldots, ag^q),$$

и задан «вес» агента $\alpha(ag^1, \ldots, ag^q) = \alpha(\mathbf{I}(ag)) = \alpha \circ \mathbf{I}(ag)$, $\alpha \circ \mathbf{I}(ag) \in [0,1]$.

При фактической реализации системы, которая будет далее описываться в настоящем разделе, надо считать, что вектор признаков агента хранится непосредственно в каждом агенте. Можно сказать, что каждый агент еще до начала распределения ресурсов «знает» свои (и только свои) признаки. В системах военной связи такими признаками может быть уровень воинской иерархии, к которой относится агент, тип воинского формирования агента, порядковый номер и тому подобные сведения. Отмечу, что в военных системах такие сведения становятся секретными лишь в совокупности и то, что они содержатся в агенте, точнее, в технических средствах, принадлежащих агенту, никак не скомпрометирует всю систему.

Множество всех векторов признаков агентов обозначим как $\mathcal{S}$. Таким образом, сущности подраздела 3.3.2 дополнятся сущностями:

10. определена функция важности агента

$$\alpha : \mathcal{S} \to [0,1];$$

11. определена биективная функция

$$\mathbf{I} : Ag \to \mathcal{S},$$

$$\sum_{ag \in Ag} \alpha \circ \mathbf{I}(ag) = 1.$$

В случае нехватки ресурсов (каналов) условие 1 подраздела 3.3.2 заменится на

1.
$$\sum_{ag \in Ag} \left| \alpha \circ \mathbf{I}(ag) - \frac{|F(ag)|}{|F|} \right| \to \min,$$

где $F(ag)$ – множество ребер Γ, инцидентных хотя бы одному средству из множества $Comm(ag) \subset Comm$, $Comm(ag) = \{comm \in Comm | f_{have}(comm) = ag\}$.

В результате задача распределения каналов в условиях нехватки канального ресурса превратится в задачу комбинаторной оптимизации.

Когда вручную задается граф Req, фактически происходит следующее. Составитель графа имеет в своем сознании ряд предикатов $\mathfrak{P}_i : \mathcal{S}^{|Ag|} \to \{0,1\}$,

позволяющих ему сгруппировать агентов в группы, которые должны быть связаны на основании неких признаков:

$$\mathfrak{P}_i(\mathbf{I}(ag_1),\ldots,\mathbf{I}(ag_m)) \to \\ \varphi_{\mathrm{Req}}(ag_{j_i}, cf_{k_i}) = n_i \wedge j_i \in J_i \subset \{1,\ldots,m\} \wedge k_i \in K_i \subset \{1,\ldots,l\}, \quad (3.14)$$

т.е. если выполнено некоторое утверждение $\mathfrak{P}_i$ о агентах ag_s, $s = \overline{1,m}$, то некоторые из них должны быть соединены $1 \leqslant n_i \leqslant p$ каналами класса cf_{k_i}.

Разумеется, эти признаки формализуемы, и весь граф Req подраздела 3.3.2 целиком заменяется

8. семейством предикатов $\mathrm{PReq} = \{\mathfrak{P}_i\}$.

Таким образом, возможно заменить сложность постоянного (при каждом распределении ресурсов) конструирования графа Req на сложность первоначальной классификации объектов и выделения признаков их объединения.

При реализации системы связи, которая будет описываться далее в разделе, необходимо полагать, что множество PReq также «содержится внутри» агента.

Как кажется, введенными понятиями можно было бы и ограничиться при формализации задачи, но в задачах формирования сетей военной связи (да и вообще в мобильных сетях) состав множества Ag непостоянен – агенты выбывают со временем, прибывают подкрепления и т.п., хотя структура взаимоотношений между агентами остается по большей части неизменной. Наконец, возможна такая ситуация, когда необходимо последовательно распределить каналы для нескольких одинаковых по организационной структуре и по оснащению, но отличных по составу воинских формирований.

3.3.5 Структурное сходство

Определения Для полного описания поставленной в разделе задачи осталось только определить, какие агенты можно считать сходными. С этой целью можно применить подход из [138]: два графа считаются сходными, если они изоморфны или имеют изоморфные подграфы. Для упрощения будем сперва полагать, что граф Req описывает потребности в связи, например, некоего идеального воинского формирования, состоящего из совершенно одинаковых более мелких единиц.

Определение 15. *Назовем изоморфизмом маркированных одинаковыми метками графов* $\mathrm{Req} = (Ag, E_{\mathrm{Req}}, \varphi_{\mathrm{Req}})$ *и* $\mathrm{Req}' = (Ag', E_{\mathrm{Req}}, \varphi'_{\mathrm{Req}})$ *взаимно однозначное отображение множеств вершин графов*

$$\mathcal{H} : Ag \to Ag',$$

переводящее смежные вершины с меткой β *в смежные вершины с той же меткой* β.

На множестве всех подграфов графа Req (обозначим его 2^{Req}) можно с помощью изоморфизма графов ввести отношение эквивалентности: два подграфа эквивалентны, если они гомоморфны в смысле вышеприведенного определения. Таким образом возможно построить фактор-множество $(2^{\mathrm{Req}}/\sim)$. В вышеприведенном примере идеального воинского формирования классами эквивалентности Req, содержащими более одного подграфа, будут подграфы, соответствующие потребностям в связи взводов, рот, батальонов и т.п. однотипных воинских формирований.

Будем говорить, что два различных эквивалентных подграфа – максимальные эквивалентные подграфы, если никакие содержащие их подграфы уже не эквивалентны никакому подграфу, кроме себя. Вершины ag_1, ag_2, $\mathcal{H}(ag_1) = ag_2$ двух максимальных эквивалентных подграфов можно отождествить друг с другом, введя отношение эквивалентности уже на множестве агентов, полагая, что $ag_1 \sim ag_2$, если $\mathcal{H}(ag_1) = ag_2$. Обозначим $\mathfrak{Ag} = (Ag/\sim)$.

Определение 16. *Назовем структурой графа* $\mathrm{Req} = (Ag, E_{\mathrm{Req}}, \varphi_{\mathrm{Req}})$ *граф* $\mathrm{Req}' = (\mathfrak{Ag}, E_{\mathrm{Req}}, \tilde{\varphi}_{\mathrm{Req}})$*, в котором каждая вершина* $ag \in Ag$ *заменена соответствующим классом эквивалентности* $\mathfrak{ag} \in \mathfrak{Ag}$*,* $ag \in \mathfrak{ag}$.

Для наиболее полного выделения структуры из Req необходимо пользоваться следующим алгоритмом построения изоморфизма:

Алгоритм 8 ((Структурирование))**.**

1. Задается минимальный диаметр выделяемого подграфа $\delta > 0$.
2. Устанавливается $\mathrm{Req}' := \mathrm{Req}$, $i := 0$.
3. Пока $\mathrm{diam}(\mathrm{Req}') > \delta$:
   a) из Req' выделяется максимальный подграф Req_i, изоморфный с изоморфизмом $\mathcal{H}^i$, как минимум, еще одному отличному от себя подграфу Req';

б) устанавливается $\mathrm{Req}' := \mathrm{Req}_i$ (способ выделения подграфа будет описан в подразделе 3.3.6);

в) устанавливается $i := i + 1$.

4. Устанавливается $m := i$.
5. Конструируется общий изоморфизм $\mathcal{H}_\delta$, $\mathcal{H}_\delta(ag) = \mathcal{H}^j(ag)$, если ag является вершиной Req_j и не является вершиной Req_{j+1}.

Очевидно, что построение фактор-множества классов агентов можно производить и через предикаты из PReq, полагая эквивалентными два агента ag_1, ag_2, на которых совпадают значения предикатов из PReq при подстановки в них всех возможных значений других агентов из Ag.

Таким образом, возможно рассматривать при распределении ресурсов не все признаки $(ag^1, \ldots, ag^q)$ агента ag, а лишь те, которые одинаковы для всех агентов класса $\mathfrak{ag}$, $ag \in \mathfrak{ag}$, и распределять ресурсы не между агентами, а между классами агентов, принимая во внимание лишь общее количество агентов. Однократно выделив структуру по какому-либо графу PReq и распределив по данной структуре каналы, можно использовать этот вариант распределения и для других графов потребностей, которые полностью или в части какого-либо своего достаточно большого подграфа можно представить как вариант реализации исходной структуры.

В этом случае возможно добиться как масштабируемости и переносимости решения задачи распределения ресурсов на другие множества агентов с той же структурой, так и ликвидировать зависимости решения задачи от текущего состава агентов.

Остался еще один вопрос: а что если граф потребностей состоит не из в точности изоморфных между собой подграфов, а из в некоторой мере сходных подграфов? В этом случае зададим меру сходства подграфов следующим образом:

Определение 17. *Назовем ε-сходством маркированных одинаковыми метками графов* $\mathrm{Req}' = (Ag', E_{\mathrm{Req}}, \varphi'_{\mathrm{Req}})$ *и* $\mathrm{Req}'' = (Ag'', E_{\mathrm{Req}}, \varphi''_{\mathrm{Req}})$, $Ag' \subseteq Ag$, $Ag'' \subseteq Ag$ *взаимно однозначное отображение множеств вершин графов*

$$\mathcal{H}_\varepsilon : Ag' \setminus BAg' \to Ag'' \setminus BAg'',$$

переводящее смежные вершины из $Ag' \setminus BAg'$ с меткой β в смежные вершины из $Ag'' \setminus BAg''$ с той же меткой β, такое что мера несходства графов

$$\mu(\mathrm{Req}', \mathrm{Req}'') = \frac{|BAg'| + |BAg''|}{|Ag'| + |Ag''| - |Ag' \setminus BAg'|} < \varepsilon.$$

Очевидно, что так определенная мера несходства изоморфных графов равна нулю. Данная мера, в сущности, совпадает с биотопическим расстоянием, которое, в свою очередь, является частным случаем расстояния Штейнхауса.

Если задано $\varepsilon > 0$, то можно определить максимальные ε-сходные подграфы аналогично максимальным изоморфным подграфам. То есть два ε-сходные подграфа максимальные, если никакие содержащие их подграфы графа Req не являются ε-сходными ни с одним подграфом кроме себя самих. Отображение $\mathcal{H}_\varepsilon$ порождает отношение эквивалентности $ag_1 \sim ag_2 \equiv ag_1 = \mathcal{H}_\varepsilon(ag_2)$, и справедливы все вышеупомянутые рассуждения о построении структуры графа Req.

Подытоживая, можно сказать, что задача о распределении каналов между агентами по заданному образцу заменяется на задачу о распределении каналов между классами эквивалентности агентов по заданному образцу с последующей заменой классов агентов на попарно различных произвольных представителей этих классов, имеющихся в данный момент функционирования системы в наличии. Иначе говоря, граф Req подраздела 3.3.2 целиком заменяется

8. структурой графа потребностей $\mathrm{Req}' = (\mathfrak{Ag}, E_{\mathrm{Req}}, \tilde{\varphi}_{\mathrm{Req}})$ или эквивалентным ей набором предикатов.

3.3.6 Структурирование графа потребностей как задача кластеризации

Описанное в предыдущем подразделе структурирование графа близко по своей сути к задаче кластеризации, известной в машинном обучении [139]. Для работы с такой задачей определим, что в пространстве векторов признаков агентов $\mathcal{S}$ задана функция $\rho : \mathcal{S}^2 \to \mathbb{R}$, $\rho(sag_1, sag_2) > 0$, $\rho(sag_1, sag_2) = \rho(sag_2, sag_1)$, sag_1, $sag_2 \in \mathcal{S}$, $sag_1 \neq sag_2$, $\rho(sag,sag) = 0$, $sag \in \mathcal{S}$, которая показывает, насколько близки типы двух агентов.

Если бы кластерная структура графа Req была бы известна заранее (в случае военной системы связи это означает, что сразу полностью известно, из

каких полков, батальонов и т.п. подразделений состоит воинское формирование, для которого мы хотим распределить частоты), то необходимо было бы лишь проверить кластера примерно одинакового размера (различающиеся не более чем на $\mathcal{E}_0 \geqslant 0$) на изоморфность или на ε-сходство. Хотя существует довольно много методов построения изоморфизмов графов, нам, очевидно, подойдет не любой из них. Для выбора нужного отображения рассмотрим приведенный ниже алгоритм.

Алгоритм 9 ((Построение изоморфизма или ε-сходства))**.**

1. Если Ag', $Ag'' \subset Ag$ – два кластера, $0 \leqslant |Ag''| - |Ag'| < \mathcal{E}_0$, то отождествим с вершиной $ag' \in Ag'$ вершину
$$ag'' \in \arg\min_{ag \in Ag''} \rho(\mathbf{I}(ag'),\mathbf{I}(ag)). \tag{3.15}$$
Для того чтобs запретить выбор «за неимением лучшего» в качестве близких по типу слишком уж несходных агентов, можно ввести дополнительно условие
$$\rho(\mathbf{I}(ag'),\mathbf{I}(ag'')) < \mathcal{E}_1. \tag{3.16}$$
Получим семейство отображений $\mathfrak{H}_1 = \{\mathcal{M}^i : Ag' \to Ag''\}$.
2. Отберем из $\mathfrak{H}_1$ только инъективные отображения $\mathcal{H}_\varepsilon : D' \to Ag''$, $D' \subseteq Ag'$, получив в результате семейство отображений $\mathfrak{H}_2$. Отметим, что в силу (3.16) отображение $\mathcal{H}_\varepsilon$ может быть определено не на всем Ag'
3. Обозначим как $B(\mathcal{H}_\varepsilon)$ количество ребер, таких что если ag_1, $ag_2 \in Ag'$ соединены ребром с меткой β, то и $\mathcal{H}_\varepsilon(ag_1)$, $\mathcal{H}_\varepsilon(ag_2) \in Ag''$ соединены ребром с меткой β.
4. Из $\mathfrak{H}_2$ выберем отображение с наибольшей по мощности областью определения D' и с наибольшим значением $B(\mathcal{H}_\varepsilon)$. Таким образом, при выборе $\mathcal{H}_\varepsilon$ необходимо максимизировать величину $|D'| + B(\mathcal{H}_\varepsilon)$.

Если кластерная структура Req заранее не известна, а агенты представлены только своими векторами признаков, поскольку мы заведомо знаем, что агенты организованы иерархически, может быть применен алгоритм иерархической кластеризации с построением дендрограммы (dendrogram). После этого останется лишь просмотреть дендрограмму и применять к кластерам, оказавшимся на одном уровне дендрограммы, алгоритм 9.

Проиллюстрируем все изложенное на максимально упрощенном примере графа Req, изображенного на рис. 3.3. Пусть представленные на графе агенты

Таблица 5 — Значения функции **I** для агентов из рис. 3.3

Агент	Вектор признаков
ag_1	(1,1,1,0)
ag_2	(1,1,2,0)
ag_3	(1,1,3,0)
ag_4	(1,2,4,0)
ag_5	(1,2,5,0)
ag_6	(1,2,6,0)
ag_7	(1,1,7,1)
ag_8	(1,2,8,1)
ag_9	(2,1,9,1)

имеют вектора признаков, приведенные в таблице 5, т.е. $\mathcal{S} \subset \mathbb{Z}^4$. Здесь если

$$\mathbf{I}(ag) = sag = (ag^1, ag^2, ag^3, ag^4),$$

то ag^1 – это уровень агента в иерархии, ag^2 – номер некоторой общности агентов внутри уровня иерархии (например, «1 мотострелковый взвод 1 мотострелкового батальона»), ag^3 – идентификатор агента, ag^4 – признак того, является ли агент ведущим (например, командиром взвода). В качестве меры сходства объектов ag_1 и ag_2, $\mathbf{I}(ag_1) = sag_1 = (ag_1^1, ag_1^2, ag_1^3, ag_1^4)$, $\mathbf{I}(ag_2) = sag_2 = (ag_2^1, ag_2^2, ag_2^3, ag_2^4)$ примем функцию

$$\rho(sag_1, sag_2) = \sum_{i=1}^{3} [ag_1^i - ag_2^i] + \frac{1}{2}[ag_1^4 - ag_2^4], \tag{3.17}$$

где $[x - y] = 0$, если $x - y = 0$ и $[x - y] = 1$ во всех иных случаях.

На рис. 3.3 пунктирной линией показано, что, например, при формировании кластеров из агентов с мерой несходства ρ, строго меньшей чем 2, получатся кластера $Ag'(2)$ и $Ag''(2)$ (и, возможно, другие), а при формировании кластеров из агентов с мерой несходства ρ, меньшей чем 5, – только один кластер $Ag'(5)$.

Если выбирать в качестве соответствующих пары агентов из $Ag'(2)$ и $Ag''(2)$, для которых ρ минимальна (шаг 1 алгоритма 9), то получим изоморфизм $\mathcal{H} : Ag'(2) \to Ag''(2)$, указанный в таблице 6. Вообще возможно несколько вариантов построения изоморфизма, но в силу конструкции функции ρ (3.17) «ведущий» агент в одном кластере всегда будет переходить в «ведущего» агента в другом, что принципиально важно в приложениях.

Таблица 6 — Конструкция изоморфизма $\mathcal{H} : Ag'(2) \to Ag''(2)$

Агент в $Ag'(2)$	ag_1	ag_2	ag_3	ag_7
Агент в $Ag''(2)$	ag_6	ag_5	ag_4	ag_8
Значение ρ	2	2	2	2

В результате применения алгоритма структурирования с минимальным диаметром выделяемого подграфа 2 будет получен граф, изображенный на рис. 3.5, $\mathfrak{ag}_1 = \{ag_1, ag_6\}$, $\mathfrak{ag}_2 = \{ag_2, ag_5\}$, $\mathfrak{ag}_3 = \{ag_3, ag_4\}$, $\mathfrak{ag}_4 = \{ag_7, ag_8\}$, $\mathfrak{ag}_5 = \{ag_9\}$. Таким образом, вместо графа потребностей с девятью вершинами

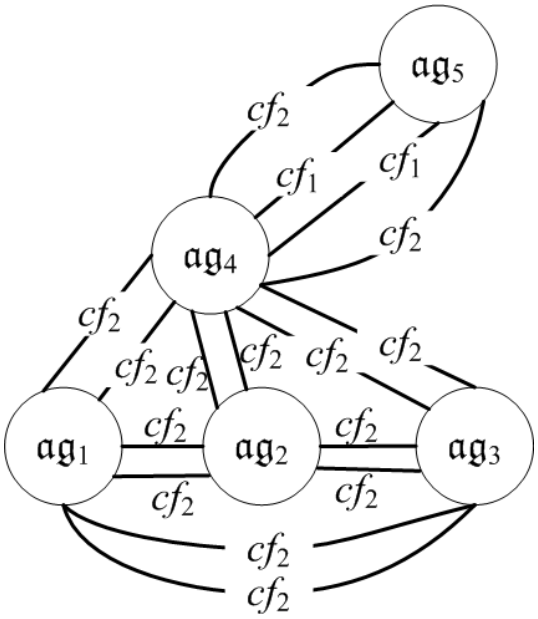


Рисунок 3.5 — Структура Req' графа потребностей из рис. 3.3

получен граф с пятью вершинами. Если взять меньший минимальный диаметр выделяемого подграфа, то возможно построить структуру и с тремя вершинами. При построении графа связи Г такая процедура может существенно ускорить процесс распределения частот, так как нужно будет просматривать практически вдвое меньшее количество вершин графа потребностей.

Можно отметить, что в разобранном случае правила построения графа Req из множества PReq, упомянутого в разделе 3.3.4, формулируются следующим образом

1. Если $ag_1 \neq ag_2$ и $\rho(\mathbf{I}(ag_1), \mathbf{I}(ag_2)) < 2$, то вершины ag_1 и ag_2 должны быть соединены ребром, помеченным cf_2.

2. Если $(\mathbf{I}(ag_1))^1 = 1$ и $(\mathbf{I}(ag_1))^4 = 1$ и $(\mathbf{I}(ag_2))^1 = 2$, то вершины ag_1 и ag_2 должны быть соединены ребром, помеченным cf_1.

3. Если $(\mathbf{I}(ag_1))^1 = 1$ и $(\mathbf{I}(ag_1))^4 = 1$ и $(\mathbf{I}(ag_2))^1 = 2$, то вершины ag_1 и ag_2 должны быть соединены ребром, помеченным cf_2.

Если задано, что $F = \{(f_1; cf_2),(f_2; cf_2),(f_3; cf_2),(f_4; cf_1)\}$, $f_{have}(comm_i; ccomm_2) = ag_i$, $1 \leqslant i \leqslant 9$, $f_{have}(comm_{10}; ccomm_1) = ag_9$, $f_{have}(comm_{11}; ccomm_1) = ag_7$, $f_{have}(comm_{12}; ccomm_1) = ag_8$, а также что $fcomm(ccomm_1, cf_1) = fcomm(ccomm_2, cf_2) = 16$, $fcomm(ccomm_2, cf_1) = fcomm(ccomm_1, cf_2) = 0$, то из графа Req, показанного на рис. 3.3, или из структуры Req' графа потребностей (рис. 3.5) в соответствии с правилами раздела 3.3.2 легко получается граф связи, приведенный на рис. 3.4.

3.3.7 Пример системы связи, основанной на структурном подходе

Мы привели формальное описание задачи распределения каналов по системе агентов в соответствии с заданным шаблоном, исправляющее практически все недостатки обычного подхода, приведенные в начале работы. Остался лишь один вопрос – можно ли построить систему связи, вообще исключающую единый центр распределения канального ресурса, но при этом способную занимать каналы связи в соответствии с предназначением каждого агента, а не хаотично? Предложим в качестве ответа на этот вопрос систему связи (рис. 3.6), которая является дальнейшим развитием системы, описанной в [15].

Рисунок 3.6 — Схема иерархической самоорганизующейся сети

Пусть каждый агент $ag \in Ag$ «знает» вектор своих признаков (дескриптор) $\mathbf{I}(ag) = (ag^1, \ldots, ag^n)$ и семейство предикатов PReq. Далее, каждый агент может иметь три роли:

1) главный центр управления связи (ГЦУС);

2) центр управления связи (ЦУС);

3) узел сети (УС).

Агенты могут менять свои роли по мере необходимости (в качестве прототипа реального средства связи с такой функциональностью могут быть предложены устройства типа «когнитивное радио», описанные в [24; 33]). Для всех агентов задана одна общая технологическая сетка каналов $F' \subset F$. Функционирование системы связи вкратце выглядит так:

Алгоритм 10 ((Функционирование системы))**.**

1. $ag_M \in Ag$ с ролью «ГЦУС» выбирает лучший по неким признакам канал $f_t \in F'$ и начинает на нем передачу своего дескриптора $d_M = \mathrm{I}(ag_M)$ в составе маячка настройки (МН), который также содержит канал ответа $f_r \in F'$. Агенты с другими ролями последовательно циклически перебирают каналы из F' и ожидают получение какого-либо дескриптора. Одновременно «ГЦУС» ждет ответов на канале $f_r \in F'$
2. Если агент ag_1 с ролью «ЦУС» или «УС» и семейством предикатов PReq в процессе перебора канала обнаружил маячка настройки с дескриптором $d_2 \in \mathcal{S}$ агента ag_2, такой, что из предикатов в соответствии с (3.14) следует, что ag_1 и ag_2 должны быть соединены, то ag_1 отправляет маячок ответа (МО) по каналу f_r, содержащий дескриптор $d_1 = \mathrm{I}(ag_1)$ и идентификатор передатчика $comm \in Comm$.
3. Когда «ГЦУС» получает достаточное количество маячков настройки, он строит по полученным из них дескрипторам агентов и соответствующим идентификаторам передатчика фрагмент графа Req и решает по имеющимся внутри него правилам PReq для него задачу распределения канального ресурса. Полученный в результате фрагмент графа связи Γ отсылается всем учтенным отправителям маячка ответа в виде пакета настроечных данных (НД). Таким образом, «ГЦУС» разделяет частотный ресурс F между агентами ag_i с ролью «ЦУС» на подмножества F_i, $\cup_i F_i = F$.
4. Агент ag_i с ролью «ЦУС» после получения ПНД настраивает в соответствии с ним свои средства и начинает на некоторой частоте $f_t^i \in F_i$ передачу маячка настройки, аналогично тому, как это ранее делал «ГЦУС».

5. Если агент ag_1 с ролью «УС» и семейством предикатов PReq в процессе перебора канала обнаружил маячок настройки от «ЦУС» с дескриптором $d_2 \in \mathcal{S}$ агента ag_2, такой, что из предикатов в соответствии с (3.14) следует, что ag_1 и ag_2 должны быть соединены, то ag_1 отправляет маячок ответа по каналу f_r, содержащий дескриптор $d_1 = \mathbf{I}(ag_1)$ и идентификатор передатчика $comm \in Comm$.
6. Когда «ЦУС» получает достаточное количество маячков, он строит по полученным из них дескрипторам агентов и соответствующим идентификаторам передатчика фрагмент графа Req и решает по имеющимся внутри него правилам PReq для него задачу распределения канального ресурса. Полученный в результате фрагмент графа связи Γ отсылается всем учтенным отправителям маячок ответа в виде пакета настроечных данных. Таким образом, «ЦУС» распределяет частотный ресурс F между агентами ag_i с ролью «УС».

Работа алгоритма может быть проиллюстрирована следующей диаграммой UML (рис. 3.7):

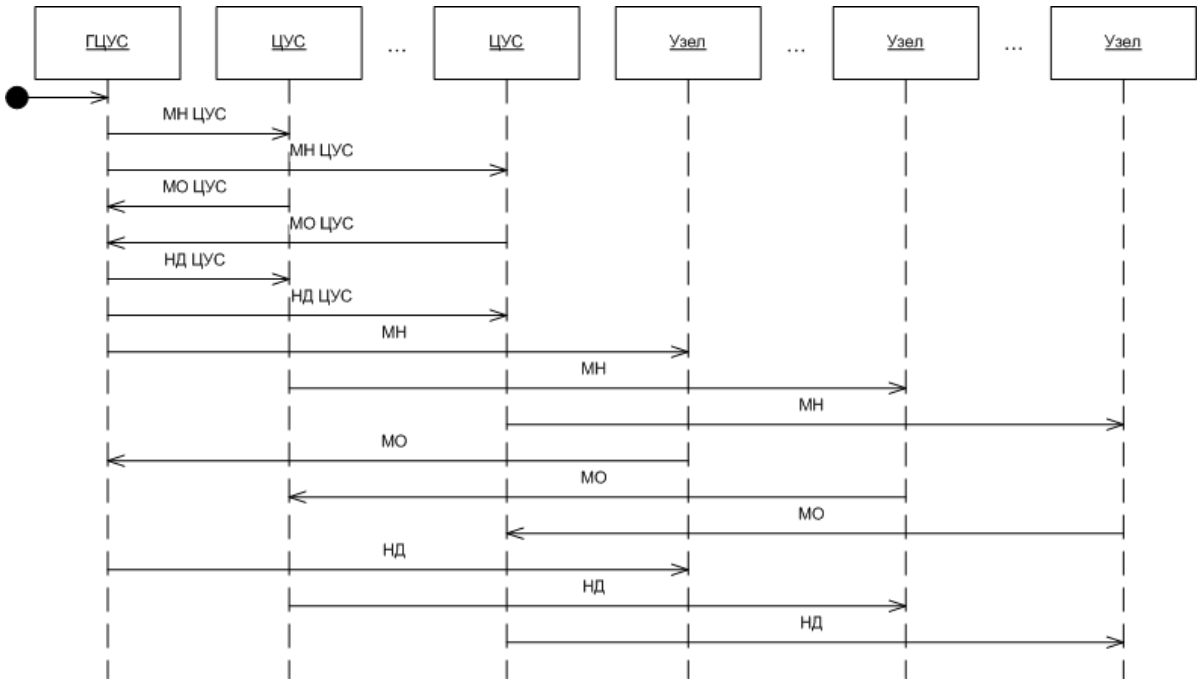


Рисунок 3.7 — Обмен маячками

Как упоминалось ранее, при необходимости закрытие и удостоверение подлинности маячков может производиться с помощью асимметричной криптосистемы типа RSA.

3.3.8 Выводы и перспективы

В разделе предложено математическое описание особого вида задачи распределения ресурсов в соответствии с заданным шаблоном. Также предложен алгоритм структурирования множества потребителей ресурсов. Надо отметить, что эта задача не обязательно связана с системами радиосвязи и распределением частот. «Каналам» могут соответствовать, например, информационные каналы разной пропускной способности между неравноправными пользователями глобальной вычислительной сети с заданными Quality of Service (QoS). Также «каналами» могут быть варианты прокладки железнодорожных путей между населенными пунктами при заданной матрице связности населенных пунктов и с ограничением на общую длину рельсов.

Выбор приведенной в разделе постановки задачи, использующей понятийный аппарат теории графов, связан с тем, что автор использовал именно такие конструкции при построении имитационной модели движения и связи иерархически организованных агентов, в том числе и для визуализации изменений состояния системы связи. Вышеуказанная модель основана на клеточном автомате, симулирующем перемещение групп агентов по пересеченной местности [2]. При этом вершины графа Г в каждый момент времени поставлены в соответствие клеткам автомата, в которых находятся агенты. В процессе движения агенты теряют друг друга из видимости из-за особенностей проходимого ландшафта, в результате чего возникает необходимость в перестройке графа связи Г.

Дальнейшее развитие может лежать в направлении совершенствования мер сходства графов потребностей в связи, например, в представлении графа потребностей связи в виде модели состояний и переходов (state transition model) и в замене изоморфизма подграфов в определении сходства подграфов на отношения взаимной симуляции (simulation) или бисимуляции (bisimulation) и в переходе при задании структуры системы связи к сущностям типа коданных (codata).

3.4 Моделирование начального распределения частот

3.4.1 Введение

В настоящее время число роботизированных систем, состоящих из большого количества автономных мобильных агентов (наземных роботов, БПЛА и т. д.) снабженных сетью беспроводной радиосвязи, постоянно увеличивается. Агенты взаимодействуют друг с другом для выполнения общей задачи, например, для ликвидации чрезвычайных ситуаций, обнаружения вторжений в охраняемую зону, разведки или сетей в тех областях, где трудно организовать сеть связи обычными способами [74] и так далее.

Развертывание или восстановление системы связи с помощью людей для таких агентов может быть затруднено – например, в чрезвычайной ситуации это может быть слишком опасно. По этой причине, необходимо предоставить самим агентам всю логику, необходимую для поддержания сети связи, изначально устанавливая только основные принципы взаимодействия агентов. Эта проблема подробно исследуется в [16], и ниже мы вводим простую модель такой самоорганизации. Вкратце, агент периодически проверяет радиоканалы для обнаружения других агентов или останавливается на одном из каналов и передает последовательность маяков, чтобы их могли обнаружить другие агенты.

Проблема, которую мы рассматриваем, является одной из проблем *ad hoc* сетей Cognitive Radio. В статье [131] описывается динамический доступ к спектру (DSA), который использует рассуждения о политике спектра для определения разрешенных частот, запроса периодов чувствительности на этих частотах, классификации результатов от событий обнаружения, а затем предоставления списка разрешенных частот для использования в частотном присвоении, в статьях [140;141] описаны различные физические и информационные аспекты доступа к динамическому спектру. Согласно этим работам, существуют два основных подхода к DSA: динамическое распределение спектра и оппортунистический доступ к спектру. Динамическое распределение спектра использует статистику временного и пространственного трафика и нацелено на повышение эффективности использования спектра за счет временного и пространственного

разделения спектра между сосуществующими радиослужбами. В отличие от динамического распределения спектра, который использует статистику занятости спектра, оппортунистический доступ к спектру использует мгновенную доступность спектра, открывая лицензионный спектр для вторичных пользователей. Идея состоит в том, чтобы позволить второстепенным пользователям идентифицировать доступные ресурсы спектра и сообщать при получении возможности таким образом, чтобы ограничивать уровень помех, воспринимаемых первичными пользователями.

Предлагаемый в настоящем подразделе алгоритм был разработан в качестве попытки ответить на вопрос: возможно ли создать полностью автономную сеть Cognitive Radio, без базовых станций или начального распределения спектра, или же координация с базовой станции необходима для распределения радиоканалов между агентами за конечное число итераций?

Целью настоящего подраздела является попытка разъединить самоорганизацию сети и ее радиофизическую основу и изучить самоорганизацию как чисто математическую задачу сегрегации. Динамическая сетевая организация, по своей сути, представляет собой модель сегрегации по типу модели сегрегации Шеллинга [142]. Клеточный автомат Шеллинга (рис. 3.8) представляет собой сетку, где каждая клетка соответствует человеку. Человек может иметь один из нескольких «цветов», который символизирует его принадлежность к определенной социальной группе. Цвета распределяются случайным образом по сетке, а также имеется некоторое количество незанятых ячеек. Каждый человек хочет, чтобы определенная часть людей вокруг него была похожа на него. Если количество соседних ячеек одного цвета ниже заданного порога, то человек переходит в следующую свободную ячейку, в противном случае он остается на месте. Однако, в отличие от двумерной классической модели Шеллинга, предложенный автомат практически одномерен: координата n агента нам нужна только для упрощения описания динамики автомата.

3.4.2 Клеточно-автоматная модель

Обозначим как $\mathcal{ID} \subset \mathbb{Z}$, $0 \in \mathcal{ID}$ множество возможных идентификаторов агентов, как $\mathcal{Q} \subset \mathbb{N}$ – множество качеств каналов.

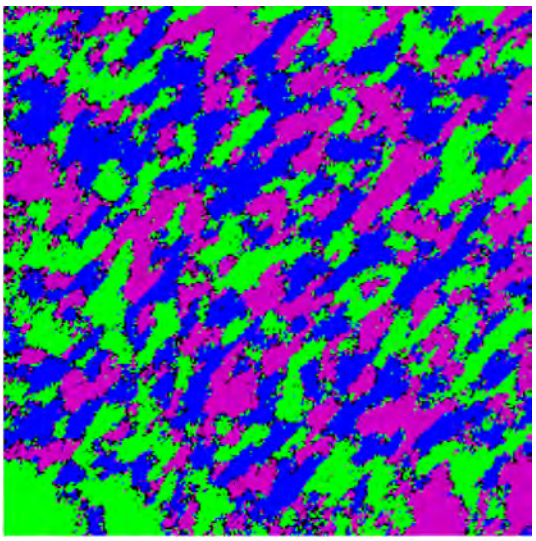


Рисунок 3.8 — Модель сегрегации Шеллинга

Определение 18. *Определим множество каналов как односвязное множество* $\mathcal{C} \subseteq \mathbb{Z}^2$*, где каждая пара* $(n,f) \in \mathcal{C}$ *соответствует клетке* $(q_{nf}, id_{nf}) \in \mathcal{Q} \times \mathcal{ID}$ *и* $q_{kf} = q_{lf}$ *для каждого* (k,f), $(l,f) \in \mathcal{C}$*. Предположим, что если* $id_{nf} = 0$*, то в клетке* (n,f) *нет агента. Таким образом,* $f-$*я колонка клеток* $\{(i,f)\}$ *соответствует* $f-$*му каналу. Количество* N_f *клеток в* $f-$*й колонке соответствует максимально возможному количеству агентов, использующих* $f-$*й канал.*

Определение 19. *Назовем агентом нижеследующий вектор*

$$\mathbf{ag} = (id_{ag}, col_{ag}, f_{self}, n_{self}; f_{target}\mathrm{r}_{ag}, \mathrm{Q}_{ag}; \mathbf{sensed}; t),$$

где $id_{ag} \in \mathcal{ID}$ *– уникальный идентификатор агента,* f_{self} *– номер текущего канала,* n_{self} *– положение агента на текущем канале* f_{self}*,* f_{target} *– номер «целевого» канала. Далее,* $color_{ag} \in \mathcal{COL}$ *– цвет агента,* $\mathrm{r}_{ag} : \mathcal{COL} \times \mathcal{ID} \to \{0,1\}$ *– предикат группировки,* $\mathrm{Q}_{ag} : \mathcal{Q} \to [0,1]$ *– функция предпочтений агента,* t *– общее время функционирования агентов в тактах автомата. Множество всех агентов обозначим как* $\mathcal{AG}$*. Вектор* **sensed** *содержит знания агента об окружающей среде(см. ниже).*

Определим предикат $\mathrm{R} : \mathcal{AG} \times \mathcal{AG} \to \{0,1\}$ следующим образом

$$\mathrm{R}(\mathbf{ag}_1, \mathbf{ag}_2) = \mathrm{r}_{ag_1}(col_{ag_2}, id_{ag_2}), \quad \mathbf{ag}_1, \mathbf{ag}_2 \in \mathcal{AG}.$$

Если агенты $\mathbf{ag}_1$, $\mathbf{ag}_2 \in \mathcal{AG}$ хотят занять один и тот же канал, то $\mathrm{R}(\mathbf{ag}_1,\mathbf{ag}_2) = 1$, иначе $\mathrm{R}(\mathbf{ag}_1,\mathbf{ag}_2) = 0$. Очевидно, r_{ag_1} должно быть определено так, что $\mathrm{r}_{ag_1}(ag_2) = \mathrm{r}_{ag_2}(ag_1)$ для любых $\mathbf{ag}_1$, $\mathbf{ag}_2 \in \mathcal{AG}$. Отметим, что R определяет граф связи для $\mathcal{AG}$.

Определение 20. *Пусть p, x – функции. Мы будем писать $p \sim x$, если существует такая неотрицательная возрастающая функция φ, что $p(x) = \varphi(x)$.*

Для упрощения, предположим, что $\mathcal{C} = \{(f,n)|f = \overline{1,F_{\max}}, n = \overline{1,N_{\max}}\}$. Также обозначим $q_{nf} = q_f$, множество агентов на канале f (т.е. с $f_{self} = f$) как $\mathcal{AG}_f \subseteq \mathcal{AG}$, $\mathcal{AG}^{\mathbf{ag}} = \{\tilde{\mathbf{ag}} \in \mathcal{AG}|\mathrm{R}(\mathbf{ag}, \tilde{\mathbf{ag}}) = 1\}$ – множество агентов из той же сети, что и агент **ag** и $\mathcal{AG}_f^{\mathbf{ag}} = \{\tilde{\mathbf{ag}} \in \mathcal{AG}_f|\mathrm{R}(\mathbf{ag}, \tilde{\mathbf{ag}}) = 1\}$ – множество агентов на канале f из той же сети, что и агент **ag**.

Обнаружение агентов может быть неуспешным. По этой причине, введем функцию $\mathrm{D} : 2^{\mathcal{AG}} \to 2^{\mathcal{AG}}$. Агент $\mathbf{ag} \in \mathcal{AG}_f$ попадает в $\mathrm{D}(\mathcal{AG}_f)$ с вероятностью $p_1 \sim q_f$.

Наш автомат функционирует в дискретном времени. Тактовое поведение агента может быть описано с помощью следующего алгоритма:

1. **Инициализация**. Агент ag сканирует все каналы в $\mathcal{C}$ и случайно выбирает (n,f) с вероятностью $p_0(n,f) \sim \mathrm{Q}_{ag}(q_{nf})$ и такую что клетка (n,f) не является уже занятой.
2. **Обнаружение**. Агент **ag** изучает канал f. Агент устанавливает
$$f_{self} := f, \quad n_{self} := n,$$
$$\mathbf{sensed}[f] := \{q_f, \mathrm{D}(\mathcal{AG}_f)\}.$$
3. Если $f_{self} = F_{\max}$, то агент выбирает канал $\tilde{f}$ из **sensed** с вероятностью $p_0(n,f)$. Например, p_0 может быть определено так, что агент выбирает канал с наилучшим качеством. Агент устанавливает
$$f_{target} := \tilde{f}.$$
4. **Принятие решения**. Если $|\mathrm{D}(\mathcal{AG}_f^{\mathbf{ag}})| = |\mathcal{AG}^{\mathbf{ag}}|$,тогда переходим к шагу 6, иначе если $|\mathrm{D}(\mathcal{AG}_f^{\mathbf{ag}})| > |\mathcal{AG}^{\mathbf{ag}}|/\alpha_0$, $\alpha_0 > 1$, переходим к шагу 5, иначе переходим к шагу 4.
5. **Смена канала**. Агент **ag** выбирает новый канал
$$f_{self} := \begin{cases} f_{self} + dir, & 1 \leqslant f_{self} + dir \leqslant F_{\max}, \\ F_{\max}, & f_{self} + dir = 0, \\ 1, & f_{self} + dir > F_{\max}. \end{cases}$$
Дополнительно, агент выбирает произвольную координату n, такую что клетка (f, n) не является занятой. Если такой клетки не существует, то

агент ждет данный ход и повторяет попытку. Если $f_{self} = f_{target}$, то переходим к шагу 5, иначе переходим к шагу 1.

6. Агент ожидает время

$$t_{wait} = \alpha_1 F_{\max} + \alpha_2 \tau(|\mathrm{D}(\mathcal{AG}_f^{\mathbf{ag}})|, |\mathcal{AG}^{\mathbf{ag}}|, |\mathrm{D}(\mathcal{AG}_f)|), \quad \alpha_1 \geqslant 0, \alpha_2 \geqslant 0, \tag{3.18}$$

$$\tau(x,y,z) \sim \min\left\{\frac{x}{y}, \frac{x}{z}\right\}, \tag{3.19}$$

ходов и выбирает dir случайно из множества $\{-1,1\}$. Переходим к шагу 1.

7. **Ожидание**.

Основная проблема заключается в том, чтобы найти параметры модели α_i, $i = \overline{1,3}$ и функцию τ, чтобы процесс сегрегации завершился в разумные сроки. Следует отметить, что неудачный выбор параметров приведет к тому, что алгоритм не сходится вообще.

Физически агенты получают информацию о других агентах и состоянии каналов, сканируя спектр и обмениваясь маяками, как описано в подразделе 3.3.7 и [16]. Один такт клеточного автомата соответствует полному циклу обмена пилот-сигналами и маяками. Подробное техническое решение, соответствующее предлагаемой модели, описано в патенте «Средство передачи данных телекоммуникационной сети и телекоммуникационная сеть» №RU 2 549 120.

Пример конечного состояния автомата показан на рис. 3.9. Более светлый тон ячеек на изображении соответствует лучшему качеству соответствующих каналов. Мы можем видеть сеть $Net_1 = \{ag_i | i = \overline{1,4}\}$, образованную на канале f_1, и $Net_2 = \{ag_i | i = \overline{5,6}\}$, образованную на канале f_2.

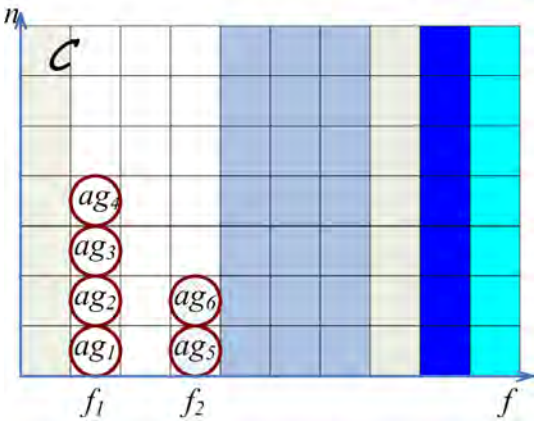


Рисунок 3.9 — Финальная конфигурация автомата

3.4.3 Вычислительный эксперимент

Выполним имитационное моделирование предложенного алгоритма с помощью программной среды «Психоход» [6]. Для этой цели был разработан особый режим работы симулятора (см. рис. 3.10).

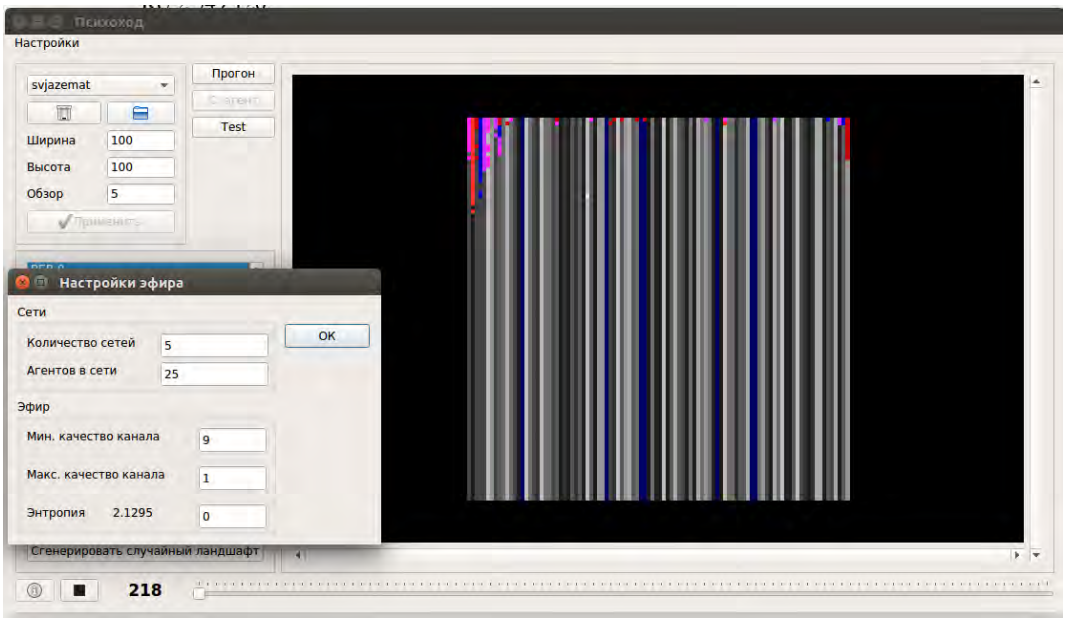


Рисунок 3.10 — Программа «Психоход» в процессе сегрегации агентов

Определим $p_1 = 1$, а предикат группировки – как

$$\mathrm{R}(\mathbf{ag}_1, \mathbf{ag}_2) = \begin{cases} 1, col_{ag_1} = col_{ag_2}, \\ 0, col_{ag_1} \neq col_{ag_2}. \end{cases}$$

Рис. 3.11 содержит результаты моделирования для 5 сетей, 50 агентов в каждой сети, и для

$$\alpha_1 = F_{\max} + 1, \quad \alpha_2 = (F_{\max} + 1)\beta, \quad \tau(x,y,z) = \min\left\{\frac{x}{y}, \frac{x}{z}\right\}.$$

Ординаты соответствуют среднему дискретному времени $\overline{T}$, абсциссы соответствуют α_0. Качества каналов $q = \overline{1,9}$ были распределены равномерно по $F_{\max} = 100$ каналам, было выполнено по 100 экспериментов для каждой точки.

Симуляция прекращалась, когда процесс сегрегации был завершен или когда $T > 30000$. Обведенные окружностями точки соответствуют случаю, когда для всех 100 экспериментов сегрегация была завершена менее чем за 30000 тактов. Для прочих точек, процесс сегрегации был завершен в 79–99% случаев. Рис.

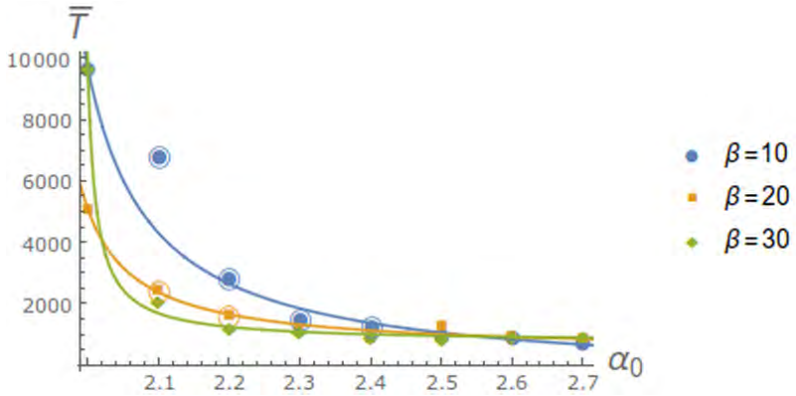


Рисунок 3.11 — Среднее время завершения сегрегации сети (50 агентов в сети, 5 сетей)

3.12 представляет гистограммы времени завершения сегрегации для разных значений α_0 и β. Мы использовали функцию

$$\overline{T}(\alpha_0) = a + \frac{b}{\alpha_0 - c}, \quad c > 0, b > 0$$

для приближения зависимости по точкам данных.

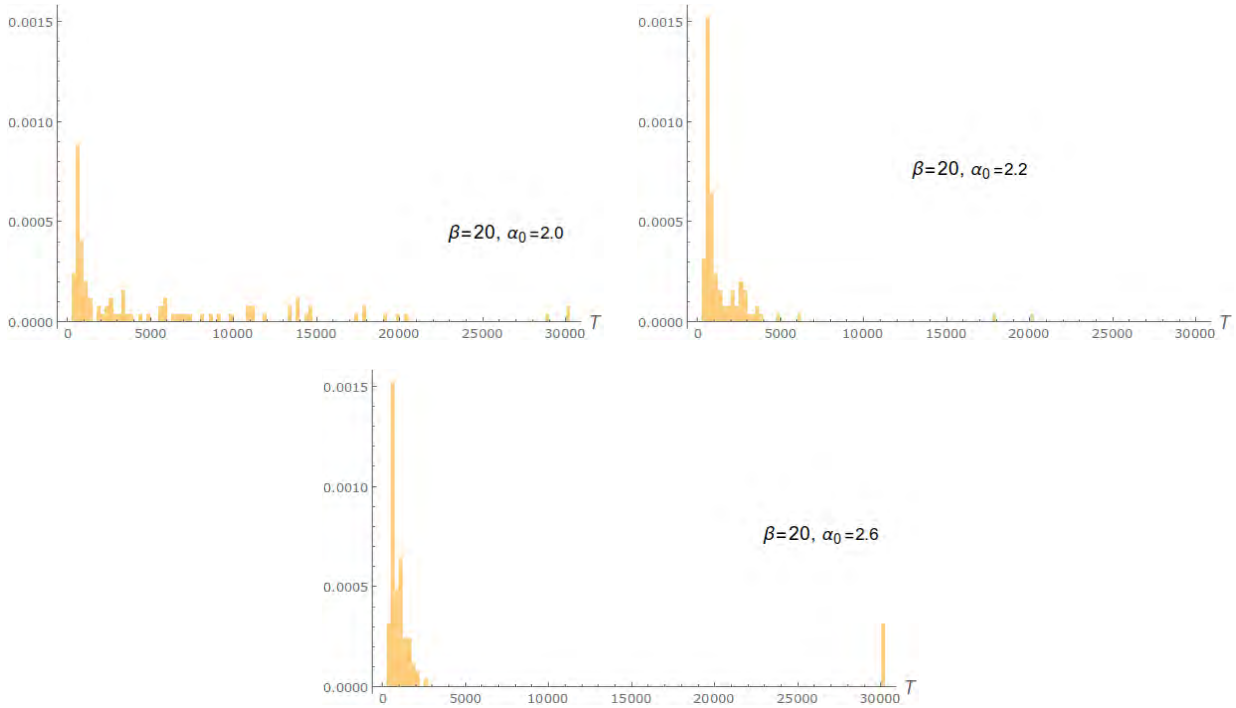


Рисунок 3.12 — Гистограммы времен сегрегации

Далее, зафиксируем $\alpha_0 = 2.2$, $\beta = 30$ и выполним эксперименты с различным количеством сетей и агентов в каждой сети, как описано в таблице 7. Отметим, что сегрегация считается неудачной, если она занимает по времени более 30000 ходов.

Таблица 7 — Процент неудачных сегрегаций

Агенты в сети	10	20	30	40	50	60	70	80	90	100
Количество неудач, пять сетей	37	15	3	4	1	0	0	5	9	30
Количество неудач, восемь сетей	97	75	59	33	22	2	1	18	29	100

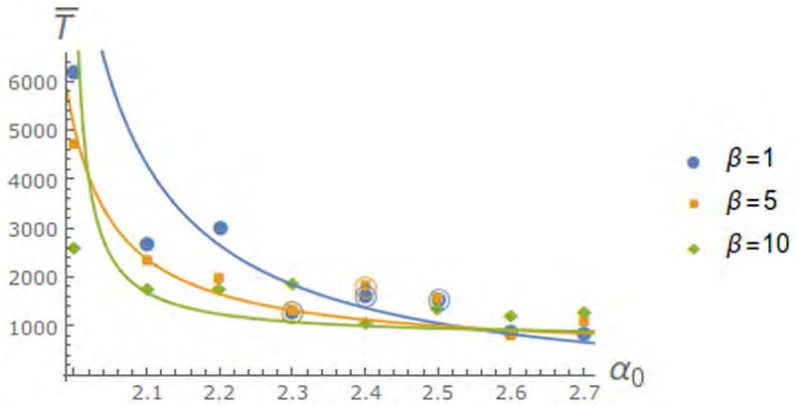


Рисунок 3.13 — Среднее время завершения сегрегации сети (25 агентов в сети, 5 сетей)

Таким образом, мы должны изменять α_0 и β пропорционально количеству агентов. Например, если у нас есть 25 агентов в каждой из пяти сетей, нам нужно уменьшить β примерно в четыре раза и немного увеличить α_0 (см. рис. 3.13), чтобы минимизировать среднее время сегрегации.

Результаты численного эксперимента сразу указывают на улучшение ранее предложенного алгоритма. Мы должны добавить следующий шаг:

2.5. Проверка прошедшего времени. Если $t > T_r$, то изменим α_i, $i = \overline{0,2}$ на произвольные малые величины.

Кроме того, мы можем запомнить все изменения параметров алгоритма и постепенно выбирать оптимальные. Дальнейшим усовершенствованием может быть разделение набора каналов на подмножества, так что агенты из одной сети будут стремиться искать каналы в подмножестве каналов, связанные с номером их сети.

Предлагаемый клеточный автомат близок к так называемой модели Шеллинга II типа. Работы [143; 144], которые вводят непрерывные по времени динамические системы Шеллинга, описывают модель Шеллинга II следующим образом.

Пусть популяция распределено по непересекающимся множествам, например, по разным районам города. Население делится на «классы». Пусть X и Y

обозначают два класса, населяющих пространство, которое разделено на m областей, обозначаемых как A_i, $i = 1,2,\ldots,m$. Кроме того, пусть X_i и Y_i обозначают общее количество классов X и Y в областях A_i, $|X|$ и $|Y|$ – общее количество каждого типа в популяции, и $N = |X| + |Y|$ – общая численность популяции. Толерантности агентов одного класса к агентам другого класса присваиваются заданному классу в заданной области через «расписание толерантностей». Эта функция описывает максимальное количество агентов класса X, которое терпит $R(X)X_i$ агентов класса Y в области A_i, причем $R(X)$ обязательно монотонно убывает. Также предполагается, что нет нижней границы толерантности, т. е. ни одна популяция одного класса не настаивает на наличии другого класса в занимаемой ей области. Простейшая, однообластная непрерывная динамическая система Шеллинга имеет следующий вид:

$$\frac{dx}{dt} = [xR_X(x) - y]x, \tag{3.20}$$

$$\frac{dy}{dt} = [yR_Y(y) - x]y, \tag{3.21}$$

где x, y – плотности популяций X, Y, $|X| = k|Y|$, $p > 0$, и R_X, R_Y – монотонно убывающие функции расписания толерантности. Здесь, например,

$$R_X(x) = a(1-x)^p, \tag{3.22}$$

$$R_Y(y) = b(1-ky)^p, \tag{3.23}$$

$a > 0$, $b > 0$, $p > 0$, $k = |X|/|Y|$.

Было обнаружено, что поведение описываемого автомата в некоторых случаях может быть описано моделью типа (3.20), (3.21). Обозначим как

$$x(t) = X_f(t)/|X(t)|$$

плотность агентов с сети 1 на канале связи f. Также обозначим как

$$y(t) = Y_f(t)/|Y(t)|$$

плотность агентов, относящихся ко всем остальным сетям на канале связи f.

Так как мы имеем 50 агентов в каждой сети и 5 сетей, $k = |X|/|Y| = 1/4$, $x(18) = 1/50$, $y(18) = 1/200$. Для начала, выберем $p = 1.7$, $a = 1$, $b = 1/2$.

Мы можем модифицировать (3.20), (3.21) способом, описанным в [143], используя экспоненциальное расписание

$$R_X(x) = \frac{e^{-6x} - e^{-6}}{1 - e^{-6}}, \tag{3.24}$$

$$R_Y(y) = \frac{e^{-6y/k} - e^{-6/k}}{2(1 - e^{-6/k})}, \tag{3.25}$$

для получения лучшей аппроксимации результатов вычислительного эксперимента. Мы можем видеть сравнение экспериментальных точек с решениями динамической системы Шеллинга на рис. 3.14, слева. Циркулярные и ромбовидные маркеры соответствуют экспериментальным значениям x и y. Черная пунктирная линия соответствует линейному расписанию (3.22), (3.23), сплошные линии соответствуют экспоненциальному расписанию (3.24), (3.25).

Интересно, что существует такое начальное условие, что возникает колебательное поведение X_f. Это случай примерно одинаковых начальных количеств агентов разных сетей на канале f (см. рис. 3.14, справа). К сожалению, модель (3.20), (3.21) не может объяснить этот случай в целом, и мы должны использовать более сложную модель сегрегации по более чем двум областям, но можно моделировать колебания путем умножения (3.24), (3.25) на периодическую функцию. Вышеупомянутая ситуация влечет за собой потенциальную невозможность самоорганизации сети в разумные сроки.

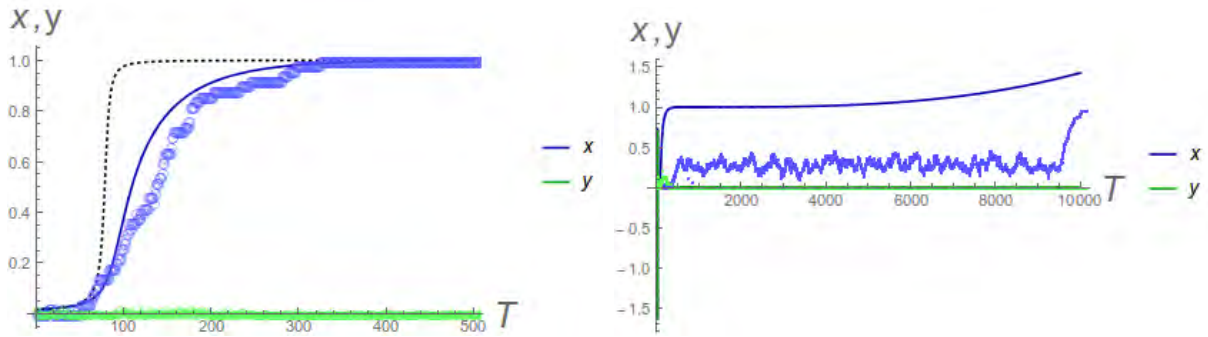


Рисунок 3.14 — Графики результатов численного эксперимента по сравнению с решениями динамической системы Шеллинга

3.4.4 Изменение прямой видимости агентов, движущихся по пересеченной местности

В качестве примера использования модели системы связи продемонстрируем зависимости возникновения обрывов соединений, требующих прямой видимости, от времени и степени неоднородности ландшафта высот $\mathcal{L}$.

Было создано по 100 случайных квадратных ландшафтов высот 50 на 50 клеток для каждого заданного числа количества препятствий $N_{obst} = 20, 40, 80, 120$ с конфигурационной энтропией $S = 1.68, 1.85, 2.02, 2.11$ соответственно, $l = 9$, в качестве классов выступали «высоты» клеток в диапазоне от 0 до 9. По каждому ландшафту с одной стороны на противоположную перемещалось 6 агентов с учетом правил (5.1), (5.2).

Обозначим как $V(t) = (v_{ij}(t))_{i,j=\overline{1,6}}$ матрицу прямой видимости агентов, $v_{ij}(t) = 0$, если агенты i и j «видят» друг друга по прямой линии ($\mathfrak{h}(b,e) = 0$ в смысле равенства (5.3)) на ходу t и $v_{ij}(t) = 1$ в противном случае.

Введем величину

$$\nu(t) = \sum_{i=1}^{t} \|V(t) - V(t-1)\|_1, \quad t > 0,$$

где $\|V\|_1 = \sum_{i,j=1}^{6} |v_{ij}|$. Величина $\nu(t)$ соответствует количеству изменений матрицы прямой видимости агентов к ходу t. Эту величину можно использовать, например, для оценки количества обрывов и восстановлений высокоскоростных каналов, например, Wi-Fi, которым необходима прямая видимость, во время движения агентов. Далее, вычисляя среднее значение $\nu(t)$ по всем экспериментам для одного значения N_{obst}, получаем зависимость, показанную на рис. 3.15 слева.

Для сравнения, аналогичная зависимость для ландшафта, у которого клетки с разными высотами разбросаны равномерно, а конфигурационная энтропия равна 1.68, что соответствует $N_{obst} = 20$ у ландшафтов ранее описанного типа, показана на рис. 3.15 справа. Следует отметить, что если мы будем исследовать зависимость количества обрывов (или восстановлений) связи к моменту времени t, то эта зависимость будет иметь также линейный вид.

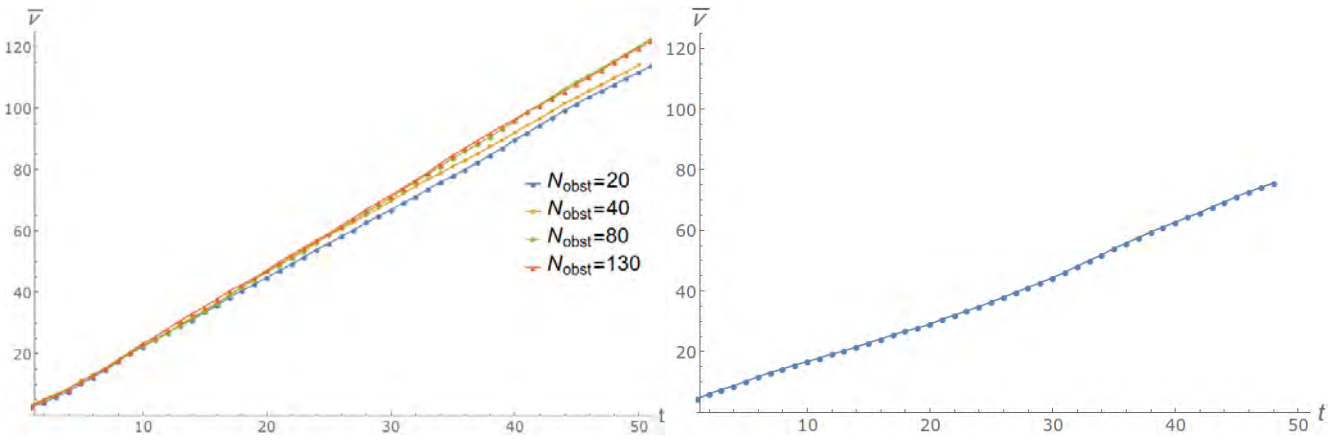


Рисунок 3.15 — Изменение матрицы прямой видимости агентов со временем для различных ландшафтов

3.5 Алгоритмы организации защищенной сети связи

3.5.1 Система автоматической настройки беспроводной сети радиостанций

Рассматриваемая радиосеть состоит из $nodenum$ радиостанций (РСт) и одной радиостанции – центра управления связью (ЦУС), которые в целях настройки могут работать на $freqnum$ частотных каналах ($freqnum > 1$), причем конкретный номер канала заранее неизвестен. Предположим, что используемые радиочастотные каналы пронумерованы числами из множества $F = \{1, \ldots, freqnum\}$. Все каналы должны допускать одновременную работу на каждом из них. Будем считать, что в период настройки все радиостанции и ЦУС передают v бит в секунду. Предполагается также, что заранее создана ключевая пара из секретного ключа, который хранится в ЦУС, и открытого ключа, который распространен по всем радиостанциям сети. Обозначим время задержки в канале τ_d, а время перестройки с одного канала на другой – τ_c.

В качестве шифрования используется асимметричное шифрование, например RSA [145]. Закрытый ключ должен будет содержаться лишь в одной радиостанции, находящейся в контролируемой зоне, а все другие радиостанции должны будут иметь только открытые ключи, сгенерированные на базе закрытого ключа. Распространение сеансовых ключей между радиостанциями и ЦУС происходит с помощью шифрования сеансового ключа открытым ключом ЦУС.

3.5.2 Предлагаемый алгоритм настройки радиосети

Рассмотрим возможность одновременной организации защищенной сети в защищенном режиме посредством обмена по радиоканалу сигналами-маячками, содержащими информацию для аутентификации, проверки целостности сообщения и асимметричного шифрования.

Описание этапов предлагаемого алгоритма настройки радиостанций

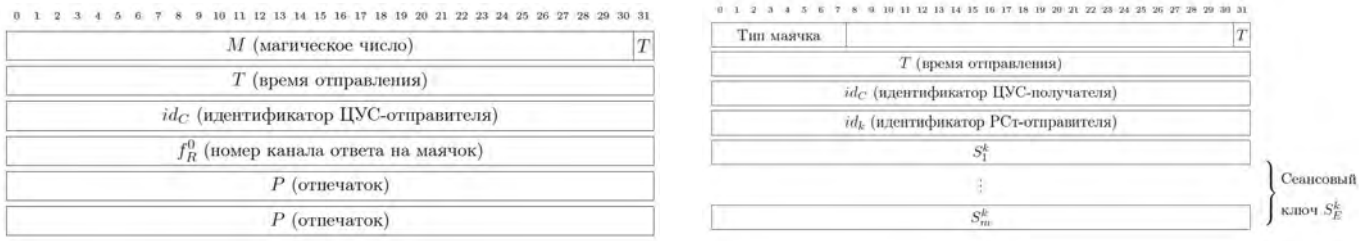
Предлагаемый алгоритм настройки радиостанций состоит из трех этапов.

1-й этап алгоритма. В начальные моменты времени t_0^k и в ЦУС (для которого $k = 0$), и в радиостанциях (для которых k – номер радиостанции, $1 \leqslant k \leqslant nodenum$) происходит процесс оценки каналов из сетки F (0-й этап алгоритма настройки сети):

1. В момент времени t_0^k ЦУС или радиостанция устанавливает $i := 1$.
2. ЦУС или радиостанция дает своему приемнику команду перейти на частоту f_i с номером $i \in F$.
3. Приемник ЦУС останавливается на частоте f_i в течение $\tau_c + \tau_d + \tau_m$ секунд, где τ_m – время, необходимое для вычисления отношения сигнал/шум в канале.
4. ЦУС или радиостанция вычисляет для частоты f_i соотношение SNR_i. Если во время измерения нет возможности использовать тестовые сигналы, SNR_i определяется с учетом мощности передатчика ЦУС или радиостанции и замеренного уровня шума.
5. Если $i \leqslant freqnum$, то $i := i + 1$, и переходим к шагу 2; если иное, то к шагу 6.
6. Каналы сортируются по SNR_i. Из числа каналов с лучшим отношением сигнал/шум (т.е. таких, что $SNR_j = \max_{0 \leqslant i \leqslant freqnum} SNR_i$) случайно выбираются два разных канала f_T^k и f_R^k. Если канал f_j с максимальным SNR_j только один, выбирают $f_T^k := f_j$, а канал f_R^k выбирают из числа каналов с отношением сигнал/шум, ближайшим к SNR_j.

2-й этап алгоритма. После завершения в момент t_1^0 времени 1-го этапа алгоритма настройки в ЦУС начинается следующий процесс (2-й этап алгоритма настройки):

1. ЦУС настраивает свой передатчик на найденный на 1-м этапе канал f_T^0.
2. ЦУС настраивает свой приемник на найденный на 1-м этапе канал f_R^0.
3. ЦУС начинает непрерывно передавать на приведенном канале маячок настройки (рис. 3.16а).
4. Если выбранный канал f_T^0 становится предположительно непригодным для работы (например, не приходят маячки ответа, см. ниже), повторяется 1-й этап алгоритма настройки.



а) б)

Рисунок 3.16 — Формат маячка настройки и маячка ответа

В маячке настройки M – это уникальная битовая последовательность («магическое число»), известная всем радиостанциям в сети; T – время отправления маячка, в миллисекундах, отсчет ведется с начала текущих суток; P – криптографический хеш маячка настройки (электронная цифровая подпись ЦУС по типу RSA), исключающий подделку маячка с целью дезорганизации сети или введения ложных узлов. На рис. 3.16а изображен 24-байтный маячок настройки, в общем случае длину маячка настройки будем обозначать L_T.

В k-й радиостанции 2-й этап алгоритма настройки начнется во время t_1^k. Он заключается в следующем:

1. Устанавливаем $i := j$, $f_j := f_T^k$, где f_T^k с номером j обнаружен на 1-м этапе.
2. РСт дает команду приемнику переключиться на канал f_j.
3. В течение времени $\tau_c + \tau_d + 3L_T/v$ приемник задерживается на канале, обозначенном в пункте 1, пытаясь обнаружить в принятых битах «магическую» последовательность M. Если канал оказывается недоступен (например, поражен помехой), на некоторое случайное время для ускорения перебора каналов можно исключать его из перебираемых.

4. Если последовательность M обнаружена, приемник считывает идущий за ней маячок настройки, проверяет его целостность и неподдельность с помощью открытого ключа ЦУС и отпечатка P. Если проверка удалась и ранее не было получено маячка настройки с T больше, чем у полученного, переходим к шагу 5; иначе – к шагу 3. Если повторная проверка не удалась, переходим к шагу 10. Если был получен маячок настройки с T больше, чем у полученного, отбрасываем полученный маячок настройки.
5. РСт настраивает свой передатчик на канал f_0^R из маячка настройки.
6. Если канал f_0^R свободен, переходим к шагу 7; иначе – к шагу 8 (реализуем CSMA/CA).
7. РСт высылает маячок ответа (рис. 3.16б) и переходит к шагу 9.
8. РСт ждет время $L_R/v + \xi$, где L_R – длина маячка ответа, $\xi \leqslant nL_R/v$, $n \in \{1, 2, 3\}$ – случайная величина, и переходит к шагу 6.
9. Конец 2-го этапа настройки.
10. Пусть $f_k \in F$ – следующий сразу после f_i по отношению сигнал/шум канал. Устанавливаем $i := k$ и переходим к шагу 2.

На рис. 3.16б id_C – идентификатор ЦУС-получателя, в ответ на маячок настройки которого отправляется данный маячок ответа; $S_E^k = S_1^k || \ldots || S_m^k$ – сеансовый симметричный ключ S^k, зашифрованный открытым ключом ЦУС-отправителя маячка настройки.

Следует отметить, что, если помеховая обстановка у ЦУС и у k-й радиостанции одинакова, время 2-го этапа алгоритма настройки существенно сокращается, так как с большой вероятностью f_T^k совпадет с f_T^0.

3-й этап алгоритма. На ЦУС 3-й этап алгоритма настройки начинается при приеме маячков ответа от радиостанции. Предполагается, что на ЦУС присутствует полный набор каким-либо образом рассчитанных настроечных данных (частот, мощностей, видов модуляций, ключей псевдослучайной перестройки рабочей частоты (ППРЧ) и пр.) для всех станций сети связи, причем эти данные могут быть получены с учетом предыдущих этапов алгоритма настройки (как минимум, могут быть исключены выявленные на 1-м этапе частоты, заведомо непригодные по отношению сигнал/шум) или же быть совершенно новыми:

1. При получении и успешной расшифровке маячка ответа от k-й радиостанции ЦУС готовит для k-й станции настроечные

данные и шифрует их ключом S_E^k, полученным от маячка ответа.

2. Когда получены маячки ответа от некоторого порогового количества N_f радиостанций, ЦУС начинает рассылку пакетов с зашифрованными настроечными данными по каналу f_T^0.

Для повышения скрытности и надежности системы связи есть возможность дополнить алгоритм настройки рассылкой ЦУС по каналу f_T^0 промежуточного маячка настроечных данных (рис. 3.17а), содержащего набор частот, на которых будет осуществляться передача настроечных данных в формате ППРЧ.

а) б)

Рисунок 3.17 — Формат промежуточного маячка настроечных данных (а) и маячка завершения (б)

3-й этап алгоритма в радиостанции начинается с получения пакета с настроечными данными или с промежуточного маячка настроечных данных (рис. 3.17б). После успешного применения полученных настроечных данных (в режиме ППРЧ или без такового) радиостанция высылает в адрес ЦУС id_C маячок завершения (см. рис. 3.17б) по каналу f_R^0, используя механизм CSMA/CA, настраивается этими данными и начинает игнорировать все ранее описанные маячки.

На рис. 3.17б буква $I = I_1||I_2$ обозначает имитовставку, сгенерированную на основе маячка завершения и сеансового ключа S^k.

ЦУС после получения и успешной проверки имитовставки от некоторого порогового количества радиостанций N_f маячков подтверждения или по истечении некоторого времени завершает алгоритм настройки. Если маячки подтверждения или маячки ответа получены не от всех радиостанций, ЦУС может периодически повторять процесс настройки с 1-го этапа, исключая из множества перебираемых каналов настройки F те, которые входят в настроечные данные, или же те, которые электромагнитно не совместимы с каналами, входящими в состав настроечных данных.

Процесс настройки в виде диаграммы последовательностей UML изображен на рис. 3.18.

Рисунок 3.18 — Последовательность настройки радиосети специального назначения

3.5.3 Модель системы настройки радиосети специального назначения

Определим пространство сигналов B как множество ступенчатых функций

$$s : [0,T] \to \{-1,0,1,2\} = Z4.$$

Обозначим множество $\{-1,0\} = -\mathbb{Z}_2$ как $Z2$, множество $\{0,\ 1,\ 2\}$ как $\mathbb{Z}_3$. Будем отождествлять функцию из B с битовой последовательностью (сигналом) в некотором дискретном канале $f \in F$; тогда $s(t) = 1$, $s \in S$, если в момент $t \in [0,T]$ передается бит «0», $s(t) = 2$, если в момент $t \in [0,T]$ передается бит «1» и $s(t) = 0$, если информация не передается. Далее $e(t) = -1$, $e \in B$, если в момент времени t присутствует помеха, и $e(t) = 0$ – в противоположном случае. Будем считать, что в один момент времени в одном канале может быть только один содержащий информацию сигнал.

Модель среды распространения будет представлять собой дискретный преобразователь $Channel$ (типа автомата Мили), входной алфавит которого

$$X_{Ch} = \{((b^0,f^0),\ldots,(b^{nodenum},f^{nodenum})),(e^1,\ldots,e^{freqnum})|$$
$$b^k \in \mathbb{Z}_3, f^k \in F, e^j \in -\mathbb{Z}_2, 1 \leqslant k \leqslant nodenum, 1 \leqslant j \leqslant freqnum\}.$$

Иначе говоря, у $Channel$ есть $nodenum + 1$ входов для принятия пары из сигнала b^k и канала f_k, на котором передается сигнал, для ЦУС и для каждой из радиостанций сети и $freqnum$ входов для принятия ошибки e^j на каждом из каналов с номером j.

Выходным алфавитом $Channel$ будет множество $Y_{Ch} = \{(b^1, \ldots, b^{freqnum}) | b^j \in Z4, 1 \leqslant j \leqslant freqnum\}$,т.е. у $Channel$ есть $freqnum$ выходов, соответствующих сигналу на каждом из каналов после добавления помех и с учетом коллизий.

Функция выходов

$$\lambda = (\lambda^1, \ldots, \lambda^{freqnum}) : [0,T] \times X_{Ch} \to Y_{Ch}$$

у $Channel$ определяется следующим образом. Пусть $I_k = \{i | 0 \leqslant i \leqslant nodenum \wedge f^i = f^k\}$, тогда:

если $t < t_1^k$ или $b^i = 0$, $i \in I_k$, $e^k = 0$, $1 \leqslant k \leqslant freqnum$,

то $\lambda^k \left(t, \left(b^0, f^0\right), \ldots, \left(b^{nodenum}, f^{nodenum}\right), \left(e^1, \ldots, e^{freqnum}\right)\right) = 0$;

если $t \geqslant t_1^k$, $b^i = 0$, $i \in I_k$, $i \neq l$, $e^k = 0$, $1 \leqslant k \leqslant freqnum$,

то $\lambda^k \left(t, \left(b^0, f^0\right), \ldots, \left(b^{nodenum}, f^{nodenum}\right), \left(e^1, \ldots, e^{freqnum}\right)\right) = b^l$;

если $t \geqslant t_1^k$, $e^k \neq 0$ или есть $b_i \neq 0$, $b_j \neq 0$, $i \neq j$, $1 \leqslant i,j \leqslant nodenum$, такие что $f_i = f_j = f_k$,

то $\lambda^k(t, (b^0, f^0), \ldots, (b^{nodenum}, f^{nodenum}), (e^1, \ldots, e^{freqnum})) = |b^k - 3|$.

Наличие помехи (точнее, битовой ошибки) $e^j(t)$ в канале j в момент времени t будем считать случайной величиной, распределенной по закону Бернулли, т.е. $P\{e(t) = -1\} = p$, $P\{e(t) = 0\} = 1 - p$. Время в предлагаемой математической модели дискретно, минимальный квант времени Δt соответствует времени, необходимому для передачи одного бита информации со скоростью v, определенной в предыдущем разделе.

РСт и ЦУС можно моделировать как автомат Мили $Station$ с входным алфавитом $X_{St} = \{(b^1, \ldots b^{freqnum}) | b^k \in Z4,\ 1 \leqslant k \leqslant freqnum$, выходным алфавитом $Y_{St} = \{(b,j) | b \in \mathbb{Z}_3, j \in F\}$ и множеством состояний F, т.е. как дискретный преобразователь с $freqnum$ входами, соответствующими сигналам в каждом из каналов, и с одним выходом, соответствующим паре из передаваемого сигнала и канала, на котором передается сигнал (битовая последовательность). На рис. 3.19 показано, как автоматы $Station$ и $Channel$ соединены друг с другом.

Математически поведение 2-го, наиболее интересного, этапа работы алгоритма настройки можно описать с помощью следующих уравнений:

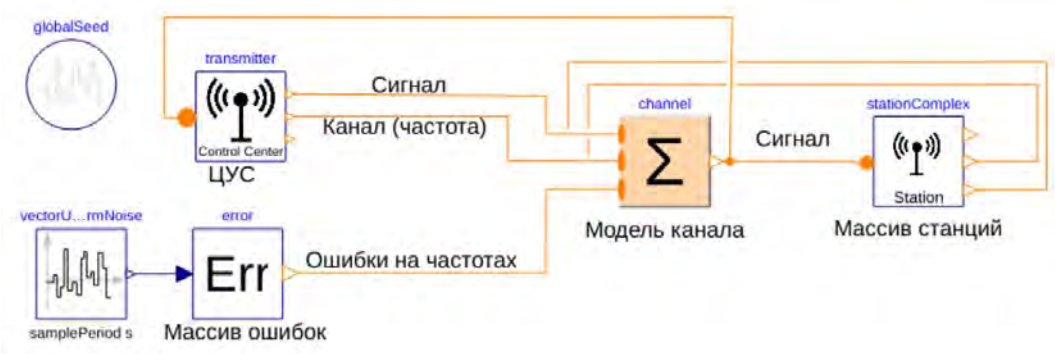


Рисунок 3.19 — Модель системы связи

$$b_R^k(t) = \begin{cases} \lambda^k\left(\left(b^0(t), f^0(t)\right), \ldots, \left(b^{nodenum}, f^{nodenum}\right), \left(e^1(t), \ldots, e^{freqnum}(t)\right)\right), \\ 0, \quad f_R^k(t) \neq f_T^0(t), \end{cases}$$

где $b_R^k(t)$ – последовательность, принимаемая k-й радиостанцией.

Канал приемника радиостанции с номером k изменяется с некоторыми упрощениями (в предположении, что каналы упорядочены по отношению сигнал-шум и перенумерованы подряд) следующим образом:

$$f^k(t+t_w) = \begin{cases} 0, \quad t + t_w < t_1^k; \\ \left(f^k(t) + 1 \; mod \; freqnum\right), \quad M \not\subseteq b_R^k[t, t_w]; \\ f^k(t), \quad M \subset b_R^k[t, t_w]; \end{cases} \qquad f^k(\tau_i - 0) = 0, f^k(\tau_i) = f_R^k,$$

где $t_w = \tau_c + \tau_d + 3L_T/v$.

Очевидно, что прием маячка настройки совершится не ранее чем через время

$$T_b = \max_{i=1, \ldots, nodenum} t_1^i + f_T^0 t_w.$$

В 1-м этапе работы алгоритма на радиостанции далее может применяться обычный CSMA/CA, поэтому достаточно только сказать, что время завершения этапа будет не меньше $T_0 = T_b + nodenum \; L_R/v$.

3.5.4 Моделирование преднамеренной помехи

Кроме ситуации, когда вероятность $\mathbf{P}\{\mathbf{e}^k(\mathbf{t}) = -\mathbf{1}\}$ битовой ошибки в канале $\mathbf{k}$ в каждый момент времени $\mathbf{t}$ постоянна, возможна ситуация, в которой $\mathbf{P}\left\{\mathbf{e}^k(\mathbf{t}) = -\mathbf{1}\right\} = \mathbf{p}_k(\mathbf{t})$ меняется со временем. Это будет соответствовать,

например, тому, что, начиная с некоторого момента времени τ_k, будет резко увеличиваться вероятность подавления радиочастотного канала k.

Кроме ситуации, когда вероятность $\mathbf{P}\{\mathbf{e}^k(\mathbf{t}) = -\mathbf{1}\}$ битовой ошибки в канале k в каждый момент времени $\mathbf{t}$ постоянна, возможна ситуация, в которой $\mathbf{P}\left\{\mathbf{e}^k(\mathbf{t}) = -\mathbf{1}\right\} = \mathbf{p}_k(\mathbf{t})$ меняется со временем. Это будет соответствовать, например, тому, что, начиная с некоторого момента времени τ_k, будет резко увеличиваться вероятность подавления радиочастотного канала $\mathbf{k}$. Тогда

$$p_k(t) = \begin{cases} 0, & t < \tau_k; \\ 1, & t \geqslant \tau_k. \end{cases}$$

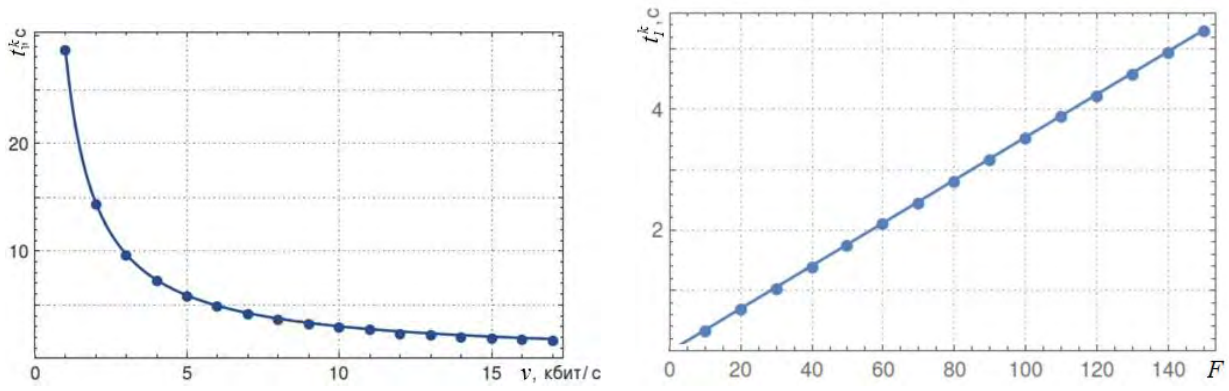
Очевидно, что, чем больше каналов используется при настройке, тем меньше вероятность их одновременного подавления; поэтому времена τ_k $(1 \leqslant k \leqslant freqnum)$ необходимо генерировать так, чтобы

$$P\left(\sum_{\substack{1 \leqslant i,j \leqslant freqnum \\ i \neq j}} |\tau_i - \tau_j| < \varepsilon\right) \leqslant C_1\varepsilon,\ C_1 > 0,$$

где C_1 – некоторая константа.

3.5.5 Имитационное моделирование системы настройки радиосети специального назначения

Моделирование канального уровня. Работа описанного в предыдущем разделе алгоритма была промоделирована в среде Wolfram System Modeler. Будем считать, что для всех каналов $f_k \in F$, $t \in [0,T]$ значение функции помех $e(t) = e^k(t)$ является распределенной по закону Бернулли случайной величиной: $P\{e(t) = -1\} = 10^{-3}$; используется до 150 каналов; время перестройки с канала на канал 1 мс. Время начала работы 2-го этапа алгоритма в k-й радиостанции $t_1^k = 0{,}171427$ с. Схема системы связи в Wolfram System Modeler приведена на рис. 6. В результате были установлены (рис. 3.20) закономерности (с помощью метода наименьших квадратов зависимости аппроксимируются как $t_1^k = 0{,}15 + 28{,}65/v$ и $t_1^k = -0{,}02 + 0{,}036\ F$ соответственно).



а) б)

Рисунок 3.20 — Зависимость продолжительности 2-го этапа настройки t_1^k от скорости передачи данных v при использовании 50 частотных каналов (а) и зависимость продолжительности 2-го этапа настройки t_1^k от количества частотных каналов F при скорости передачи данных $v = 16$ кбит/с (б)

Моделирование физического уровня. Моделирование обмена сообщениями, описанными в подразделе 3.5.2, на физическом уровне было произведено в среде MatLab. Непосредственная работа в MatLab и построение графиков было выполнено по просьбе автора диссертации Жарковым С.Н. Сделаны следующие предположения:

1. При настройке радиостанции используются три сообщения: маячок настройки (МН), маячок ответа (МО) и маячок завершения (МЗ).
2. Сообщение МН отправляется через радиоканал с параметрами: радиочастота f_1 и отношение сигнал/шум SNR_1.
3. Сообщение МО отправляется через радиоканал с параметрами: радиочастота f_2 и отношение сигнал/шум SNR_2.
4. Сообщение МЗ отправляется через радиоканал с параметрами: радиочастота f_3 и отношение сигнал/шум SNR_3.
5. Сообщения передаются при *SNR*=10 дБ, а при наличии помехи сообщение не передается. Вероятность подавления обозначается как P_i.
6. $SNR_k = 4{,}5$ дБ (в отсутствие помехи) и $SNR_k = 0$ дБ (при наличии помехи).
7. Если частота f_k, на которой передается одно из трех сообщений подавлено, происходит переход на другую радиочастоту. Допускаются три попытки перехода на другую частоту. Если попытки перейти на другие частоты оказались неудачными, выжидается 3 мс, после чего попытки повторяются.

8. Задержка в радиоканале равна 1 мс.

Структурная схема системы настройки показана на рис. 3.21, характеристики автоматической настройки радиостанции приведены на рис. 3.22.

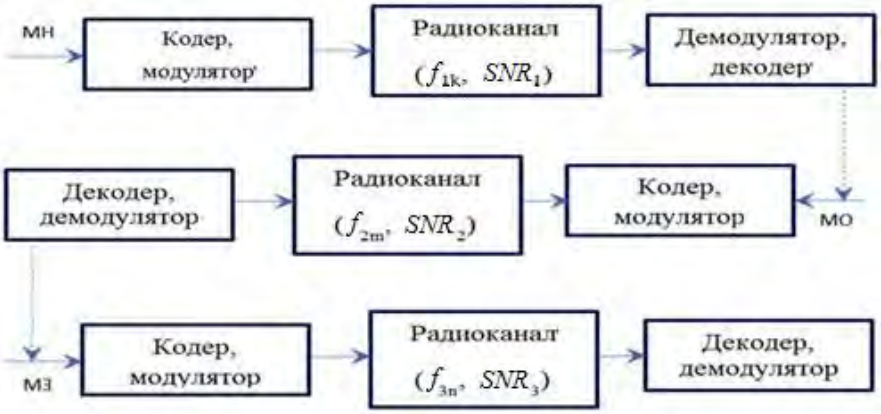
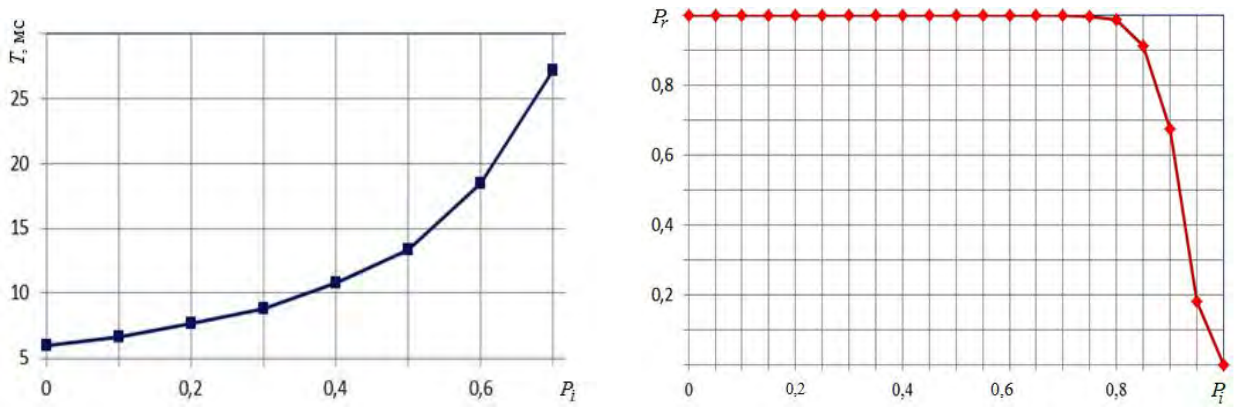


Рисунок 3.21 — Автоматическая настройка радиостанции в сети специального назначения



а) б)

Рисунок 3.22 — Зависимость времени автоматической настройки T от вероятности подавления используемой частоты P_i (а) и зависимость вероятности автоматической настройки P_r от вероятности подавления используемой частоты P_i (б)

Глава 4. Непрерывная постановка задачи движения агента по оптимальному по времени маршруту и связанные задачи

4.1 Введение

Общий подход к созданию алгоритмов обхода препятствий и построения в различные виды строя для роботов изложен, например, в [146]. В настоящей работе описывается связь между непрерывной постановкой с использованием вышеуказанного подхода задачи оптимального по времени обхода препятствий роботами и ранее сконструированных автором клеточных автоматов (см. [1; 2]).

Разд. 4.2 настоящей работы посвящен непрерывной постановке задачи движения агента по пересеченной местности и начинается с рассмотрения элементарного случая, решаемого классическими методами вариационного исчисления, а далее в разделе рассматриваются вопросы существования оптимального маршрута агента. Разд. 4.3 посвящен дискретизации непрерывной модели движения и получению оценок аппроксимации оптимального маршрута ломаными специального вида, а также конструкции клеточного автомата, с помощью которого ищется дискретный оптимальный маршрут.

В главе применяются стандартные обозначения: $\|u\| = \sqrt{\sum_{i=1}^{n} u_i^2}$ – евклидова норма вектора $u \in \mathbb{R}^n$, $u = (u_1, \ldots, u_n)$, $W_p^m(A; \mathbb{R}^n)$ – пространства Соболева определенных на A функций со значениями в $\mathbb{R}^n$, которые вместе со своими m слабыми производными содержатся в $L_p(A; \mathbb{R}^n)$ с нормой

$$\|u\|_{W_p^m(A;\mathbb{R}^n)} = \sqrt{\sum_{i=1}^{n} \Bigg(\sum_{|\alpha| \leqslant m} \int_A |D^\alpha u_i(x)|^p dx \Bigg)^{2/p}},$$

$C(A; \mathbb{R}^n)$ – пространство определенных на A непрерывных функций со значениями в $\mathbb{R}^n$ с нормой $\|u\|_{C(A;\mathbb{R}^n)} = \max_{x \in A} \|u(x)\|$, $C^m(A; \mathbb{R}^n)$ – пространство m раз непрерывно-дифференцируемых определенных на A функций с нормой $\|u\|_{C^m(A;\mathbb{R}^n)} = \sum_{|\alpha| \leqslant m} \max_{x \in A} \|D^\alpha u(x)\|$. Будет также для краткости обозначаться $C(A; \mathbb{R}) = C(A)$, $C^m(A; \mathbb{R}) = C^m(A)$, $\nabla u(t,x,y) = (u'_x, u'_y)$, $\nabla_{(t,x,y)} u(t,x,y) = (u'_t, u'_x, u'_y)$.

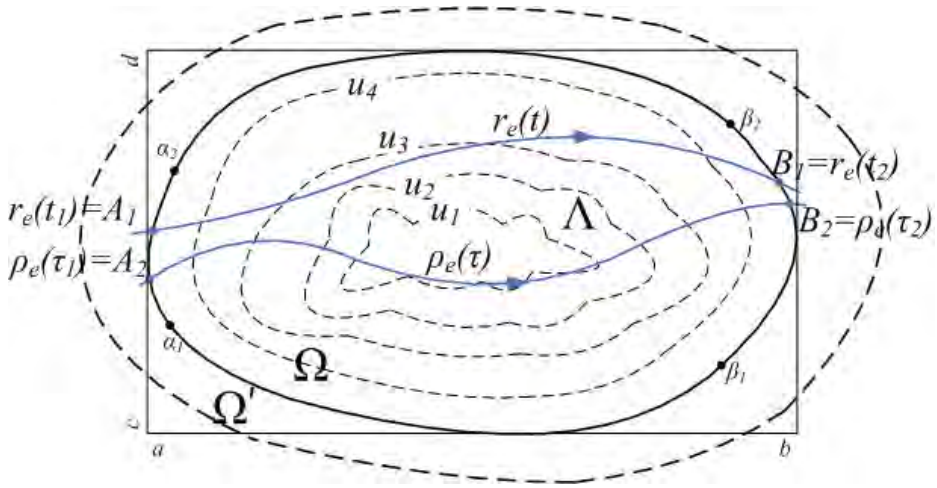


Рисунок 4.1 — Движение агента около предварительно заданного маршрута ρ. Линии уровня функции u обозначены как u_1, u_2, u_3, u_4.

4.2 Непрерывная модель

4.2.1 Тривиальный случай

Пусть агент e движется в ограниченной области $\Omega \subset \Omega' \subset \mathbb{R}^2$ по пути $r_e = r_e(t)$, $t \in [0,T]$, причем задана функция

$$u : [0,T] \times \Omega' \to \mathbb{R}, \quad \forall (r \in \Omega) \quad u(t,r) \geqslant 0,$$

такая что $\forall (t \in [0,T])$ $\|\dot{r}_e(t)\| = u(t, r_e(t))$, задающая ограничение на скорость перемещения в области. Практический способ построения такой функции проходимости описан, например, в [147]. В случае, если некоторое препятствие совершенно непроходимо и в содержащей его подобласти Ω_o верно что $u(t,r) = 0$, $t \in [0,T]$, $r \in \Omega_o$, то содержащая его область исключается из Ω.

Известно, что $r(0) = A_1 = (a_1^1, a_1^2) \in \partial\Omega$, т.е. путь агента e начинается на границе области в точке (a_1^1, a_1^2) и задан некоторый образцовый маршрут $\rho : [0,T] \to \Omega$, от которого агент не должен слишком отклоняться, $\rho(0) = A_2 = (a_2^1, a_2^2) \in \partial\Omega$ и существует $\tau_1 \in (0,T]$ такое что $\rho(\tau_1) = B_2 = (b_2^1, b_2^2) \in \partial\Omega$ (см. фиг. 4.1).

Параметризация временем как для r_e, так и для ρ заранее не известна, хотя для ρ задан график $\Gamma(\rho) = \{(x(t), y(t)) | t \in [0,\tau_1]\}$. Необходимо найти маршрут агента r_e, не слишком отличающийся от образцового и при этом занимающий как можно меньше времени.

Введем обозначение

$$a = \inf_{(x,y)\in\Omega} x, \quad b = \sup_{(x,y)\in\Omega} x, \quad c = \inf_{(x,y)\in\Omega} y, \quad d = \sup_{(x,y)\in\Omega} y.$$

Начнем обсуждение с самого простого случая, когда функция проходимости u не зависит от t, то есть $u(t,r) = u(r) = u(x,y)$ и $u \in C(\Omega)$, $\min_{r\in\Omega} u(r) = u_{\min} > 0$. Также предположим, что $\Omega = [a,b] \times [c,d] = \Omega'$, образцовый маршрут ρ допускает представление в виде графика явно заданной функции $y_\rho : [a,b] \to \mathbb{R}$, все возможные искомые маршруты r_e также имеют регулярное представление в виде графиков функций $y_e \in C^2[a,b]$, $d \leqslant y_e \leqslant c$, $\|y_e\|_{C^2[a,b]} \leqslant \alpha$. Множество таких функций-представлений возможных маршрутов обозначим через $\mathcal{Y}$.

Поиск наилучшего маршрута r_e является задачей оптимизации, в которой необходимо минимизировать функционал близости маршрута к образцовому:

$$J_\rho : \mathcal{Y} \to \mathbb{R}, \tag{4.1}$$

$$J_\rho(y_e) = \|y_e - y_\rho\|_{C[a,b]}. \tag{4.2}$$

Время, за которое агент e будет проходить по заданному маршруту, будет определяться в виде

$$\Theta(y_e) = \int_0^{t_1} \mathrm{d}t = \int_0^{t_1} \frac{u(r_e(t))}{u(r_e(t))}\mathrm{d}t = \int_0^{t_1} \frac{\|\dot{r}_e(t)\|\mathrm{d}t}{u(r_e(t))} = \int_{\widehat{A_1B_1}} \frac{\mathrm{d}l}{u(r_e)} = \int_a^b \frac{\sqrt{1+y_e'^2(x)}\mathrm{d}x}{u(x,y_e(x))}. \tag{4.3}$$

Таким образом, необходимо минимизировать на $\mathcal{Y} \subset C^2[a,b]$ функционал

$$J(y) = \alpha_1 J_\rho(y) + \alpha_2 \Theta(y) \to \min, \tag{4.4}$$

где α_1, $\alpha_2 \geqslant 0$, $\alpha_1 + \alpha_2 > 0$ – параметры модели, указывающие на то, стремится ли агент к кратчайшему пути или к наибольшему следованию ранее заданному маршруту.

Справедлива

Лемма 1. *Функционал $J : C^1[a,b] \to \mathbb{R}$ непрерывен и достигает минимума на $\mathcal{Y}$.*

Доказательство. Заметим, что для всех α, $\beta \in \mathbb{R}$ имеет место неравенство

$$\frac{|\alpha+\beta|}{\sqrt{1+\alpha^2}\sqrt{1+\beta^2}} \leqslant \frac{|\alpha+\beta|}{\sqrt{1+\alpha^2}+\sqrt{1+\beta^2}} \leqslant 1. \tag{4.5}$$

В силу данного неравенства справедлива оценка

$$
\begin{aligned}
|\Theta(y)-\Theta(y_k)| &\leqslant \int_a^b \left|\frac{\sqrt{1+y'^2(x)}}{u(x,y(x))}-\frac{\sqrt{1+y_k'^2(x)}}{u(x,y_k(x))}\right|\mathrm{d}x \leqslant \\
&\leqslant \int_a^b \frac{u(x,y_k(x))}{u(x,y(x))u(x,y_k(x))}|\sqrt{1+y'^2(x)}-\sqrt{1+y_k'^2(x)}|\mathrm{d}x+ \\
&\quad + \int_a^b \frac{|u(x,y_k(x))-u(x,y(x))|}{u(x,y(x))u(x,y_k(x))}\sqrt{1+y_k'^2(x)}\mathrm{d}x \leqslant \\
&\leqslant \frac{\|u\|_{C(\Omega)}}{u_{\min}^2}\int_a^b |y'(x)-y_k'(x)|\frac{|y'(x)+y_k'(x)|}{\sqrt{1+y'^2(x)}+\sqrt{1+y_k'^2(x)}}\mathrm{d}x+\frac{c_0}{u_{\min}^2}\int_a^b |u(x,y_k(x))- \\
&\quad - u(x,y(x))|dx \leqslant \frac{\|u\|_{C(\Omega)}}{u_{\min}^2}\|y'-y_k'\|_{L_1(a,b)}+\frac{c_0}{u_{\min}^2}\int_a^b |u(x,y_k(x))-u(x,y(x))|dx.
\end{aligned}
\tag{4.6}
$$

Из неравенства (4.6) следует непрерывность функционала $\Theta : C^1[a,b] \to \mathbb{R}$, а непрерывность функционала $J_\rho : C^1[a,b] \to \mathbb{R}$ очевидна. В силу теорем Реллиха-Кондрашова (см. [148]) имеет место компактное вложение $\mathcal{Y} \subset W_2^2(a,b) \subset C^1[a,b]$, что по теореме Вейерштрасса (см., например, [149]) влечет достижение функционалом J минимума на $\mathcal{Y}$. □

Обратим внимание, что уравнение Эйлера-Лагранжа для экстремалей функционала Θ сводится к уравнению

$$
\left(y'(x)^2+1\right)\left(y'(x)u^{(1,0)}(x,y(x))-u^{(0,1)}(x,y(x))\right)-y''(x)u(x,y(x))=0.
$$

Однако для практических задач движения агентов, например, для создания алгоритмов движения роботов, использовать вышеуказанное уравнение затруднительно: во-первых, оно нелинейно и трудно решается даже при относительно простом виде функции u, во-вторых, функция u обычно задана в табличном виде, следовательно, ее необходимо дополнительно сглаживать.

4.2.2 Движение агента в произвольной ограниченной области с непрерывной кусочно-гладкой границей

Как и ранее, задана функция проходимости области $u : [0,T] \times \Omega' \to \mathbb{R}$, $u \geqslant 0$, $u \in C([0,T]; C^1(\Omega))$. Пусть маршрут движения агента описывается

параметрически заданной кривой $r = r(l) = (x(l), y(l))$, где l – например, натуральный параметр, $r \in C^1([0,L],\Omega')$. Нашей задачей будет найти такую параметризацию $l = l(t)$, что $\|\partial_t r(l(t))\| = u(t,x(l(t)),y(l(t)))$ и выбрать маршрут $r(l)$ таким образом, чтобы время выхода агента по данному маршруту на границу области $\partial\Omega$, $\Omega \subset \Omega'$ было минимально. Для упрощения дальнейшего изложения будем считать, что $u(t,x,y) = 0$ для всех $t \in [0, +\infty)$, $(x,y) \in \mathbb{R}^2 \setminus \Omega'$. Положим, что граница $\partial\Omega$ – многообразие класса C^1.

Заметим, что длина маршрута, пройденная до момента $t \in [0,T]$, $0 < T \leqslant \infty$, выражается из известной формулы дифференциальной геометрии:

$$l(t) = \int_0^t \|\partial_s r(l(s))\| ds = \int_0^t u(s, x(l(s)),y(l(s)))ds,\ t \geqslant 0. \tag{4.7}$$

Далее мы будем строить алгоритм численного решения задачи построения скорейшего маршрута, приближая гладкий маршрут агента ломаной, а функцию проходимости – ступенчатой функцией. Таким образом, в качестве множества маршрутов мы будем рассматривать $\mathcal{Y} \subset W_p^1(0,L;\mathbb{R}^2)$, состоящее из всех вектор-функций из $C([0,L];\mathbb{R}^2)$, которые

– представимы в виде не более чем счетного множества дважды непрерывно дифференцируемых кусков, т. е. для любой $r \in \mathcal{Y}$ существует такое разбиение $\mathbb{R}$ точками $\{t_i\}$, что $r|_{(t_i,t_{i+1})} \in C^2((0,L);\mathbb{R}^2)$;

– производные всех функций из $\mathcal{Y}$ равномерно ограничены (кроме, возможно, счетного множества точек, в которых производной не существует):

$$\max_{l\in[0,L]} \|r'(l)\| \leqslant C_1;$$

– вторые производные всех функций из $\mathcal{Y}$ равномерно ограничены в сумме:

$$\sum_i \max_{l\in(L_i,L_{i+1})} \|r''(l)\|^p \leqslant C_s,$$

где L_i – точки разрыва первой и второй производных r.

Введем обозначение $\mathcal{Y}_0 = C^2(0,L;\Omega') \cap \mathcal{Y}$, $\mathcal{Y}_L = \mathcal{Y} \setminus \mathcal{Y}_0$.

Далее потребуются несколько утверждений о разрешимости и непрерывной зависимости решения от параметра для уравнения (4.7).

Лемма 2. *Пусть $u \in C([0,T];C^1(\Omega))$ – ограниченная функция, $r \in \mathcal{Y}$. Тогда уравнение (4.7) имеет единственное решение в $C^1[0,T]$.*

Доказательство. Доказательство следует из классического результата для уравнений Вольтерра. Действительно, если $r = (x(l),y(l)) \in \mathcal{Y}$, то в силу обобщения теоремы Лагранжа о среднем значении для вектор-функций (см. [150], Theorem 3, Theorem 4)

$$r(l+h) = r(l) + h\sum_{i=1}^{3}\lambda_i r'_+(l+\theta_i h) \leqslant r(l) + Kh, \theta_i \in (0,1), \lambda_i \geqslant 0, \sum_{i=1}^{3}\lambda_i = 1. \tag{4.8}$$

где r'_+ – односторонняя производная r, $K = (K_1,K_2)$. Поэтому и в силу ограниченности производных r функция u удовлетворяет условию Липшица

$$|u(s,x(l+h),y(l+h))-u(s,x(l),y(l))| = |u(s,x(l)+K_1h,y(l)+K_2h)-u(s,x(l),y(l))| \leqslant$$
$$\leqslant \|\nabla u(t,\cdot)\|_{C(\Omega)}\|K\||h| \leqslant C_1\|\nabla u(t,\cdot)\|_{C(\Omega)}h \leqslant C_1\|\nabla u\|_{C([0,T]\times\Omega)}h = C_2h. \tag{4.9}$$

В силу этого и ограниченности u уравнение (4.7) имеет единственное решение в $C[0,T]$. А так как $l'(t) = u(t,x(l(t)),y(l(t)))$ – непрерывная функция, то $l \in C^1[0,T]$. □

Лемма 3. *Пусть выполнено условие леммы 2 и $r_k \to r_0$ в $W^1_p([0,L];\mathbb{R}^2)$, $p > 1$, $\{r_k\}_{k=1}^{\infty} \subset \mathcal{Y}_L$, $r_0 \in \mathcal{Y}_0$. Обозначим через l_k решение уравнения (4.7) с $r = r_k = (x_k(l),y_k(l))$. Тогда $l_k \to l_0$ в $C^1[0,T]$.*

Доказательство. Перепишем уравнение (4.7) в виде дифференциального уравнения

$$l'_k(t) = u(t,x_k(l_k(t)),y_k(l_k(t))), \tag{4.10}$$
$$l_k(0) = 0. \tag{4.11}$$

В силу неравенства (4.9) справедливы следующие соотношения

$$(l_1 - l_2)(u(t,x_0(l_1),y_0(l_1)) - u(t,x_0(l_2),y_0(l_2))) \leqslant C_2(l_1 - l_2)^2, \tag{4.12}$$

$$|u(t,x_k(l),y_k(l)) - u(t,x_0(l),y_0(l))| \leqslant \|\nabla u(t,\cdot)\|_{C(\Omega)}\|r_k(l) - r_0(l)\| \leqslant$$
$$\leqslant \|\nabla u\|_{C([0,T]\times\Omega)}\|r_k - r_0\|_{C([0,T],\mathbb{R}^2)} \tag{4.13}$$

для всех $t \in [0,T]$, l, l_1, $l_2 \in [0,L]$. В соответствии с теоремой 9 из [151] следует, что

$$|l_k(t) - l_0(t)| \leqslant \varepsilon(t), \tag{4.14}$$

где ε – решение задачи

$$\varepsilon'(t) = C_2\varepsilon(t) + \|\nabla u\|_{C([0,T]\times\Omega)}\|r_k - r_0\|_{C([0,L];\Omega')}, \tag{4.15}$$

$$\varepsilon(0) = 0. \tag{4.16}$$

Известно, что вложение $W_p^1([0,L];\Omega') \subset C([0,L];\Omega')$ непрерывно. Задача (4.15), (4.16) имеет решение

$$\varepsilon(t) = \frac{\|\nabla u\|_{C([0,T]\times\Omega)}\|r_k - r_0\|_{C([0,L];\Omega')}}{C_2}(e^{C_2 t} - 1) \leqslant$$
$$\leqslant \frac{\|\nabla u\|_{C([0,T]\times\Omega)}\|r_k - r_0\|_{C([0,L];\Omega')}}{C_1\|\nabla u\|_{C([0,T]\times\Omega)}}(e^{C_2 T} - 1) = C_3\|r_k - r_0\|_{C([0,L];\Omega')} \to 0 \tag{4.17}$$

при $\|r_k - r_0\|_{C([0,L];\Omega')} \leqslant c_1\|r_k - r_0\|_{W_p^1([0,L];\Omega')} \to 0$. Теперь положим $l_k(t) = l_0(t) + (l_k(t) - l_0(t))$ и из неравенств (4.9), (4.14) получим оценку

$$|l_k'(t) - l_0'(t)| = |u(t,x_k(l_k(t)),y_k(l_k(t))) - u(t,x_0(l_0(t)),y_0(l_0(t)))| \leqslant$$
$$\leqslant C_2|l_k(t) - l_0(t)| \leqslant C_2C_3\|r_k - r_0\|_{C([0,L];\Omega')} \to 0. \tag{4.18}$$

Из неравенств (4.14), (4.18) следует, что

$$\max_{t\in[0,T]}|l_k(t)-l_0(t)|+\max_{t\in[0,T]}|l_k'(t)-l_0'(t)| \leqslant \|\nabla u\|_{C([0,T]\times\Omega)}(e^{C_2T}-1)(1+\frac{1}{C_2})\|r_k-r_0\|_{C([0,L];\Omega')}$$

□

Лемма 4. *Пусть выполнены условия леммы 3. Тогда* $r_k \circ l_k \to r_0 \circ l_0$ *в* $C([0,T],\mathbb{R}^2)$, *а если* $0 < u_{\min} \leqslant \|u\|_{C([0,T]\times\Omega')}$, *то и в* $W_p^1([0,T],\mathbb{R}^2)$.

Доказательство. То, что $r_k \circ l_k \to r_0 \circ l_0$ в $C([0,T],\mathbb{R}^2)$ немедленно очевидно из оценок (4.8), (4.14):

$$|r_k(l_k(t)) - r_0(l_0(t))| \leqslant |r_k(l_k(t)) - r_k(l_0(t))| + |r_k(l_0(t)) - r_0(l_0(t))| \leqslant$$
$$\|K_k\||l_k(t)-l_0(t)|+\|r_k-r_0\|_{C([0,L],\mathbb{R}^2)} \leqslant \|K_k\|C_3\|r_k-r_0\|_{C([0,L],\mathbb{R}^2)}+\|r_k-r_0\|_{C([0,L],\mathbb{R}^2)} =$$
$$= (C_1C_3 + 1)\|r_k - r_0\|_{C([0,L],\mathbb{R}^2)}. \tag{4.19}$$

Пусть $r_k = (x_k,y_k)$. Покажем утверждение леммы для x_k (для y_k оно доказывается аналогично). Пользуясь неравенством Гёльдера, выполним оценку

$$\left(\int_0^T |\partial_t(x_k \circ l_k(t) - x_0 \circ l_0(t))|^p dt\right)^{1/p} \leqslant \left(\int_0^T |\partial_t(x_k \circ l_k(t) - x_k \circ l_0(t))|^p dt\right)^{1/p} +$$
$$+\left(\int_0^T |\partial_t(x_k \circ l_0(t) - x_0 \circ l_0(t))|^p dt\right)^{1/p} \leqslant \left(\int_0^T |\partial_{l_k} x_k(l_k(t))l_k'(t) - \partial_{l_0} x_k(l_0(t))l_0'(t)|^p dt\right)^{1/p}$$
$$+\left(\int_0^L |\partial_l(x_k(l) - x_0(l))|^p |l_0'(t)|^{p-1} dl\right)^{1/p} \leqslant \left(\int_0^T |\partial_{l_k} x_k(l_k(t))l_k'(t) - \partial_{l_0} x_k(l_0(t))l_k'(t)|^p dt\right)$$
$$+\left(\int_0^T |\partial_{l_0} x_k(l_0(t))l_k'(t) - \partial_{l_0} x_k(l_0(t))l_0'(t)|^p dt\right)^{1/p} + \left(\int_0^L |\partial_l(x_k(l) - x_0(l))|^p |l_0'(t)|^{p-1} dl\right)^{1/}$$
(4

Обратим внимание, что

$$|\partial_{l_k} x_k(l_k(t))l_k'(t) - \partial_{l_0} x_k(l_0(t))l_k'(t)| = |\partial_{l_k} x_k(l_k(t)) - \partial_{l_0} x_k(l_0(t))||l_k'(t)| \to 0$$

почти всюду в $[0,T]$ при $l_k \to l_0$ в $C^1[0,T]$. Поэтому, в силу теоремы Лебега о предельном переходе имеем

$$\int_0^T |\partial_{l_k} x_k(l_k(t))l_k'(t) - \partial_{l_0} x_k(l_0(t))l_k'(t)|^p dt \to 0. \tag{4.21}$$

Далее

$$\int_0^T |\partial_{l_0} x_k(l_0(t))l_k'(t) - \partial_{l_0} x_k(l_0(t))l_0'(t)|^p dt = \int_0^T |\partial_{l_0} x_k(l_0(t))(l_k'(t) - l_0'(t))|^p dt \leqslant$$
$$\leqslant \|l_k' - l_0'\|^p_{C[0,T]} \int_0^T |\partial_{l_0} x_k(l_0(t))|^p dt \leqslant \|l_k - l_0\|^p_{C^1[0,T]} \int_0^L |\partial_{l_0} x_k(l)|^p |l_0'(t)|^{-1} dl \leqslant$$
$$\leqslant \frac{1}{u_{\min}} \|l_k - l_0\|^p_{C^1[0,T]} \int_0^L |\partial_{l_0} x_k(l)|^p dl \leqslant \frac{1}{u_{\min}} \|l_k - l_0\|^p_{C^1[0,T]} \|x_k\|^p_{W_p^1(0,L)}. \tag{4.22}$$

Наконец, получим

$$\int_0^L |\partial_l(x_k(l) - x_0(l))|^p |l_0'(t)|^{p-1} dl \leqslant \frac{1}{u_{\min}^{1-p}} \|x_k - x_0\|^p_{W_p^1(0,L)}. \tag{4.23}$$

Оценивая слагаемые в (4.20) с помощью неравенств (4.21)–(4.23), получаем утверждение леммы. □

Определим функционал $\Theta : \mathcal{Y} \to \mathbb{R}$, который каждому $r \in \mathcal{Y}$ сопоставляет момент его первого пересечения с заданным участком Γ границы $\partial\Omega$:

$$\Theta(r) = \min\{t \in [0,T] | r(l(t)) \in \Gamma\},$$

где l – решение уравнения (4.7) для $r = (x(l),y(l))$. Нам необходимо установить существование минимума этого функционала в $\mathcal{Y}$

Теорема 1. *Пусть* $0 < u_{\min} = \min_{t\in[0,T],(x,y)\in\Omega} u(t,x,y)$. *Тогда функционал* $\Theta : \mathcal{Y} \to \mathbb{R}$ *– непрерывен в* $W_p^1([0,T],\mathbb{R}^2)$.

Доказательство. Пусть $\{r_k\}_{k=1}^{\infty} \subset \mathcal{Y}$, $r_k \to r_0$ в $W_p^1([0,T],\mathbb{R}^2)$ и $t_k = \Theta(r_k)$, $0 \leqslant k < \infty$. Воспользуемся обобщением теоремы Лагранжа о среднем значении для вектор-функций (см. [150], Theorem 3, Theorem 4)

$$r_k(l_k(t_k)) - r_k(l_k(t_0)) = (t_k - t_0)\sum_{i=1}^{3}\lambda_i(r_k\circ l_k)'_+(\xi_i),\ \xi_i \in (t_0,t_k),\ \lambda_i \geqslant 0,\ \sum_{i=1}^{3}\lambda_i = 1. \tag{4.24}$$

Введем обозначение

$$C_4 = \left\|\sum_{i=1}^{3}\lambda_i(r_k\circ l_k)'_+(\xi_i)\right\|,$$

тогда

$$\begin{aligned}|t_k - t_0| = \frac{\|r_k(l_k(t_k)) - r_k(l_k(t_0))\|}{C_4} &\leqslant \frac{\|r_k(l_k(t_k)) - r_0(l_0(t_0))\|}{C_4} + \\ &+ \frac{\|r_0(l_0(t_0)) - r_k(l_k(t_0))\|}{C_4} \leqslant \\ &\leqslant \frac{\|r_k(l_k(t_k)) - r_0(l_0(t_0))\|}{C_4} + \frac{\|r_0\circ l_0 - r_k\circ l_k\|_{C([0,T],\mathbb{R}^2)}}{C_4}.\end{aligned} \tag{4.25}$$

Второе слагаемое в правой части (4.25) стремится к нулю в силу леммы 4. Пусть пересекаемый кусок границы $\Gamma \subseteq \partial\Omega$ задается формулой $r_\Gamma = (x_\Gamma(s),y_\Gamma(s))$, $r_\Gamma \in C^1[0,L_\Gamma]$ и этот кусок пересекается в точках $A_k = r_k(l_k(t_k)) = r_\Gamma(s_k)$ и $A_0 = r_0(l_0(t_0)) = r_\Gamma(s_0)$, s_k, $s_0 \in [0, L_\Gamma]$. Пусть $B_0 = r_0(l_0(t_k))$ и $B_k = r_k(l_k(t_0))$ и исследуем свойства четырехугольника $B_kA_0B_0A_k$.

Пусть, например, $t_0 \leqslant t_k$ (в случае $t_0 > t_k$ нужно будет просто поменять порядок соединения вершин). Нам необходимо получить оценку для длины диагонали A_kA_0, т.е. для $\|r_k(l_k(t_k)) - r_0(l_0(t_0))\|$. Очевидно, что в силу леммы 4 имеем

$$|B_kA_0| = \|r_k(l_k(t_0)) - r_0(l_0(t_0))\| \leqslant \|r_0\circ l_0 - r_k\circ l_k\|_{C([0,T],\mathbb{R}^2)} \to 0, \tag{4.26}$$

$$|B_0A_k| = \|r_k(l_k(t_k)) - r_0(l_0(t_k))\| \leqslant \|r_0\circ l_0 - r_k\circ l_k\|_{C([0,T],\mathbb{R}^2)} \to 0. \tag{4.27}$$

Положим $\alpha_k = \max\{l_k(t_0), l_0(t_0)\}$, $\beta_k = \min\{l_k(t_k), l_0(t_k)\}$. Верны соотношения

$$|A_0B_0| = \|r_0(l_0(t_0)) - r_0(l_0(t_k))\| \leqslant \int_{t_0}^{t_k} \|\partial_s r_0(l_0(s))\| ds = \int_{l_0(t_0)}^{l_0(t_k)} \|\partial_l r_0(l)\| dl, \tag{4.28}$$

$$|A_kB_k| = \|r_k(l_k(t_k)) - r_k(l_k(t_0))\| \leqslant \int_{t_0}^{t_k} \|\partial_s r_k(l_k(s))\| ds = \int_{l_k(t_0)}^{l_k(t_k)} \|\partial_l r_k(l)\| dl. \tag{4.29}$$

Предположим, что ни $|A_0B_0|$, ни $|A_kB_k|$ не стремятся к нулю (если они стремятся к нулю, то сразу получаем утверждение теоремы). Заметим, что в силу неравенства треугольника $|\|x\| - \|y\|| \leqslant \|x - y\|$, леммы 3 и непрерывного вложения $W_p^1([0,L],\mathbb{R}^2) \subset W_1^1([0,L],\mathbb{R}^2)$ имеем

$$\begin{aligned} ||A_kB_k| - |A_0B_0|| \leqslant \int_{\alpha_k}^{\beta_k} |\|\partial_l r_k(l)\| - \|\partial_l r_0(l)\||dl + \\ + \left|\int_{l_k(t_0)}^{l_0(t_0)} \|\partial_l r_k(l)\| dl\right| + \left|\int_{l_k(t_k)}^{l_0(t_k)} \|\partial_l r_k(l)\| dl\right| + \\ + \left|\int_{l_k(t_0)}^{l_0(t_0)} \|\partial_l r_0(l)\| dl\right| + \left|\int_{l_k(t_k)}^{l_0(t_k)} \|\partial_l r_0(l)\| dl\right| \leqslant \int_0^L \|\partial_l(r_k(l) - r_0(l))\| dl + \\ + 4C_1\|l_k - l_0\|_{C[0,T]} \leqslant \\ \leqslant \|r_k - r_0\|_{W_1^1([0,L],\mathbb{R}^2)} + 4C_1\|l_k - l_0\|_{C[0,T]} \to 0. \end{aligned} \tag{4.30}$$

В силу неравенства Птолемея для выпуклого четырехугольника, соотношений (4.26), (4.27) и (4.30) получаем что

$$|A_kA_0||B_kB_0| \leqslant |A_0B_0||B_kA_0| + |A_kB_k||B_0A_k| \to 0. \tag{4.31}$$

Осталось оценить $|B_kB_0|$. В силу теоремы косинусов

$$\begin{aligned} |B_kB_0|^2 = |B_0A_k|^2 + |A_kB_k|^2 - 2|B_0A_k||A_kB_k|\cos\angle B_kA_kB_0 \geqslant |B_0A_k|^2 + |A_kB_k|^2 - \\ - 2|B_0A_k||A_kB_k|. \end{aligned} \tag{4.32}$$

В силу (4.31), (4.32) получаем

$$\|r_k(l_k(t_k)) - r_0(l_0(t_0))\| = |A_kA_0| \leqslant \frac{|A_0B_0||B_kA_0| + |A_kB_k||B_0A_k|}{|B_0A_k|^2 + |A_kB_k|^2 - 2|B_0A_k||A_kB_k|} \to 0. \tag{4.33}$$

Из (4.25), (4.32) получаем, что

$$|t_k - t_0| \to 0.$$

□

Лемма 5. *Множество* $\mathcal{Y} \subset W_p^1([0,L],\mathbb{R}^2)$ *компактно в* $W_p^1([0,L],\mathbb{R}^2)$.

Доказательство. В силу теоремы 7.7.2 из [152, С. 335] для компактности ограниченного в $C([0,L],\mathbb{R}^2)$, а значит, и в $L_p([0,L],\mathbb{R}^2)$ множества $\mathcal{Y}$ необходимо и достаточно равностепенной непрерывности по сдвигу в $W_p^1(\mathbb{R},\mathbb{R}^2)$. Кроме этого, требуется равномерное убывание по норме L_p на бесконечности, но так как поведение функций за пределами ограниченной области Ω' нас не интересует, то можно считать, что они продолжаются соответствующим образом, например, нулем. Пусть L_i – точки разрыва первой и второй производных функции r (по конструкции $\mathcal{Y}$ их не более чем счетное число). Пусть $h \to 0$, тогда получаем равномерную непрерывность по сдвигу в норме $W_p^1(\mathbb{R},\mathbb{R}^2)$:

$$\int_0^L \|\partial_l(r(l+h) - r(l))\|^p dl = \sum_i \int_{L_i}^{L_{i+1}} \|\partial_l(r(l+h) - r(l))\|^p_{C((L_i,L_{i+1}),\mathbb{R}^2)} dl \leqslant$$
$$\leqslant h^p L \sum_i \|r''\|^p_{C((L_i,L_{i+1}),\mathbb{R}^2)} = h^p L C_s \to 0.$$

□

Теорема 2. *Функционал* $\Theta : \mathcal{Y} \to \mathbb{R}$ *достигает минимума на* $\mathcal{Y}$.

Доказательство. В силу теоремы 1 функционал $\Theta : \mathcal{Y} \to \mathbb{R}$ непрерывен, а в силу леммы 5 множество $\mathcal{Y}$ является компактом. Тогда по теореме Вейерштрасса (см., например, [149]) функционал $\Theta : \mathcal{Y} \to \mathbb{R}$ достигает минимального и максимального значений в $\mathcal{Y}$. □

Обратим внимание, что решение $l_k = l_k(t)$ задачи (4.10), (4.11) является строго монотонной непрерывной функцией, если $0 < u_{\min} \leqslant \min_{t\in[0,T],(x,y)\in\Omega} u(t,x,y)$. В силу этого, у нее однозначно определена непрерывная обратная функция $t_k = t_k(l)$, $t_k' = \frac{1}{l_k'}$, являющаяся решением задачи Коши

$$t_k'(l) = \frac{1}{u(t_k(l),x_k(l),y_k(l))}, \quad t_k(0) = 0. \tag{4.34}$$

Лемма 6. *Пусть t_k, t_0 – решения задачи (4.34), $u \in C^1([0,T] \times \Omega)$. Тогда справедлива оценка*

$$|t_k(l) - t_0(l)| \leqslant (e^{l\|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}u_{\min}^{-2}} - 1)\|r_k - r_0\|_{C[0,L]}.$$

Доказательство. Доказательство в целом повторяет доказательство леммы 3. Получаем оценки

$$\left|\frac{1}{u(t_1,x_0(l),y_0(l))} - \frac{1}{u(t_2,x_0(l),y_0(l))}\right| = \left|\frac{u(t_2,x_0(l),y_0(l)) - u(t_1,x_0(l),y_0(l))}{u(t_1,x_0(l),y_0(l))u(t_2,x_0(l),y_0(l))}\right| =$$
$$= \frac{|u^{(1,0,0)}(\tau,x_0(l),y_0(l))||t_2 - t_1|}{|u(t_1,x_0(l),y_0(l))u(t_2,x_0(l),y_0(l))|} \leqslant \|u^{(1,0,0)}\|_{C([0,T]\times\Omega)}u_{\min}^{-2}|t_2 - t_1| \leqslant$$
$$\leqslant \|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}u_{\min}^{-2}|t_2 - t_1|, \quad (4.35)$$

$$\left|\frac{1}{u(t,x_0(l),y_0(l))} - \frac{1}{u(t,x_k(l),y_k(l))}\right| \leqslant \left|\frac{u(t,x_k(l),y_k(l)) - u(t,x_0(l),y_0(l))}{u_{\min}^2}\right| \leqslant$$
$$\leqslant \|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}u_{\min}^{-2}\|r_k - r_0\|_{C[0,L]}. \quad (4.36)$$

Для того чтобы получить оценку на разность $|t_k(l) - t_0(l)|$, решим дифференциальное уравнение с коэффициентами, определенными оценками (4.35), (4.36):

$$\varepsilon'(l) = \|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}u_{\min}^{-2}(\varepsilon(l) + \|r_k - r_0\|_{C[0,L]}). \quad (4.37)$$

Применим теорему 9 из [151], учитывая (4.37), и получим оценку

$$|t_k(l) - t_0(l)| \leqslant (e^{l\|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}u_{\min}^{-2}} - 1)\|r_k - r_0\|_{C[0,L]}. \quad (4.38)$$

□

Маршрут, как и в предыдущем разделе, должен быть близок к некоторому заранее заданному «идеальному» маршруту $\rho \in \mathcal{Y}$. Таким образом, задан функционал $J_{\rho,\infty} : \mathcal{Y} \to \mathbb{R}$, $J_{\rho,\infty}(r) = \|\rho - r\|_{C([0,L];\mathbb{R}^2)}$ или $J_{\rho,p} : \mathcal{Y} \to \mathbb{R}$, $J_{\rho,p,m}(r) = \|\rho - r\|_{W_p^m(0,L;\mathbb{R}^2)}$, $p \geqslant 1$, $m \in \{0,1\}$. Непрерывность и минимизируемость таких функционалов и суммы $\alpha_1 J_{\rho,p} + \alpha_2\Theta$ очевидна, так как

$$|J_{\rho,p}(r_k) - J_{\rho,p}(r_0)| = \|\rho - r_k\|_H - \|\rho - r_0\|_H \leqslant \|r_k - r_0\|_H,$$

где $H = C([0,L];\mathbb{R}^2)$ или $H = W_p^m(0,L;\mathbb{R}^2)$.

4.3 Дискретная модель

4.3.1 Оценки аппроксимации

На практике выяснилось, что решение указанных в предыдущем разделе задач оптимизации представляет существенную проблему в виду нетривиальной структуры как минимизируемого функционала, так и дифференциальной связи. Также, роботам и подобным агентом как правило неизвестен способ изменения окружения в отдаленном будущем и глобальное планирование маршрутов становится не вполне применимым. Поэтому больший интерес представляет поиск оптимального маршрута в малой подобласти $\Omega_m \subset \Omega$ с последующей склейкой оптимальных в Ω_0 маршрутов в один, при этом возможно пользоваться не только классическими методами решения задач оптимального управления, но и различными численными методами. Автором была разработана программа моделирования движения и боевых действий иерархически организованных агентов «Бокоход» [35], в которой реализован поиск оптимального маршрута для группы агентов. Похожая система управления описана в [153], а сходная среда моделирования, но ориентированная не на моделирование конфликта организованных групп агентов, а на возникающую в условиях чрезвычайной ситуации панику – в статье [154]. Рассмотрим минимальное покрытие области Ω системой одина-

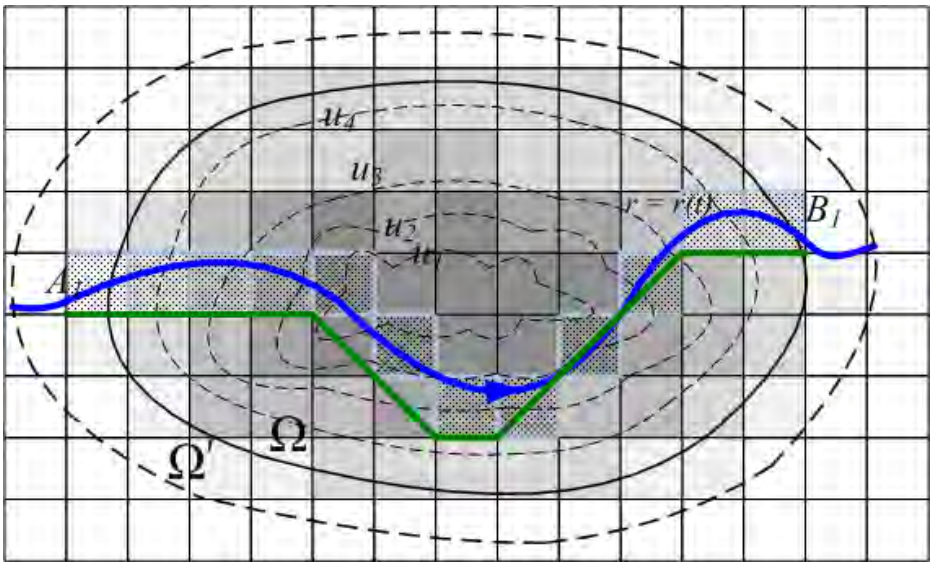


Рисунок 4.2 — Маршрут и его приближение

ковых квадратов со стороной h:

$$\mathbf{\Omega}_h = \{\omega_{ij} | (i,j) \in I_h \subset \mathbb{N}^2,\ \mu(\omega_{ij} \cap \omega_{kl}) = \varnothing, (i,j) \neq (k,l), \Omega \subseteq \bigcup_{(i,j)\in I_h} \omega_{ij}\},$$

$$\Omega_h = \bigcup_{(i,j)\in I_h} \omega_{ij}$$

Будем обозначать координаты левого нижнего угла ω_{ij} через (a_i,a_j). Разобьем также отрезок $[0,T]$ на N отрезков одинаковой длины τ и введем дискретное время $k \in T_\tau = \{0,1,\ldots,N-1\}$, $t_k = k\tau$. Всюду в разделе полагаем, что $u \in C([0,T] \times \Omega')$. Функцию u можно в каждый момент времени $t \in [0,T]$ приблизить по формуле

$$u_{\tau,h}(t,x,y) = \sum_{(k;i,j)\in T_\tau \times I_h} u_{ij}^k(h,\tau)\chi_{ij}^k(t,x,y),$$

где $\chi_{ij}^k : [0,T] \times \Omega_h \to \{0,1\}$ – характеристическая функция множества $[t_k,t_{k+1}] \times \omega_{ij}$, $u_{ij}^k(h,\tau) = u(t_k,a_i,a_j)$ – константы. Будем полагать Ω' таковым, что $\Omega_h \subset \Omega'$.

Множество всех абсцисс левого нижнего угла квадратов разбиения $\mathbf{\Omega}_h = \{a_i | i \in I_h^x\}$ будем обозначать через $\mathbf{\Omega}_h^x$, а ординат – через $\mathbf{\Omega}_h^y = \{a_j | j \in I_h^y\}$.

Обозначим как $\mathcal{Y}(h) \subset \mathcal{Y}$ множество всех непрерывных ломаных с вершинами в точках из множества $\mathbf{\Omega}_h^x \times \mathbf{\Omega}_h^y$. Такие ломаные мы будем называть клеточными графиками.

Опишем алгоритм приближения $r = (x(l),y(l)) \in \mathcal{Y}$ клеточным графиком r_h (см. фиг. 4.2). Для упрощения полагается, что клетки достаточно мелкие и функция $y = y(x)$ определена внутри каждой клетки и имеет внутри клетки не более одного экстремума.

1. Ищутся такие l_i, что $x(l_i) = a_i$, $a_i \in \mathbf{\Omega}_h^x$. Обозначим множество таких l_i через A^x. Ищутся такие l_j, что $y(l_j) = a_j$, $a_j \in \mathbf{\Omega}_h^y$. Обозначим множество таких l_j через A^y. Упорядочиваем множество $A = A^x \cup A^y$ по возрастанию: $l_j \leqslant l_{j+1}$ для всех $l_i \in A$ и перенумеровываем его элементы подряд, начиная с единицы.
2. Устанавливаем $i = 1$, $d = (0,0)$.
3. Выбираем $l_i \in A$.
4. Если $l_i \in A^x$ и существует $l_n \in A^x$, $l_n > l_i$, такой что $\Delta_i x = |x(l_n) - x(l_i)| = h$ и $\Delta_i y = |y(l_n) - y(l_i)| \leqslant h$ и все разности $x(l_{k+1}) - x(l_k)$ при $i \leqslant k \leqslant n-1$ имеют один знак и $|y(l_{k+1}) - y(l_k)| < h$ при $i \leqslant k \leqslant n-1$, то

$d_1 = \frac{\Delta_i x}{|\Delta_i x|}$;

если $a_j \in \mathbf{\Omega}_h^y$, $a_j \leqslant y(l_i) < a_j + h$ и $a_j + h < y(l_n)$, то $d_2 = 1$, если $a_j > y(l_n)$, то $d_2 = -1$, иначе $d_2 = 0$;

устанавливаем $a(l_i) = (x(l_i),a_j)$;

устанавливаем $i_{next} = n$ и переходим к шагу 7.

5. Если $l_i \in A^y$ и существует $l_n \in A^y$, $l_n > l_i$, такой что $\Delta_i y = |y(l_n) - y(l_i)| = h$ и $\Delta_i x = |x(l_n) - x(l_i)| < h$ и все разности $y(l_{k+1}) - y(l_k)$ при $i \leqslant k \leqslant n-1$ имеют один знак и $|x(l_{k+1}) - x(l_k)| < h$ при $i \leqslant k \leqslant n-1$, то

   $d_2 = \frac{\Delta_i y}{|\Delta_i y|}$;

   если $a_j \in \mathbf{\Omega}_h^x$, $a_j \leqslant x(l_i) < a_j + h$ и $a_j + h < x(l_n)$, то $d_1 = 1$, если $a_j > x(l_n)$, то $d_1 = -1$, иначе $d_1 = 0$;

   устанавливаем $a(l_i) = (a_j, y(l_i))$;

   устанавливаем $i_{next} = n$ и переходим к шагу 7.

6. Помещаем l_i в список точек соединения A^c и устанавливаем $i_{next} = i+1$ и переходим к шагу 8.
7. Устанавливаем $d = (d_1, d_2)$. Приближаем на отрезке $[l_i,l_n]$ функцию r функцией $r_h(l) = a(l_i) + hd\frac{l-l_i}{l_n-l_i}$.
8. Устанавливаем $i = i_{next}$.
9. Если $l_i \in A$, переходим к шагу 3, иначе – к шагу 10.
10. Соединяем получившиеся куски клеточного графика. Для этого для всех $l_i \in A^c$ выполняем следующие шаги.
11. Если $|\Delta_i x| = |x(l_{i+1}) - x(l_i)| < h/2$ и $|\Delta_i y| = |y(l_{i+1}) - y(l_i)| < h/2$, то

    если $\Delta_i x \neq 0$, то $d_1 = \frac{\Delta_i x}{|\Delta_i x|}$, иначе $d_1 = 0$;

    если $\Delta_i y \neq 0$, то $d_2 = \frac{\Delta_i y}{|\Delta_i y|}$, иначе $d_2 = 0$;

    иначе

    если $l_i \in A_x$, то $d_1 = 1$, $d_2 = 0$;

    если $l_i \in A_y$, то $d_2 = 1$, $d_1 = 0$;
12. Если $l_j > l_i$ и $l_i \in A^x$, $l_j \in A^y$ и $|x(l_i) - x(l_j)| < h$ или $l_i \in A^y$, $l_j \in A^x$ и $|y(l_i) - y(l_j)| < h$, то приближаем на отрезке $[l_i,l_i + h]$ функцию r функцией $r_h(l) = a(l_i) + hd\frac{l-l_i}{l_j-l_i}$.

Шаг сетки h необходимо подобрать так, чтобы он был меньше длины любого из отрезков D_i^x или D_j^y. Далее будем использовать обозначение

$$\mathcal{D} = \{(-1,1),(-1,-1),(-1,0),(0,-1),(0,0),(1,0),(0,1),(1,1),(1,-1)\}.$$

Следует заметить, что если l имеет смысл длины $L(r)$ для кривой r, то для кривой r_h этот параметр такого смысла не имеет. Более того, при любом уменьшении шага сетки параметр l отличается от длины $L(r_h)$ кривой r_h и справедлива оценка

$$L(r_h)-L(r) = \int_0^L \sqrt{x_h'^2(l) + y_h'^2(l)}dl - \int_0^L dl = \sum_i \int_{l_i}^{l_{i+1}} (\|d_i\|-1)dl \leqslant (\sqrt{2}-1)L(r),$$

где $d_i \in \mathcal{D}$.

Из самой конструкции r_h следует, что справедлива

Лемма 7. *Пусть $r = (x(l), y(l)) \in \mathcal{Y}$, $r_h \in \mathcal{Y}(h)$ и h достаточно мало. Тогда справедлива оценка*

$$\|r - r_h\|_{C[0,L]} \leqslant h\sqrt{2}. \tag{4.39}$$

Для оценки численного метода поиска оптимального по времени маршрута понадобятся утверждения, доказывающие существование решений уравнения (4.7) с функцией проходимости u, приближенной ступенчатой u_h, $r \in \mathcal{Y}$, приближенным $r_h \in \mathcal{Y}(h)$, $r_h = (x_h(l),y_h(l))$, и дающие оценку приближенного решения. Для этого запишем уравнение (4.7) в виде задачи Коши для дифференциального уравнения с разрывной правой частью:

$$l_h'(t) = u_{\tau,h}(t,x_h(l_h(t)),y_h(l_h(t))), \tag{4.40}$$

$$l_h(0) = 0. \tag{4.41}$$

Лемма 8. *Пусть $r \in \mathcal{Y}$, $u \in C^1([0,T] \times \Omega')$, $u_{\tau,h}$ – ступенчатое приближение u. Тогда задача (4.40), (4.41) имеет абсолютно непрерывное решение $l \in L_\infty(0,T)$ и справедлива оценка*

$$|l_h(t) - l(t)| < \frac{h\sqrt{2} + \tau}{\|K\|}(e^{\|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}\|K\|} - 1). \tag{4.42}$$

Доказательство. Поскольку $|u_{\tau,h}(t,x,y)| \leqslant \|u\|_{C([0,T]\times\Omega')}$, воспользуемся Теоремой 4 из [151], в соответствии с которой задача (4.40), (4.41) имеет абсолютно непрерывное продолжаемое на $[0,T]$ решение l_h. Оценим разность этого решения с решением (4.7) в соответствии с Теоремой 9 из [151]. Пусть $(a_i, a_j) \in \mathbf{\Omega}^x \times \mathbf{\Omega}^y$,

$a_i \leqslant x_h(l) < a_i + h$, $a_j \leqslant y_h(l) < a_j + h$. С помощью формулы конечных приращений и леммы 7 получим оценку

$$|u(t,x(l),y(l)) - u_{\tau,h}(t,x_h(l),y_h(l))| = |u(t,x(l),y(l)) - u(t_k,a_i,a_j)| \leqslant$$
$$\leqslant \|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}(\|r(l)-(a_i,a_j)\|+|t-t_k|) = \|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}(\|r-r_h\|_{C[0,L]}+|t-t$$
$$\leqslant \|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}(h\sqrt{2}+\tau). \quad (4.43)$$

Используя неравенство (4.12), получаем оценку

$$|l_h(t) - l(t)| < \varepsilon(t), \quad (4.44)$$

где ε – решение задачи

$$\varepsilon'(t) = \|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}\|K\|\varepsilon(t) + \|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}(h\sqrt{2}+\tau), \quad (4.45)$$
$$\varepsilon(0) = 0. \quad (4.46)$$

Решая (4.45), (4.46), получим из (4.44) оценку

$$|l_h(t) - l(t)| < \frac{h\sqrt{2}+\tau}{\|K\|}(e^{\|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}\|K\|} - 1). \quad (4.47)$$

□

Лемма 9. *Пусть $r_h \in \mathcal{Y}(h)$ – приближение r, для которого выполнена оценка леммы 7. Тогда имеет место $r_h \circ l_h \to r \circ l$ в $C([0,T],\mathbb{R}^2)$ при $h \to 0$, $\tau \to 0$.*

Доказательство. Аналогично лемме 4 получим из неравенства (4.47) и леммы 7 оценку

$$|r(l(t)) - r_h(l_h(t))| \leqslant |r(l(t)) - r(l_h(t))| + |r(l_h(t)) - r_h(l_h(t))| \leqslant$$
$$\leqslant \|K\||l(t)-l_h(t)|+\|r_h-r\|_{C([0,L],\mathbb{R}^2)} \leqslant (h\sqrt{2}+\tau)(e^{\|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}\|K\|}-1)+h\sqrt{2}. \quad (4.48)$$

□

Определим функционал $\Theta_h : \mathcal{Y}(h) \to \mathbb{R}$ аналогично функционалу Θ, то есть

$$\Theta_h(r_h) = \min\{t \in [0,T], t \neq 0 | r_h(l_h(t)) \in \partial\Omega_h\}.$$

Поскольку Θ_h определен на конечном множестве, то он, очевидно, достигает на нем минимума. Выпишем аналогично (4.3) общий вид функционала Θ_h. Выберем $r_h \in \mathcal{Y}(h)$ и сопоставим ему множество $R_h \subseteq \mathbf{\Omega}_h$, $R_h = \{\omega_{ij} | (i,j) \in I(r_h)\}$,

такое что для любого $\omega_{ij} \in R_h$, $\omega_{ij} = [a_i,a_i+h] \times [a_j,a_j+h]$ существует l_k, $l_m \in [0,L]$, такие что $r_h(l) = (a_i,a_j) + d_{ij}h\frac{l-l_k}{l_{k+1}-l_k}$, $l \in [l_k,l_{k+1}]$, $d_{ij} \in \mathcal{D}$. Пусть также r_h попадает в квадрат $\omega_{ij} \in R_h$ в момент времени $k\tau \leqslant t < (k+1)\tau$ и $t(l_{k+1}) - t(l_k) \leqslant \tau$. Тогда, если положить $L_h = l_h(\Theta_h(r_h))$

$$\Theta_h(r_h) = \int_0^{L_h} \frac{dl_h}{u_{\tau,h}(t(l_h),x_h(l_h),y_h(l_h))} = \sum_k \int_{l_k}^{l_{k+1}} \frac{dl_h}{u_{\tau,h}(t(l_h),x_h(l_h),y_h(l_h))} = \\ = \sum_{(i,j)\in I(r_h)} \frac{h\|d_{ij}\|}{u_{ij}^k(h,\tau)}. \quad (4.49)$$

Пусть $\mathcal{Y}_\infty = \bigcup_{h>0} \mathcal{Y}(h)$. Очевидно, что $\mathcal{Y}_\infty \subset \mathcal{Y}$ плотно по норме $C[0,L]$. Докажем аналог, в некотором смысле, теоремы 1.

Лемма 10. *Пусть* $r_h \to r_*$ *в* $C([0,L],\mathbb{R}^2)$, $p > 1$, *при* $h \searrow 0$, $\tau \searrow 0$, r_h, $r_* \in \mathcal{Y}_\infty$, $h > 0$. *Тогда* $\Theta_h(r_h) \to \Theta(r_*)$ *и* $\Theta(r_h) \to \Theta(r_*)$.

Доказательство. Покажем, что если $r_h \to r_*$, $r_* \in \mathcal{Y}(h_*)$, сходится в $C([0,L],\mathbb{R}^2)$, то $r_h \to r_*$ сходится в $W_p^1([0,L],\mathbb{R}^2)$. Действительно, $r_h'(l) = d_h(l)$, $r_*'(l) = d_*(l)$, $d_h(l)$, $d_*(l) \in \mathcal{D}$, $l \in (l_1,l_2)$, $r_h(l_1)$, $r_h(l_2) \in \omega_{ij} \in \mathbf{\Omega}_{h_*}$ и

$$\|r_h'(l) - r_*'(l)\| = \|d_h(l) - d_*(l)\| = \left\| \frac{r_h(l_2) - r_h(l_1)}{l_2 - l_1} - \frac{r_*(l_2) - r_*(l_1)}{l_2 - l_1} \right\| \leqslant \\ \leqslant 2h^{-1}\|r_h - r_*\|_{C([0,L],\mathbb{R}^2)} \leqslant 2h_*^{-1}\|r_h - r_*\|_{C([0,L],\mathbb{R}^2)} \to 0. \quad (4.50)$$

Из теоремы 1 следует, что $\Theta(r_h) \to \Theta(r_*)$. Остается только оценить $|\Theta_h(r_h) - \Theta(r_h)|$. Из конструкции Ω_h очевидно, что если $\omega_{ij} \in \mathbf{\Omega}_h$, ω_{ij}^0 – внутренность множества ω_{ij}, $\Gamma_{ij} = \partial\Omega \cap \omega_{ij}^0 \neq \varnothing$, $\Gamma_{ij}^h = \partial\Omega_h \cap \omega_{ij}^0$, то $\max_{r_1\in\Gamma_{ij}, r_2\in\Gamma_{ij}^h}(r_1,r_2) < h\sqrt{2}$. Между точками $r_1 \in \Gamma_{ij}$, $r_2 \in \Gamma_{ij}^h$ кривая r_h будет представлять собой отрезок прямой $r_{h,i,j}(l) = (a_i,a_j) + dl$, $d \in \mathcal{D}$ и при достаточно малом h время прохождения этого отрезка оценивается по формуле

$$t_{h,i,j} < \frac{h\sqrt{2}}{u_{ij}^k(h,\tau)}. \quad (4.51)$$

Пусть $t_1 = \Theta(r_h)$, $t_2 = \min\{t \in [0,T], t \neq 0 | r_h(l_h(t)) \in \partial\Omega\}$. Пусть

$$C_6 = \left\| \sum_{i=1}^3 \lambda_i (r_h \circ l)_+'(\xi_i) \right\|, \ \xi \in (t_2,t_1).$$

В силу обобщения формулы конечных приращений для вектор-функций, оценки (4.48) и очевидного факта $r_h(l_h(t_2)) = r_h(l(t_1))$ получим

$$|t_2 - t_1| = C_6^{-1}\|r_h(l(t_2)) - r_h(l(t_1))\| \leqslant C_6^{-1}(\|r_h(l(t_2)) - r_h(l_h(t_2))\| + \\ + \|r_h(l_h(t_2)) - r_h(l(t_1))\|) \leqslant u_{\min}^{-1}(h\sqrt{2} + \tau)(e^{\|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}\|K\|} - 1), \quad (4.52)$$

где $u_{\min} = \min_{t\in[0,T],(x,y)\in\Omega} u(t,x,y)$. Из оценок (4.51), (4.52) следует, что

$$|\Theta_h(r_h) - \Theta(r_h)| = |t_2 - t_1| + t_{h,i,j} < u_{\min}^{-1}h\sqrt{2} + u_{\min}^{-1}(h\sqrt{2} + \tau)(e^{\|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}\|K\|} - 1). \quad (4.53)$$

Из оценки (4.53) и следует утверждение леммы. □

Наконец, докажем главное утверждение раздела, говорящее, что при достаточно малых h оптимальный клеточный маршрут будет близок к настоящему оптимальному маршруту.

Теорема 3. *Существуют такие $r_h^* \in \mathcal{Y}(h)$, минимизирующие функционал Θ_h, и такой $r_0^* \in \mathcal{Y}$, минимизирующий функционал Θ, что $r_h^* \to r_0^*$ в $C([0,L],\mathbb{R}^2)$ при $h \to 0$.*

Доказательство. Так как $\mathcal{Y}_\infty \subset \mathcal{Y}$ ограничено в $W_p^1(0,L;\mathbb{R}^2)$ и $W_p^1(0,L;\mathbb{R}^2) \subset C([0,L],\mathbb{R}^2)$ компактно, то в силу леммы 10 и теоремы Вейерштрасса функционал Θ достигает на $\mathcal{Y}_\infty$ минимума в точках $\{r_\xi^*\} \subset \mathcal{Y}_\xi \subset \mathcal{Y}_\infty$ и справедлива оценка

$$\min_{r\in\mathcal{Y}} \Theta(r) \leqslant \min_{r\in\mathcal{Y}_\infty} \Theta(r) = \min_{r_\xi\in\mathcal{Y}_\xi} \Theta(r_\xi), \quad (4.54)$$

Пусть $r_0^* \in \mathcal{Y}$, $\Theta(r_0^*) \to \min$. В силу леммы 7 существует $r_\xi^{**} \in \mathcal{Y}_\xi$, такой что справедлива оценка $\|r_0^* - r_\xi^{**}\|_{C([0,L];\mathbb{R}^2)} \leqslant \xi\sqrt{2}$. Остается только показать, что $\Theta(r_\xi^{**}) = \Theta(r_\xi^*)$. Действительно, в силу оценки леммы 6 функционал $\Phi : \mathcal{Y}(h) \to \mathbb{R}$ удовлетворяет соотношению

$$0 \leqslant \Phi(r_h) = t_h(L_h) - t_0^*(L_0^*) = t_h(L_0^*) - t_0^*(L_0^*) + (t_h(L_h) - t_h(L_0^*)) \leqslant \\ \leqslant (e^{L_0^*\|\nabla_{(t,x,y)}u\|_{C([0,T]\times\Omega)}u_{\min}^{-2}} - 1)\|r_h - r_0^*\|_{C[0,L]} + (t_h(L_h) - t_h(L_0^*)), \quad (4.55)$$

где L_0^*, L_h – длины дуг кривых r_0^* и r_h соответственно до пересечения их с границей, минимизируется в $\mathcal{Y}(h)$ при r_h, являющимся клеточным приближением r_0^*, причем $\Phi^*(r_0^*) = 0$. □

Функционал $J^h_\rho : \mathcal{Y}_h \to \mathbb{R}$ близости к «идеальному» маршруту ρ удобно задавать как число квадратов, в области Λ, ограниченной кривой r_h, клеточным приближением «идеального маршрута» ρ_h и границей области $\partial\Omega_h$. Вполне очевидно, что J^h_ρ и $\alpha_1 J^h_\rho + \alpha_2 \Theta_h$ имеют минимальное значение в $\mathcal{Y}_h$. С помощью формулы Грина возможно показать связь ранее введенного функционала близости $J_{\rho,p,1}$ и J^h_ρ, выразив площадь Λ через криволинейные интегралы по $\partial\Lambda$. Функционал близости $J_{\rho,\infty}$ приближается функционалом $J^h_{\rho,\infty} : \mathcal{Y}(h) \to \mathbb{R}$, $J^h_{\rho,\infty}(r_h) = \|\rho_h - r_h\|_{C([0,L];\mathbb{R}^2)}$, где ρ_h – клеточное приближение ρ. Отметим, что в силу конструкции клеточных приближений имеем $|J^h_{\rho,\infty}(r_h) - J_{\rho,\infty}(r)| = |\|\rho_h - r_h\|_{C([0,L];\mathbb{R}^2)} - \|\rho - r\|_{C([0,L];\mathbb{R}^2)}| \leqslant \|\rho_h - r_h - \rho + r\|_{C([0,L];\mathbb{R}^2)} \leqslant \|\rho - \rho_h\|_{C([0,L];\mathbb{R}^2)} + \|r - r_h\|_{C([0,L];\mathbb{R}^2)} \leqslant 2h\sqrt{2}$.

4.3.2 Построение клеточного автомата

Для упрощения моделирования движения агентов необходимо перейти к дискретному времени в ходах несколько отличным от прежнего способом. Положим

$$t_h = \frac{h}{\max_{(i,j)\in I_h, k\in T_\tau} u^k_{ij}(h,\tau)}$$

и определим, что квадрат ω_{ij} по состоянию на момент времени $k\tau \in [0,T]$ проходится в недиагональном направлении за $m_{ij}(k)$ ходов, если

$$m_{ij}(k) = \frac{h}{t_h u^k_{ij}(h,\tau)} = \frac{\max_{(i,j)\in I_h, k\in T_\tau} u^k_{ij}(h,\tau)}{u^k_{ij}(h,\tau)}.$$

Далее возможно перейти к целым значениям $m_{ij}(k)$, отбросив дробную часть и взяв $\tilde{m}_{ij}(k) = [m_{ij}(k)]$. Для компенсации ошибки округления, возникающей при этом, агенту, находящемуся в квадрате $\omega_{i,j}$ сопоставляется величина $errc_{ij}$ накапливающейся ошибки дискретного времени, а квадрату ω_{ij} – величина ошибки $err_{ij} = \{m_{ij}(k)\}$. При переходе агента в следующий квадрат $\omega_{i'j'}$ величина $errc_{i'j'}$ увеличивается на $err_{i'j'}$, устанавливается $errc_{ij} = 0$, а если становится $errc_{i'j'} > 1$, то агент пропускает ход независимо от проходимости квадрата, в котором находится, и устанавливает $errc_{i'j'} = errc_{i'j'} - 1$.

Рассмотрим клеточный автомат со множеством ячеек в $\mathbb{Z}^2$. Состояние ячейки (i,j) описывается вектором

$$o_{ij} = (f_{ij}, d_{ij}, w_{ij}, \tilde{m}_{ij}, err_{ij}, errc_{ij}),$$

где f_{ij} – показатель занятости ячейки, $d_{ij} \in \mathcal{D}$, w_{ij} – число ходов, проведенных в ячейке, остальные величины упоминались выше. Агент может переместиться в соседнюю клетку только если $w_{ij} = \tilde{m}_{ij}$, при первом попадании агента в клетку устанавливается $w_{ij} = 0$ и каждый ход к w_{ij} добавляется единица.

Задаются вероятностные правила перехода, которые фактически генерируют некоторый клеточный маршрут, который приближается к оптимальному в смысле ранее построенных функционалов вида (4.49)

$$\Theta_h(r_h) = \sum_{(i,j)\in I(r_h)} \frac{h\|d_{ij}\|}{u_{ij}^k(h,\tau)} = t_h^{-1} \sum_{(i,j)\in I(r_h)} \|d_{ij}\| m_{ij}. \tag{4.56}$$

Для этого вероятность смещения агента в направлении $d_{ij} \in \mathcal{D}$ труднопроходимых клеток уменьшается. Более того, для того чтобы избежать полного перебора маршрутов в достаточно большой области Ω_h, выбирается окрестность $\Omega_h(i,j;N)$ квадрата ω_{ij}, занимаемого агентом, диаметром в N квадратов, и в этой окрестности ищется оптимальный маршрут. Если N мало, то возможен полный перебор всех возможных маршрутов и применение методов типа A^*, иначе используются методы типа случайной оптимизации (см. [155]), т. е. по вышеупомянутым правилам генерируется достаточно большое количество маршрутов, выходящих на границу окрестности и среди них ищутся минимизирующие функционал Θ_h. Если в $\Omega_h(i,j;N)$ содержится конец ω_{fg} полного клеточного маршрута, то вместо маршрутов, выходящих на границу окрестности, используются маршруты, заканчивающиеся в ω_{fg}.

Особо эффективен этот метод будет в случае, когда значения ступенек u_{ij}^k являются независимыми равномерно распределенными случайными величинами. Построим в $\Omega_h(i,j;N)$ множество из m случайных блужданий вдоль образцового маршрута ρ, которое обозначим через $\mathcal{Y}(h;m,\rho)$. Случайное блуждание будет иметь вид

$$\alpha_n = \alpha_0 + \sum_{j=1}^{n} d_j, d_j \in \mathcal{D}, 0 \leqslant j \leqslant n,$$

причем заданы вероятности перемещения в следующую клетку

$$p_d(\omega_\beta) = P\{\omega_{\alpha_{n+1}} = \omega_{\alpha_n+d} | \omega_{\alpha_n} = \omega_\beta\}.$$

Эти вероятности можно задать специальным образом, увеличивая их в направлении, приближающем к образцовому маршруту ρ_h и уменьшая – в направлении особо труднопроходимых препятствий. Далее мы минимизируем функционал Θ_h на $\mathcal{Y}(h;m,\rho)$, находя клеточный маршрут $r^*_{h,N}$ и склеиваем из таких оптимальных клеточных маршрутов в каждой окрестности маршрут во всей области.

Упомянем тот случай, когда в Ω одновременно перемещаются несколько агентов из множества агентов $E = \{e_1,\ldots,e_n\}$. Пусть $r_{e_i} = (x_{e_i},y_{e_i})$,

$$u(t,r_{e_i}(l_{e_i}(t))) = u(t,x_{e_i}(l_{e_i}(t)),y_{e_i}(l_{e_i}(t))),$$

$I : [0,T] \to \{1,\ldots,n\}$ – случайная функция, которая сопоставляет в момент времени $t \in [0,T]$ агентам $e_{i_1},\ldots,e_{i_m}$ из E, таким что $r_{e_{i_1}}(l_{e_1}(t)) = \cdots = r_{e_{i_m}}(l_{e_m}(t))$ одно число из $\{i_1,\ldots,i_m\}$. Движение агентов может быть описано системой уравнений

$$l'_{e_i}(t) = U_i(t,r_{e_1}(l_{e_1}(t)),\ldots,r_{e_n}(l_{e_n}(t))), \quad 1 \leqslant i \leqslant n, \tag{4.57}$$

$U_i(t,r_{e_1}(l_{e_1}(t)),\ldots,r_{e_n}(l_{e_n}(t))) = u(t,r_{e_i}(l_{e_i}(t)))$, если $\|r_{e_i}(l_{e_i}(t)) - r_{e_j}(l_{e_j}(t))\| \geqslant R_0$, $1 \leqslant i,j \leqslant n$ или $I(t) = i$ и $U_i(t,r_{e_1},\ldots,r_{e_n}) = 0$ во всех прочих случаях. Система (4.57) имеет решение, для которого, в общих чертах, верны многие из утверждений настоящей главы, так как утверждения из [151] справедливы и для систем обыкновенных дифференциальных уравнений. Для поиска оптимального по времени решения потребуется заменить условие $U_i = 0$ на $U_i = \varepsilon > 0$, $\varepsilon \ll u_{\min}$ или применять метод штрафа.

В рамках клеточно-автоматной модели это будет соответствовать тому, что если в какой-то такт функционирования клеточного автомата на одну ячейку будут претендовать несколько агентов, то случайно выбранный агент займет ячейку, а остальные должны будут избрать другое направление и вероятности перемещения в клетку, занятую другим агентом, равны нулю.

При рассмотрении движения группы агентов вместо функционала $\alpha_1 J_{\rho,p} + \alpha_2\Theta$ можно минимизировать функционал

$$\alpha_1 J_{E,\rho,p}(r_{e_1},\ldots,r_{e_n}) + \alpha_2\Theta_E(r_{e_1},\ldots,r_{e_n}) = \alpha_1\sum_{i=1}^{n} J_{\rho,p}(r_{e_i}) + \alpha_2\sum_{i=1}^{n}\Theta(r_{e_i}).$$

Однако, минимизация такого функционала предполагает, что все агенты полностью осведомлены о том, каким именно маршрутом идет каждый агент из E, что может быть чересчур жестким требованием. Поэтому возможно также предположить, что каждый агент ищет кратчайший маршрут самостоятельно,

но задан функционал близости агентов из E, например как $F(r_{e_1},\ldots,r_{e_n}) = \max_{1\leqslant i,j\leqslant n}\|r_{e_i}-r_{e_j}\|_H$, где $H = C([0,L];\mathbb{R}^2)$ или $H = W_p^m(0,L;\mathbb{R}^2)$.

4.4 Задача многоагентного поиска оптимального пути

В статье [156] обсуждается многоагентный поиск пути (multi-agent pathfinding, MAPF), т.е. задачу поиска путей без коллизий для заданного количества агентов из их заданных начальных позиций в заданные целевые позиции в заданной среде. Авторы указывают, что задачи MAPF, как правило, возникают для авиационных буксировочных машин, офисных роботов, персонажей видеоигр, складских роботов и т. д. Однако агенты обычно не могут выполнять свои планы MAPF как запланировано, поскольку они часто проходят свои пути медленнее, чем предполагалось. Чтобы решить эту проблему, авторы формализуют проблему MAPF с вероятностями задержки и обсуждают несколько политик выполнения плана MAPF.

В настоящем разделе будет построена аналитическая теория многоагентного поиска пути, для чего потребуется обобщить полученные в предыдущем разделе результаты на движение нескольких агентов.

4.4.1 Постановка задачи

Пусть имеется m агентов, которые движутся по области Ω, $\partial\Omega \in C^1$, по маршрутам $r_k : [0,T] \to \Omega$ из точек $A_k \in \partial\Omega$ в точки $B_k \in \partial\Omega$. Максимальная скорость агента в точке x в момент времени t ограничена функцией проходимости $u \in C([0,T],C^1(\Omega))$, $u : [0,T] \times \Omega \to \mathbb{R}_+$. Агенты стремятся двигаться с максимальной возможной скоростью:

$$\|\dot{r}_k(t)\| = u(t,r_k(t)), \tag{4.58}$$

Необходимо минимизировать время $\Theta_k(r_k)$, которое агент тратит на прохождение маршрута r_k:

$$\Theta_k(r_k) \to \min\,, \quad k = \overline{1,m} \text{ or} \tag{4.59}$$

$$\max_{k=\overline{1,m}} \Theta_k(r_k) \to \min .$$

Кроме того, мы полагаем, что агенты имеют радиус ε, и их траектории не могут пересекаться в один и тот же момент:

$$\forall (i,j \in \overline{1,m}, i \neq j) \quad \|r_i(t) - r_j(t)\| > \varepsilon, t \in [0,T]. \tag{4.60}$$

Видно, что условия (4.58) и (4.60) взаимно противоречат. Например, Ω может состоять из двух частей с одним тонким мостом, ширина которого равна 2ε, и два агента не могут одновременно перемещаться по этому мосту с максимальной скоростью.

Чтобы устранить противоречие, мы заменяем условие (4.58) одним из следующих условий с торможением

$$\|\dot{r}_k(t)\| = u(t,r_k(t)) - v_{k,\varepsilon}(t, r_1(t), \ldots, r_m(t)), \tag{4.61}$$

$$v_{k,\varepsilon}(t, r_1(t), \ldots, r_m(t); \mathcal{I}^k) = \begin{cases} u(t,r_k(t)), & \exists (i \in \mathcal{I}^k) \quad \|r_k(t) - r_i(t)\| \leqslant \varepsilon, \\ 0, & \min\limits_{i \in \mathcal{I}^k} \|r_k(t) - r_i(t)\| > \varepsilon, \end{cases}$$

или, если нам нужна более гладкая правая часть уравнения движения (4.61),

$$v_{k,\varepsilon}(t, r_1(t), \ldots, r_m(t); \mathcal{I}^k) = \begin{cases} u(t,r_k(t)), & \xi_k(r_1(t), \ldots, r_m(t)) \leqslant \varepsilon, \\ \sigma_\varepsilon(\alpha,\xi_k(r_1(t), \ldots, r_m(t))), & \varepsilon + \alpha \geqslant \xi_k(r_1(t), \ldots, r_m(t)) > \\ 0, & \xi_k(r_1(t), \ldots, r_m(t)) > \varepsilon + \alpha, \end{cases}$$

где, например, $\mathcal{I}^k = \{\overline{1,m}, i \neq k\}$ и

$$\sigma_\varepsilon(\alpha,\xi) = \frac{\xi - \alpha}{\varepsilon - \alpha} u(t, r_k(t)),$$

или

$$\sigma_\varepsilon(\alpha,\xi) = u(t, r_k(t)) \left(\frac{1}{2} + \frac{1}{2} \mathrm{erf} \left(\frac{z_0(\alpha + 2\varepsilon - 2\xi)}{\alpha} \right) \right),$$

$$z_0 = \min\{z | \mathrm{erf}(z) = 1\}.$$

Множество $\mathcal{I} = \{\mathcal{I}^k | k = \overline{1,m}\}$ должно удовлетворять следующим условиям:

1. Если один агент идет вперед, то другой должен уступить дорогу:

$$\forall(k=\overline{1,m}) \quad j \in \mathcal{I}^k \to k \notin \mathcal{I}^j. \tag{4.62}$$

2. Порядок перемещения задан для всех агентов. Предположим, что $\mathcal{I}^0 = \varnothing$ или $\mathcal{I}^0 = \{j \in \overline{1,m}\}$, $\mathcal{I}^j = \overline{1,m} \setminus \{j\}$. Тогда

$$\bigcup_{k=1}^{m} \mathcal{I}^k = \overline{1,m} \setminus \mathcal{I}^0. \tag{4.63}$$

Здесь $\xi_k \in C((\mathbb{R}^2)^m)$ – некоторые аггрегатные функции от $\|r_i - r_j\|$, $i,j = \overline{1,m}$ наподобие

$$\xi_k(r_1,\ldots,r_m) = \min_{i\in\mathcal{I}^k} \|r_k - r_i\|$$

или

$$\xi_k(r_1,\ldots,r_m) = \frac{1}{m-1}\sum_{i\in\mathcal{I}^k} \|r_k - r_i\|.$$

4.4.2 Маршруты и функционалы качества

Данный раздел содержит адоптацию понятий подраздела 4.2.2 для многоагентного поиска пути.

Обозначим как $\mathbf{L}_s$ произвольное упорядоченное конечное (или, возможно, счетное) множество точек $\mathbf{L}_s = \{l_i \in [0,L] | l_i < l_{i+1}, l_0 = 0, l_s = T, i \in \mathbb{N}\}$. Мы также обозначаем множество таких разбиений отрезка $[0,L]$ как $\mathcal{L}$.

Определение 21. *Обозначим через $\mathcal{Y}$ множество таких непрерывных отображений из $[0,L]$ в Ω, что для всех $r \in \mathcal{Y}$ выполняются следующие условия:*

1. $\exists(\mathbf{L}_s \in \mathcal{T})\forall(l_{i-1},l_i \in \mathbf{L}_s) \quad r|_{(l_{i-1},l_i)} \in C^2((l_{i-1},l_i);\mathbb{R}^2)$.

2. Множество $r([0,L]) \cap \partial\Omega$ содержит не менее двух элементов.

3. Производные всех функций из $\mathcal{Y}$ равномерно ограничены (за исключением, возможно, счетного множества точек, в которых производная не существует):

$$\max_{l\in[0,L]} \|r'(l)\| \leqslant C_1. \tag{4.64}$$

4. Вторые производные всех функций из $\mathcal{Y}$ равномерно ограничены в сумме:

$$\sum_{j\in J} \max_{l\in(l_{j-1},l_j)} \|r''(l)\|^p \leqslant C_s,$$

$p > 1$, где $\{l_j | j \in J\}$ – счетное множество точек, в которых первая и вторая производные не существуют.

Мы можем вычислить длину $l(t)$ кривой $r \in \mathcal{Y}$, пройденную за время t как

$$l(t) = \int_0^t \|\dot{r}(l(s))\| ds. \tag{4.65}$$

Пример функции $l(t) = r(t)$ изображен на рис. 4.3, где мы полагаем, что $y(t) = 0$.

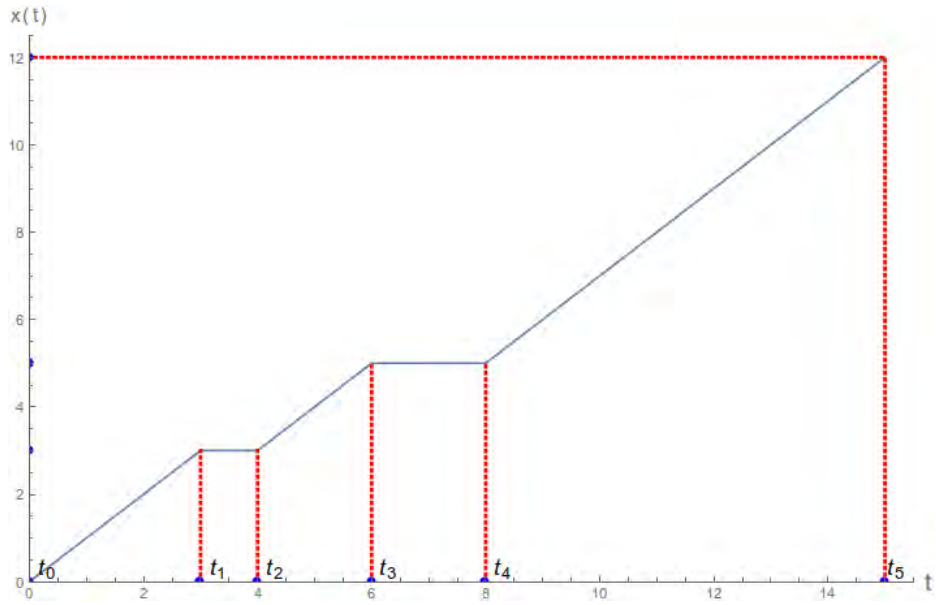


Рисунок 4.3 — Пример маршрута r и $l(t) = x(l(t))$

Определение 22. *Определим функционал временной стоимости кривой $\Theta : \mathcal{Y} \times C^1[0,T] \to \mathbb{R}_+$ как*

$$\Theta(r,l) = \min\{t \in (0,T] | r(l(t)) \in \partial\Omega\}.$$

4.4.3 Существование оптимального решения для одного агента

Предположим, что функции ξ_k непрерывно дифференцируемы по r_i, когда $\xi_k > \varepsilon$, так что $v_{k,\varepsilon} \in C([0,T],C^1(\Omega))$. Обозначим

$$\nu_\varepsilon(t, r_k(t)) = u(t,r_k(t)) - v_{k,\varepsilon}(t, r_1(t), \ldots, r_m(t); \mathcal{I}^k).$$

Конечно, $\nu_\varepsilon \in C([0,T],C^1(\Omega))$ является ограниченной функцией так как u – ограниченная функция.

Положим $r_k = r$. Мы можем переписать (4.61), (4.65) как

$$l(t) = \int_0^t \|\partial_s r(l(s))\| ds = \int_0^t \nu_\varepsilon(l(s), r(l(s)))ds,\ t \geqslant 0 \tag{4.66}$$

и изучить существование решения для задачи

$$\|\dot{r}(l(t))\| = \nu_\varepsilon(t, r(l(t))), \tag{4.67}$$

$$\Theta(r,l) \to \min, \quad r \in \mathcal{Y}. \tag{4.68}$$

Далее, обозначим $\Theta(r,l) = \Theta(r)$ для l, являющегося решением (4.66).

Также, можно переписать (4.67) в форме

$$\dot{r}(l(t)) = U(t,r(l(t))) = \begin{bmatrix} \operatorname{sgn}(\dot{x}(l(t))\sqrt{\nu_\varepsilon^2(t,r(l(t))) - \dot{y}^2(l(t))} \\ \operatorname{sgn}(\dot{y}(l(t))\sqrt{\nu_\varepsilon^2(t,r(l(t))) - \dot{x}^2(l(t))} \end{bmatrix} \tag{4.69}$$

в попытке применить общую теорию нелинейных дифференциальных уравнений, но это почти бесполезно из-за разрывности и сложности U.

В предыдущем разделе было показано, что уравнение (4.7) имеет единственное решение в $C^1[0,T]$. Далее в этом разделе через l_n будет обозначаться решение (4.7) при $r = r_n$.

Также в предыдущем разделе было показано, что задача (4.67), (4.68) имеет решение (теорема 2) для одного агента, движущегося без коллизий.

Для решения задачи (4.67), (4.68) в случае коллизий агентов, мы должны дополнительно рассмотреть случай

$$0 \leqslant \min_{t\in[0,T], r\in\Omega} \nu_\varepsilon(t,r) < \eta, \quad \eta > 0.$$

Далее, будем считать, что l является натуральным параметром кривой r. На самом деле, если r – траектория некоторого объекта, то $\|\dot{r}(l(t))\| = 0$ означает, что объект стоит на месте.

Мы должны поставить дополнительные требования к функции ν_ε. Если $\nu_\varepsilon(t,r) = 0$, $t \in [0,T]$, то существует интервал $(t_1,t_2) \in [0,T]$, такой что $t \in (t_1,t_2) \vee t = t_1 = 0 \vee t = T = t_2$ и

$$\forall(t_0 \in (t_1,t_2))\forall(r \in \mathcal{Y}) \quad \nu_\varepsilon(t_0,(r \circ l)(t_0)) = 0$$

или существуют $t_1 < t \vee t = t_1 = 0$, $t_2 > t \vee t = t_2 = T$, t_1, $t_2 \in [0,T]$, такие что

$$\forall(t_0 \in (t_1,t) \cup (t,t_2))\forall(r \in \mathcal{Y}) \quad \nu_\varepsilon(t_0,(r \circ l)(t_0)) \neq 0.$$

Кроме того, граница Ω всегда проходима:

$$\forall(t \in (0,T])\forall(r \in \partial\Omega) \quad \nu_\varepsilon(t,(r \circ l)(t)) > 0. \tag{4.70}$$

Множество всех непересекающихся интервалов $(t_{i-1},t_i) \subset [0,T]$ на которых $\nu_\varepsilon(t,r(l(t))) = 0$ будет обозначаться как $\mathcal{Z}(r)$, множество всех непересекающихся интервалов $[0,T]$ где $\nu_\varepsilon(t,r(l(t))) \geqslant \eta > 0$ – как $\mathcal{E}(r)$. Также, обозначим как $\mathcal{N}(r)$ множество всех непересекающихся интервалов $[0,T]$ где $0 < \nu_\varepsilon(t,r(l(t))) < \eta$.

Лемма 11. *Предположим, что* $r_n \to r_0$ *в* $W^1_p([0,L];\mathbb{R}^2)$, $p > 1$, $\{r_n\}_{n=1}^\infty \subset \mathcal{Y}$, $r_0 \in \mathcal{Y}$, *тогда* $r_n \circ l_n \to r_0 \circ l_0$ *in* $W^1_p([0,T];\mathbb{R}^2)$.

Доказательство. Let's $r_n = (x_n,y_n)$. Покажем утверждение леммы для x_n (для y_n доказательство аналогично). Используя неравенство Гёльдера, получаем

$$\begin{aligned}
&\left(\int_0^T |\partial_t(x_n\circ l_n(t)-x_0\circ l_0(t))|^p dt\right)^{1/p} \leqslant \left(\int_0^T |\partial_{l_n}x_n(l_n(t))l_n'(t)-\partial_{l_0}x_n(l_0(t))l_0'(t)|^p dt\right)^{1/}\\
&+\left(\int_0^T |\partial_t(x_n\circ l_0(t)-x_0\circ l_0(t))|^p dt\right)^{1/p} \leqslant \left(\int_0^T |\partial_{l_n}x_n(l_n(t))l_n'(t)-\partial_{l_0}x_n(l_0(t))l_n'(t)|^p dt\right)^{1/}\\
&+\left(\int_0^T |\partial_{l_0}x_n(l_0(t))l_n'(t)-\partial_{l_0}x_n(l_0(t))l_0'(t)|^p dt\right)^{1/p}+\left(\int_0^T |\partial_t(x_n\circ l_0(t)-x_0\circ l_0(t))|^p dt\right)^{1/p}.
\end{aligned} \tag{4.71}$$

В силу теоремы Лебега о предельном переходе под знаком интеграла, получаем

$$\int_0^T |\partial_{l_n}x_n(l_n(t))l_n'(t) - \partial_{l_0}x_n(l_0(t))l_n'(t)|^p dt \to 0. \tag{4.72}$$

Принимая во внимание, что $l_0|_{\Delta\in\mathcal{Z}(r_0)} = const$, мы получаем

$$\int_0^T |\partial_t(x_n\circ l_0(t) - x_0\circ l_0(t))|^p dt = \sum_{\Delta\in\mathcal{E}(r_0)}\int_\Delta |\partial_t(x_n\circ l_0(t) - x_0\circ l_0(t))|^p dt +$$
$$+\sum_{\Delta\in\mathcal{N}(r_0)}\int_\Delta |\partial_t(x_n\circ l_0(t) - x_0\circ l_0(t))|^p dt + \sum_{\Delta\in\mathcal{Z}(r_0)}\int_\Delta |\partial_t(x_n\circ l_0(t) - x_0\circ l_0(t))|^p dt =$$
$$= \sum_{\Delta\in\mathcal{E}(r_0)}\int_\Delta |\partial_t(x_n\circ l_0(t) - x_0\circ l_0(t))|^p dt + \sum_{\Delta\in\mathcal{N}(r_0)}\int_\Delta |\partial_t(x_n\circ l_0(t) - x_0\circ l_0(t))|^p dt. \tag{4.73}$$

Для $\Delta\in\mathcal{N}(r_0)$ and $\Delta\in\mathcal{E}(r_0)$, мы имеем, что

$$\int_\Delta |\partial_t(x_n\circ l_0(t) - x_0\circ l_0(t))|^p dt = \int_{l(\Delta)} |\partial_l(x_n(l) - x_0(l))|^p |l_0'(t)|^{p-1} dl \leqslant$$
$$\leqslant \max_{t\in[0,T]} |\mathbf{v}_\varepsilon(t, r_n(l_0(t)))|^{p-1} \|x_n - x_0\|^p_{W^1_p(l(\Delta))} \leqslant \|\mathbf{v}_\varepsilon\|^{p-1}_{C(0,T;C(\Omega))} \|x_n - x_0\|^p_{W^1_p(l(\Delta))} \to 0. \tag{4.74}$$

Из свойства (4.64) множества $\mathcal{Y}$, мы имеем

$$\int_0^T |\partial_{l_0} x_n(l_0(t)) l_n'(t) - \partial_{l_0} x_n(l_0(t)) l_0'(t)|^p dt = \int_0^T |\partial_{l_0} x_n(l_0(t))(l_n'(t) - l_0'(t))|^p dt \leqslant$$
$$\leqslant \|l_n' - l_0'\|^p_{C[0,T]} \int_0^T |\partial_{l_0} x_n(l_0(t))| dt \leqslant \|l_n' - l_0'\|^p_{C^1[0,T]} \max_{l\in[0,L]} |\partial_l x_n(l)| T \leqslant$$
$$\leqslant C_1 T \|l_n' - l_0'\|^p_{C^1[0,T]} \to 0. \tag{4.75}$$

Оценивая члены в (4.71) с помощью неравенств (4.72)–(4.74), мы получаем утверждение леммы.

□

Теорема 4. *Предположим что свойство (4.70) выполнено. В этом случае, функционал* $\Theta : \mathcal{Y}\times C^1[0,T] \to \mathbb{R}_+$ *непрерывен для каждого фиксированного* $l\in C^1[0,T]$*, являющегося решением (4.7).*

Доказательство. Доказательство в сущности аналогично приведенному в теореме 2.

Пусть $\{r_n\}_{n=1}^\infty$, $r_n \to r_0$ в $W^1_p([0,L];\mathbb{R}^2)$ и $t_n = \Theta(r_n,l_n) = \Theta(r_n)$, $0 \leqslant n < \infty$. Мы используем обобщение теоремы о среднем значении Лагранжа для векторных функций (см. [150], Theorem 3, Theorem 4)

$$r_n(l_n(t_n)) - r_n(l_n(t_0)) = (t_n - t_0)\sum_{i=1}^{3}\lambda_i(r_n \circ l_n)'_+(\xi_i), \tag{4.76}$$

где $\xi_i \in (t_0,t_n)$, $\lambda_i \geqslant 0$, $\sum_{i=1}^{3}\lambda_i = 1$, $(r_n \circ l_n)'_+$ обозначает правую производную $r_n \circ l_n$.

Таким образом,

$$\|r_n(l_n(t_n)) - r_n(l_n(t_0))\| = |t_n - t_0|\left\|\sum_{i=1}^{3}\lambda_i(r_n \circ l_n)'_+(\xi_i)\right\|. \tag{4.77}$$

Отметим, что справедлива оценка

$$\frac{|l_n(t_n) - l_n(t_0)|}{\min_{(t,r)\in\Omega}\nu_\varepsilon(t,r)} \geqslant \frac{|l_n(t_n) - l_n(t_0)|}{|l'_n(\xi^*)|} = |t_n - t_0| = \frac{\|r_n(l_n(t_n)) - r_n(l_n(t_0))\|}{\left\|\sum_{i=1}^{3}\lambda_i(r_n \circ l_n)'_+(\xi_i)\right\|}, \tag{4.78}$$

где $\xi^* \in (t_0,t_n)$, $\min_{(t,r)\in\Omega}\nu_\varepsilon(t,r) > 0$.

Теперь мы должны изучить два разных случая отдельно, потому что у нас нет отделения от нуля для ν_ε.

Рассмотрим случай

$$\left\|\sum_{i=1}^{3}\lambda_i(r_n \circ l_n)'_+(\xi_i)\right\| = \left\|\sum_{i=1}^{3}\lambda_i U(\xi_i, r_n \circ l_n(\xi_i))\right\| \geqslant \varepsilon = C_4 > 0.$$

$$|t_n - t_0| \leqslant \frac{\|r_n(l_n(t_n)) - r_n(l_n(t_0))\|}{\left\|\sum_{i=1}^{3}\lambda_i(r_n \circ l_n)'_+(\xi_i)\right\|} = \frac{\|r_n(l_n(t_n)) - r_n(l_n(t_0))\|}{\varepsilon}.$$

$$|t_n - t_0| = \frac{\|r_n(l_n(t_n)) - r_n(l_n(t_0))\|}{C_4} \leqslant \frac{\|r_n(l_n(t_n)) - r_0(l_0(t_0))\|}{C_4} + \\ + \frac{\|r_0(l_0(t_0)) - r_n(l_n(t_0))\|}{C_4} \leqslant \frac{\|r_n(l_n(t_n)) - r_0(l_0(t_0))\|}{C_4} + \frac{\|r_0 \circ l_0 - r_n \circ l_n\|_{C([0,T];\mathbb{R}^2)}}{C_4}. \tag{4.79}$$

Второй член в правой части (4.79) стремится к нулю в силу леммы 11.

Пусть пересеченный кусок границы $\Gamma \subseteq \partial\Omega$ задан как $r_\Gamma = (x_\Gamma(s), y_\Gamma(s))$, $r_\Gamma \in C^1[0,L_\Gamma]$ и этот кусок пересекается в точках $A_n = r_n(l_n(t_n)) = r_\Gamma(s_n)$ и $A_0 = r_0(l_0(t_0)) = r_\Gamma(s_0)$, s_n, $s_0 \in [0, L_\Gamma]$. Обозначим $B_0 = r_0(l_0(t_n))$ и $B_n = r_n(l_n(t_0))$ и рассмотрим свойства четырехугольника $B_nA_0B_0A_n$ (см. рис. 4.4).

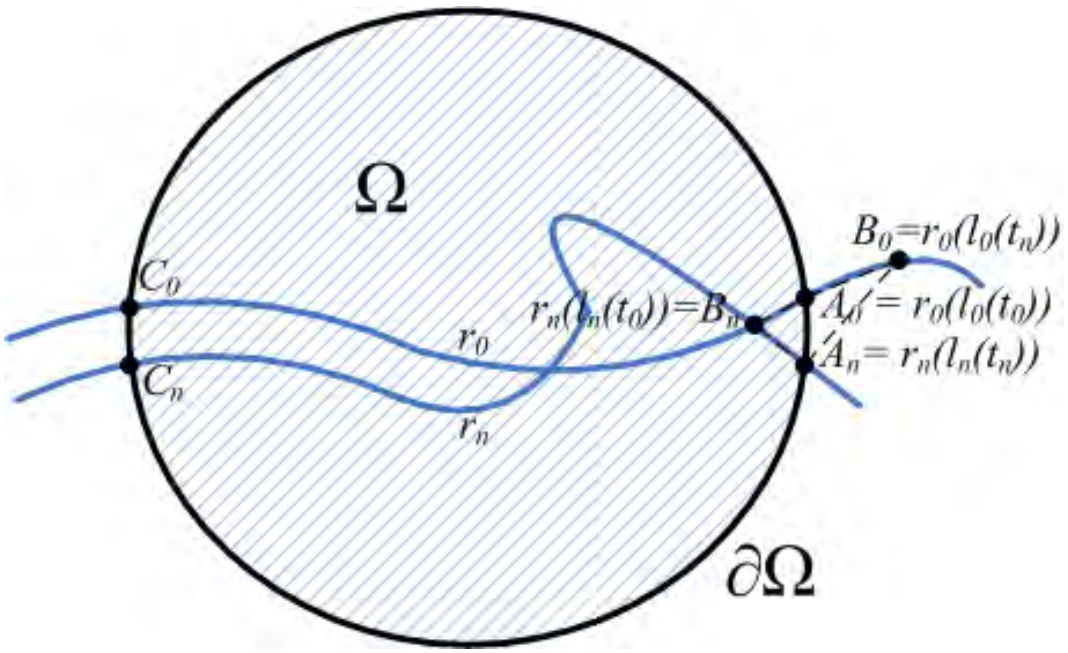


Рисунок 4.4 — Кривые r_n и r_0

Например, предположим, что $t_0 \leqslant t_n$ (необходимо просто изменить порядок соединения вершин в случае $t_0 > t_n$). Нам нужно получить оценку длины диагонали A_nA_0, т.е. для $\|r_n(l_n(t_n)) - r_0(l_0(t_0))\|$. В силу леммы 11, у нас есть

$$|B_nA_0| = \|r_n(l_n(t_0)) - r_0(l_0(t_0))\| \leqslant \|r_0 \circ l_0 - r_n \circ l_n\|_{C([0,T],\mathbb{R}^2)} \to 0, \tag{4.80}$$

$$|B_0A_n| = \|r_n(l_n(t_n)) - r_0(l_0(t_n))\| \leqslant \|r_0 \circ l_0 - r_n \circ l_n\|_{C([0,T],\mathbb{R}^2)} \to 0. \tag{4.81}$$

Положим $\alpha_n = \max\{l_n(t_0), l_0(t_0)\}$, $\beta_n = \min\{l_n(t_n), l_0(t_n)\}$. Справедливо, что

$$|A_0B_0| = \|r_0(l_0(t_0)) - r_0(l_0(t_n))\| \leqslant \int_{t_0}^{t_n} \|\partial_s r_0(l_0(s))\| ds = \int_{l_0(t_0)}^{l_0(t_n)} \|\partial_l r_0(l)\| dl, \tag{4.82}$$

$$|A_nB_n| = \|r_n(l_n(t_n)) - r_n(l_n(t_0))\| \leqslant \int_{t_0}^{t_n} \|\partial_s r_n(l_n(s))\| ds = \int_{l_n(t_0)}^{l_n(t_n)} \|\partial_l r_n(l)\| dl. \tag{4.83}$$

Предположим, что ни $|A_0B_0|$, ни $|A_nB_n|$ не стремятся к нулю (если они стремятся к нулю, то мы сразу получаем утверждение теоремы).

По неравенству треугольника $|\|x\| - \|y\|| \leqslant \|x - y\|$ и непрерывности вложения $W_p^1([0,L];\mathbb{R}^2) \subset W_1^1([0,L];\mathbb{R}^2)$, мы получаем

$$||A_nB_n| - |A_0B_0|| \leqslant \int_{\alpha_n}^{\beta_n} |\|\partial_l r_n(l)\| - \|\partial_l r_0(l)\||dl + \left|\int_{l_n(t_0)}^{l_0(t_0)} \|\partial_l r_n(l)\|dl\right| +$$
$$+ \left|\int_{l_n(t_n)}^{l_0(t_n)} \|\partial_l r_n(l)\|dl\right| + \left|\int_{l_n(t_0)}^{l_0(t_0)} \|\partial_l r_0(l)\|dl\right| + \left|\int_{l_n(t_n)}^{l_0(t_n)} \|\partial_l r_0(l)\|dl\right| \leqslant$$
$$\leqslant \int_0^L \|\partial_l(r_n(l) - r_0(l))\|dl + 4C_1\|l_n - l_0\|_{C[0,T]} \leqslant$$
$$\leqslant \|r_n - r_0\|_{W_1^1([0,L],\mathbb{R}^2)} + 4C_1\|l_n - l_0\|_{C[0,T]} \to 0. \quad (4.84)$$

По неравенству Птолемея для выпуклого четырехугольника, соотношениям (4.80), (4.81) и (4.84), мы получаем, что

$$|A_nA_0||B_nB_0| \leqslant |A_0B_0||B_nA_0| + |A_nB_n||B_0A_n| \to 0. \quad (4.85)$$

Остается оценить $|B_nB_0|$. По теореме косинусов, мы имеем

$$|B_nB_0|^2 = |B_0A_n|^2 + |A_nB_n|^2 - 2|B_0A_n||A_nB_n|\cos\angle B_nA_nB_0 \geqslant |B_0A_n|^2 + |A_nB_n|^2 -$$
$$- 2|B_0A_n||A_nB_n|. \quad (4.86)$$

В силу (4.31), (4.32), получаем

$$\|r_n(l_n(t_n)) - r_0(l_0(t_0))\| = |A_nA_0| \leqslant \frac{|A_0B_0||B_nA_0| + |A_nB_n||B_0A_n|}{|B_0A_n|^2 + |A_nB_n|^2 - 2|B_0A_n||A_nB_n|} \to 0. \quad (4.87)$$

Из (4.79), (4.86) мы находим что

$$|t_n - t_0| \to 0.$$

Рассмотрим случай

$$\sum_{i=1}^{3} \lambda_i (r_n \circ l_n)'_+(\xi_i) \to 0. \quad (4.88)$$

Соотношения (4.76), (4.88) влекут

$$\|r_n(l_n(t_n)) - r_n(l_n(t_0))\| \to 0.$$

Предположим, что $t_n \nrightarrow t_0$, то есть $\exists(\varepsilon_0 > 0)\forall(n \in \mathbb{N}) \quad |t_n - t_0| \geqslant \varepsilon_0$.

Параметр l_n строго позрастает по t_n по утверждению теоремы, так как $r_n(l_n(t_n))$ – точка на границе $\partial\Omega$, так что если $|t_n - t_0| \geqslant \varepsilon_0$, то

$$\exists(\varepsilon_1 > 0)\forall(n \in \mathbb{N}) \quad |l_n(t_n) - l_n(t_0)| > \varepsilon_1.$$

Кривые r_n сходятся к r_0 в $W_p^1([0,L];\mathbb{R}^2)$, поэтому длины этих кривых также сходятся: $l_n \to l_0$ в $C^1[0,T]$. Если расстояние $\|B_n - A_0\| \to 0$, то $|l_0(t_0) - l_n(t_n)| \to 0$. Это можно показать по аналогии с (4.84). Таким образом, мы получили противоречие

$$0 < \varepsilon_1 < |l_n(t_n) - l_n(t_0)| \leqslant |l_n(t_n) - l_0(t_0)| + |l_0(t_0) - l_n(t_0)| \leqslant \\ \leqslant |l_n(t_n) - l_0(t_0)| + \|l_0 - l_n\|_{C^1[0,T]} \to 0. \quad (4.89)$$

□

Теорема 5. *Задача (4.67), (4.68) имеет решение.*

Доказательство. В силу теоремы 4, компактности $\mathcal{Y}$ (лемма 5) и теоремы Вейерштрасса, функционал (4.67) имеет минимум на $\mathcal{Y}$. □

Также, мы имеем следующее следствие:

Лемма 12. *Пусть* $\min\limits_{(t,r)\in\Omega} \nu_\varepsilon(t,r) > 0$ *и выполнены условия теоремы 4. Тогда*

$$\left\| \sum_{i=1}^{3} \lambda_i (r_n \circ l_n)'_+(\xi_i) \right\| \geqslant (1-\eta) \min_{(t,r)\in\Omega} \nu_\varepsilon(t,r),$$

$\xi^*, \xi_i \in (\Theta(r_n),\Theta(r_0))$, $i = \overline{1,3}$, $0 \leqslant \eta < 1$.

Доказательство. Из (4.78) мы получаем

$$\left\| \sum_{i=1}^{3} \lambda_i (r_n \circ l_n)'_+(\xi_i) \right\| = \frac{\|r_n(l_n(t_n)) - r_n(l_n(t_0))\|}{|l_n(t_n) - l_n(t_0)|} |l'_n(\xi^*)| \geqslant \alpha \min_{(t,r)\in\Omega} \nu_\varepsilon(t,r),$$

и $0 < \alpha \leqslant 1$.

Так как $\Theta(r_n) = t_n \to t_0 = \Theta(r_0)$, $r_n \in W_p^1([0,L];\mathbb{R}^2)$, $l_n \in C^1[0,T]$, то

$$\frac{\|r_n(l_n(t_n)) - r_n(l_n(t_0))\|}{|l_n(t_n) - l_n(t_0)|} \to 1, \quad n \to \infty.$$

□

4.4.4 Оптимальное решение для всех агентов

Чтобы получить оптимальное решение, мы должны решить, какие агенты пойдут вперед и которые будут тормозить и уступать дорогу. В разделе 4.4.1 приоритет агентов для k-го агента определяется набором $\mathcal{I}^k$. Точное решение этой задачи можно описать следующим образом.

Обозначим через $\mathcal{II}$ множество всех возможных $\mathcal{I}$, удовлетворяющих условиям (4.62)–(4.63), определим $\mathbf{r} = (r_1, \ldots, r_m) \in (\mathcal{Y})^m$. Перепишите (4.61) как

$$\|\dot{r}_k(t)\| = w_{k,\varepsilon}(t, \mathbf{r}(t); \mathfrak{i}_k(t)), \quad k = \overline{1,m}, \tag{4.90}$$

где

$$w_{k,\varepsilon}(t, \mathbf{r}(t); \mathcal{I}^k) = u(t, r_k(t)) - v_{k,\varepsilon}(t, r_1(t), \ldots, r_m(t); \mathcal{I}^k),$$
$$\mathfrak{i}_k : t \mapsto \mathcal{I}^k \in \mathcal{I}_t \in \mathcal{II}, \quad k = \overline{1,m}, t \in [0,T].$$

Обозначим как $\mathfrak{I}_k$ множество всех отображений $\mathfrak{i}_k$, $\mathfrak{i} = (\mathfrak{i}_1, \ldots, \mathfrak{i}_m)$, и

$$\Theta_*(\mathbf{r}, \mathfrak{i}) = \sum_{k=1}^{m} \Theta(r_k, l_k(\mathfrak{i}_k)), \tag{4.91}$$

где $l_k(\mathfrak{i}_k)$ – решение (4.7) при $r = r_k$:

$$l(t) = \int_0^t w_{k,\varepsilon}(s, \mathbf{r}(l(s)); \mathfrak{i}_k(s)) ds, \; t \geqslant 0. \tag{4.92}$$

Мы должны минимизировать

$$\Theta_*(\mathbf{r}, \mathfrak{i}) \to \min, \quad \mathbf{r} \in (\mathcal{Y})^m, \mathfrak{i} \in \mathfrak{I}_1 \times \cdots \times \mathfrak{I}_m. \tag{4.93}$$

Эта задача известна как многоагентный поиск пути [156].

Чтобы упростить нашу задачу, мы можем кодировать множества $\mathcal{I}^k \subset \overline{1,m}$ как вектора $\mathbf{I}^k = (i_1^k, \ldots, i_m^k) \in \mathbb{Z}^n$ следующим образом

$$i_j^k = \begin{cases} 1, j \in \mathcal{I}^k, \\ 0, j \notin \mathcal{I}^k \end{cases}$$

и таким образом полагать, что $\mathfrak{i}_k : [0,T] \to \mathbb{Z}^m$.

Как следует из теоремы 5, задача

$$\Theta(r_k, l_k(\mathfrak{i}_k)) \to \min$$

имеет решение (возможно, не уникальное) для достаточно гладких

$$\nu_\varepsilon(t, r_k) = w_{k,\varepsilon}(t, \mathbf{r}; \mathfrak{i}_k(t)), \quad k = \overline{1,m},$$

например, где $\mathfrak{i}_k = const_k$. Таким образом, (4.91) также можно минимизировать с каждым фиксированным $\mathfrak{i} \in \mathfrak{I}_1 \times \cdots \times \mathfrak{I}_m$.

Кроме того, мы можем предположить, что $\mathfrak{i}_k$, $k = \overline{1,m}$ являются кусочно-постоянными отображениями с конечным числом частей:

$$\mathfrak{i}_k(t) = \mathfrak{i}_k^i, \quad t \in [t_{i-1}, t_i) \subset [0,T], t_i \in \mathbf{T}_{pc}. \tag{4.94}$$

Обозначим как $\mathbf{T}_1, \ldots, \mathbf{T}_q$, $\mathbf{T}_i \cap \mathbf{T}_j = \varnothing$, $i,j = \overline{1,q}$, интервалы времени, в которых существует риск столкновения:

$$\forall (t \in \mathbf{T}_i) \exists (k,j \in \overline{1,m}) \quad \|r_k \circ l_k(t; \mathfrak{i}_k(t)) - r_j \circ l_j(t; \mathfrak{i}_k(t))\| \leqslant \varepsilon + \alpha, \tag{4.95}$$

$l_k(t; \mathfrak{i}_k(t)) \; = \; l_k(\mathfrak{i}_k)(t)$,

$$\forall (t \in [0,T] \setminus \bigcup_{i=1}^{q} \mathbf{T}_i) \forall (k,j \in \overline{1,m}) \quad \|r_k \circ l_k(t; \mathfrak{i}_k(t)) - r_j \circ l_j(t; \mathfrak{i}_k(t))\| > \varepsilon + \alpha.$$

Ясно, что $w_{k,\varepsilon}(t, \mathbf{r}(t); \mathcal{I}^k) = u(t, r_k(t))$, $t \in [0,T] \setminus \bigcup\limits_{i=1}^{q} \mathbf{T}_i$, и если

$$\mathbf{T}_{pc} \cap \bigcup_{i=1}^{q} \mathbf{T}_i = \varnothing, \tag{4.96}$$

то $w_{k,\varepsilon} \in C([0,T], C^1(\Omega))$ и теорема 5 может быть применена, и имеет место следующая теорема:

Теорема 6. *Предположим, что выполнены условия (4.94)–(4.96). Тогда задача (4.93) имеет решение* $\mathbf{r} \in (\mathcal{Y})^m$.

Тем не менее, множество решений для разных $\mathfrak{i}$ может быть довольно большим, и зависимость $w_{k,\varepsilon}$ от $\mathfrak{i}$ может вообще иметь совершенно не непрерывный характер.

Из-за вычислительной и концептуальной сложности этой задачи, автор использовал метод аукциона, определяющий, какие агенты будут тормозить, для дискретного аналога проблемы (4.60), (4.90), (4.93) в статьях [6; 92]. Это означает, что в случае возможного столкновения агентов мы произвольно упорядочиваем набор сталкивающихся агентов (согласно некоторым правилам), и агенты с меньшими номерами идут вперед, а агенты с большим номером – тормозят. Следует отметить, что какой-то подобный алгоритм многоагентного поиска оптимального пути на основе аукциона описан в статье [157].

4.4.5 Заключение

В главе были математически обоснованы способы поиска агентом кратчайшего маршрута на местности с переменной проходимостью с помощью клеточного автомата, типа изложенных в [1; 2]. Также были получены оценки для аппроксимации маршрутов ломаными специального вида, в которых звенья могут образовывать угол друг с другом только в 0, 45, 90 и 180 градусов и соединяются только в узлах сетки из квадратов и построены дискретные аналоги функционалов времени выхода маршрута на границу области и близости к заданному маршруту, для данных функционалов также получены оценки. Показано, что последовательность оптимальных по времени для определенного шага сетки маршрутов, составленных из ломаных, при уменьшении шага сетки стремится к оптимальному маршруту.

Глава 5. Программная среда многоагентного моделирования «Психоход»

В настоящем разделе опишем разработанный автором программный продукт « Программная среда многоагентного моделирования «Психоход»» зарегистрированный Роспатентом в реестре программ для ЭВМ под номером 2017619605. Программа предназначена для имитационного моделирования движения по пересеченной местности, взаимодействия и конфликта агентов разных типов и основана на математической модели клеточного автомата. Поддерживается моделирование строя и роя агентов. Программа предоставляет возможность сбора и обработки статистических данных, задания различных сценариев поведения агентов с помощью скриптового языка Lua. Включена поддержка параллельных вычислений. Предусмотрено графическое интерактивное взаимодействие программы с пользователем. Тип реализующей ЭВМ: IBM PC – совмест. ПК, язык программирования: C++, вид и версия операционной системы: Windows 7, Ubuntu 16.04.

Для этой цели вкратце повторим ранее описанные результаты диссертации в том виде, в каком они реализованы в программе.

5.1 Модель движения

Пусть задано множество n мобильных агентов $Ag = \{ag_1, \dots, ag_n\}$. Все агенты передвигаются в едином дискретном времени $t = \overline{0,T_a}$ по области Ω. Также заданы ограниченное множество $\Omega \subset \mathbb{Z}^2$, множество векторов состояний клеток $\mathcal{S}$ и функция $\mathfrak{l} : \overline{0,T_a} \times \Omega \to \mathcal{S}$.

Мгновенное состояние клетки (i,j) $s_{ij}(t) = \mathfrak{l}(t; i,j)$ представляет собой

$$s_{ij} = (*ag_{ij}, u_{ij}, h_{ij}, \dots),$$

где $*ag_{ij}$ – это некий идентификатор (например, указатель на объект агента в памяти ЭВМ), характеризующий агента, находящегося в клетке (i,j), или 0, если агента в клетке нет, $u_{ij} \in \mathbb{N}$ – количество ходов, необходимое для агента, движущегося с максимально возможной в системе скоростью, для пересечения клетки (i,j) (непроходимость), $h_{ij} \in \mathbb{R}$ – это высота клетки (i,j).

Множества высот и непроходимостей клеток мы будем далее называть ландшафтом высот $\mathcal{L}_h = \{h_{ij}|(i,j) \in \Omega\}$ и ландшафтом непроходимостей $\mathcal{L}_u = \{u_{ij}|(i,j) \in \Omega\}$ соответственно.

Каждый агент ag_k обладает более или менее полной копией множества Ω и функции $\mathfrak{l}$. Набор $\mathcal{B}_0 = (\Omega, \mathfrak{l}, \dots)$ мы будем далее называть слоем объективной реальности агентов. Для описания примерного (т.е. реализованного автором статьи) слоя субъективной реальности ag_k определим функцию посещений $\mathfrak{vis}_k : \overline{0,T_a} \times \Omega \to \mathbb{Z}$ и радиус видимости агента r_k. В момент посещения агентом клетки (i,j) устанавливается, что $\mathfrak{vis}_k(t+1; i,j) = \mathfrak{vis}_k(t; i,j) + 1$ и для всех клеток (i',j') из окрестности $V_r(i,j) \cap \Omega$ клетки (i,j) радиуса r копируются в «субъективную реальность» агента $\mathfrak{l}_k(t; i',j) = \mathfrak{l}(t; i,j)$. Итого, слой субъективной реальности агента ag_k представляет собой набор $\mathcal{O}_k = (\Omega, \mathfrak{vis}_k, \mathfrak{l}_k, \dots)$. Слои $\mathcal{B}_i$, $i = \overline{1,n}$ представляют собой Believs в терминологии BDI агентов.

Многоточие «...» в описании означает, что размеры и состав описываемых структур не предопределены заранее и могут дополняться в процессе работы реализации модели в виде программного обеспечения.

При движении агенты стремятся минимизировать время прохождения от исходной точки до заданной цели. Таким образом, множество целей (Desires, в терминологии BDI) описываемых агентов включает в себя, как минимум, наборы конечных и начальных местоположений $\mathcal{G}_k = \{(i_b^k,j_b^k),(i_e^k,j_e^k)\}$.

Определим множество возможных направлений движения $\mathcal{D} = \{(i,j)|i,j = \overline{-1,1}\}$. Назовем маршрутом агента, находящегося в клетке (i_0,j_0) в клетку (i_m,j_m) в направлении d_0 множество

$$\mathfrak{m}(d_0; i_m,j_m) = \{(i_p,j_p)|(i_{k+1},j_{k+1}) = d_k + (i_k,j_k), (i_k,j_k) \in \Omega, d_k \in \mathcal{D}, k = \overline{0,m-1}\}.$$

Для достижения своей цели каждый агент, находящийся в некоторой точке (i_0,j_0) и движущийся к точке (i_e,j_e), ищет каждый ход оптимальный маршрут следующим образом. Пусть (i_m,j_m) – клетка пересечения границы окрестности $V_r(i_0,j_0)$ и прямой, проходящей через (i_0,j_0) в направлении (i_e,j_e), $\mathfrak{m}(d; i_m,j_m) \subseteq V_r(i_0,j_0)$. Тогда агент выбирает такое направление движения $d \in \mathcal{D}$, чтобы

$$\mathfrak{U}(d) = \sum_{(i,j) \in \mathfrak{m}(d; i_m,j_m)} u_{ij} \to \min, \tag{5.1}$$

$$\mathfrak{V}(d) = \sum_{(i,j) \in \mathfrak{m}(d; i_m,j_m)} \mathfrak{vis}_k(i,j) \to \min. \tag{5.2}$$

Если дополнительно вводится условие поддержания строя, то необходимо также минимизировать отклонение от заданного графа строя (при организации строя) или, альтернативно, чтоб количество других агентов в его окрестности соответствовало заданному (при организации роя). Сходство графов определяется с помощью меры несходства, описанной в работах [5; 11; 12]. Указанные правила поведения агента можно отнести к понятию Intentions модели BDI.

5.2 Модель системы связи

Обратим внимание на то, что объекты модели системы связи функционируют не в дискретном времени модели движени, а в моментах своего собственного дискретного времени $\tau = \overline{0,T_{na}}$, так как обмен сообщениями обычно происходит намного быстрее физического перемещения агентов.

У каждого $ag \in Ag$ может быть одно или несколько «средств связи» $netAg \in NetAg$, характеризующееся предельной дальностью (в клетках) $r_{\max}(netAg)$, предельной высотой препятствия между приемником и передатчиком $h_{\max}(netAg)$ и максимальным быстродействием (абстракцией скорости) $\sigma_{\max}(netAg)$. Список средств связи агента ag_k будем обозначать как $NetAg_k$.

Будем обозначать средство, используемое агентом в момент времени τ как $netAg(\tau, ag)$. Средства $netAg_1$ и $netAg_2$ могут быть совместимыми с друг другом, т.е. пригодными для организации двухстороннего канала связи, в этом случае будем писать $\Gamma\,(netAg_1,netAg_2) = 1$ и несовместимыми, в таком случае будем писать $\Gamma\,(netAg_1,netAg_2) = 0$.

Высота препятствия между двумя клетками $b = (i_b,j_b)$, $e = (i_e,j_e)$ «ландшафта» $\mathcal{L}_h$ определяется как

$$\mathfrak{h}(b,e) = \max_{(i,j)\in L(b,e)} (h_{ij} - h(b,e;i,j)), \tag{5.3}$$

где $L(b,e)$ – множество квадратов, пересекающихся с прямой, соединяющей центры клеток b и e, $h(b,e;i,j)$ – прямая, проходящая через точки $(i_b,j_b,h_{i_bj_b})$ и $(i_s,j_s,h_{i_sj_s})$.

На рис. 5.1 показано взаимодействие 4 агентов ag_i, $i = \overline{1,4}$. Для средств $netAg_j$, $j = 1,4,5,8$ предельная высота препятствия $h_{\max}(netAg_j) = 0$, для $j =$

2,3,6,7 предельная высота препятствия $h_{\max}(netAg_j) = 2$. Поэтому, например, ag_1 и ag_3 не могут связываться напрямую.

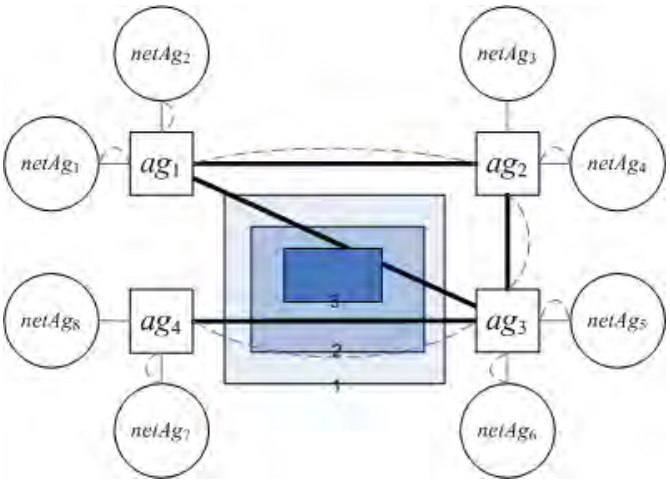


Рисунок 5.1 — Взаимодействие агентов и средств связи при наличии препятствия

Определим теперь клеточный автомат, моделирующий поведение радиоэфира. Для всех агентов задано множество каналов в виде матрицы Q. Номер столбца этой матрицы соответствует номеру канала, номер строки нужен для учета размещения нескольких агентов на одном канале, а также для моделирования, например, номера временного окна агента на частотном канале. Компонента q_{ij} матрицы Q соответствует качеству канала j. Каждое средство может находиться на одном из каналов, канал, на который настроено средство $netAg$, обозначим как $j(netAg)$. Таким образом, состояние ячейки КА – модели эфира определяется как

$$c_{ij} = (*netAg_{ij}, q_{ij}, \dots),$$

где $*netAg_{ij}$ – идентификатор средства $netAg_{ij}$, находящегося в клетке (i,j) или 0, если средства в клетке (i,j) нет.

Также задается реальное быстродействие связи у средства $netAg$ на канале j как функция

$$\mathfrak{s}(netAgent, j) = \sigma(\sigma_{\max}(netAgent), q_{ij}(netAgent), N_{\text{ag}}(j)),$$

где $N_{\text{ag}}(j)$ – количество агентов на канале j. Функция σ возрастает по первому и второму аргументу и убывает по третьему. Технически, $\mathfrak{s}$ может реализовываться как свойство класса средства связи или же как элемент состояния ячейки КА – модели эфира.

Каждый агент ag_k обладает расписанием сообщений

$$\mathcal{R}_k = \{(\tau_i; ag_i; v_i) | \tau_i = \overline{0,T_{na}}, ag_i \in Ag, v_i \in \mathbb{N}\}.$$

На основании расписания для каждого агента $ag_k \in Ag$ можно построить графом потребностей в связи Π_k с вершинами из Ag. Ребра в данном графе будут соответствовать необходимости поддерживать связь между соответствующими им парами агентов. Для упрощения дальнейшего изложения, будем писать $X \Rightarrow y$, если во множестве $X = \{(y_1,x_1), \ldots, (y_n,x_n)\}$ существует элемент (y_k,x_k) с $y = y_k$, при этом обозначать $x_k = X[y_k] = X[y]$. Данное обозначение в целом соответствует, например, обозначениям в духе контейнеров «ключ-значение» типа `std::map` из STL языка C++.

У каждого агента ag_k есть список загруженности его средств связи $\mathcal{AIL}_k = \{(netAg, v) | netAg \in NetAg_k, v \in \mathbb{Z}\}$.

Правила действий агента (т.н. агентная функция) в модели системы связи описываются следующим образом:

Шаг 0. В начальный момент времени $\mathcal{AIL} = \{(netAg, 0) | netAg \in NetAg_k\}$.

Шаг 1. Агент ag_k каждый такт τ просматривает расписание $\mathcal{R}_k$. Если $\mathcal{R}_k \Rightarrow \tau$ и найдено $\mathcal{R}_k[\tau] = (ag_i, v_i)$, то переходим к шагу 2, иначе – к шагу 4.

Шаг 2. Агент ag_k просматривает свои средства связи $\mathcal{AIL}$ в поиске средства, отвечающего следующим критериям:

- расстояние между ag_k и ag_i не превосходит $r_{\max}(netAg)$;

- максимальная высота препятствия между ag_k и ag_i не превосходит $h_{\max}(netAg)$.

Шаг 3. После нахождения такого средства $\mathcal{AIL} \Rightarrow netAg$ агент ag_k устанавливает $\mathcal{AIL}[netAg] = \mathcal{AIL}[netAg] + v_i$ и сообщает средству $netAg$ о необходимости передать сообщение.

Шаг 4. Средства из $NetAg_k$ опрашиваются. Если $netAg$ ответило об успехе передачи, то если $\mathcal{AIL}[netAg] > 0$, устанавливается $\mathcal{AIL}[netAg] = \mathcal{IL}[netAg] - \mathfrak{s}(netAg, j(netAg))$.

Шаг 5. Переход к шагу 1.

Средства $netAg$ также являются, в общем случае (при моделировании Cognitive Radio), агентами, которые перемещаются по сетке Q в поисках лучшего канала связи и обмениваются сообщениями, которые передали им агенты из Ag. Каждый $netAg_k \in NetAg$ обладает расписанием $\mathcal{NIL}_k = \{(\tau_i; netAg_i; v_i) | i = \overline{1,s}\}$. Список заданий заполняется на шаге 3 предыдущего алгоритма из расписания

$\mathcal{R}_k$, конкретный способ формирования зависит от типа моделируемой системы связи.

Эти агенты руководствуются правилами вида

Шаг 1. $netAg_p$ проверяет $\mathcal{NIL}_k$, нет ли у него заданий от его владельца ag_k.

Шаг 2. Если есть $(\tau_l; netAg_l; v_l)$ с $v_l = 0$, то оно исключается из $\mathcal{NIL}_k$.

Шаг 3. Если есть задание $(\tau_l; netAg_l; v_l)$, и нет других заданий, и $j(netAg_p) = j(netAg_l)$, и $\Gamma\,(netAg_p,netAg_l) = 1$, то объем v_i уменьшается на $\beta = \mathfrak{s}(netAgent_p, j(netAg_p))$.

Шаг 4. Если заданий несколько, то они обрабатываются в соответствии с заданной дисциплиной обслуживания.

Шаг 5. Переход к шагу 1.

Дисциплина обслуживания может быть, например, FIFO или же величина β из шага 2 может делиться равномерно между заявками. Дополнительно возможно введение приоритетов сообщений.

Видно, что на базе семейства расписаний $\{\mathcal{R}_k | k = \overline{1,n}\}$ можно построить матрицу потоков для графа Γ_0 с вершинами из Ag и (или) $NetAg$. В данном графе вершины ag_i, ag_j будут смежными, если

- существуют $netAg_p \in NetAg_k$, $netAg_l \in NetAg_k$, $j(netAg_p) = j(netAg_l)$ и $\Gamma\,(netAg_p,netAg_l) = 1$;

- расстояние между ag_k и ag_i не превосходит $r_{\max}(netAg)$;

- максимальная высота препятствия между ag_k и ag_i не превосходит $h_{\max}(netAg)$.

По сути, задача агентов – организовать, с учетом вышеуказанных ограничений, такую сеть Γ_0, чтоб поток сообщений через каждое ребро был бы минимален.

Таким образом, пр движении агенты стремятся к тому, чтоб максимизировать реальное быстродействие канала и к тому, чтоб граф связи системы соответствовал бы графу потребностей в связи. Это достигается, например, за счет того, что агенты избегают препятствий, загораживающих от них других агентов и т.д. Возможно задавать и другие критерии оптимальности, соответствующие моделируемой системе.

В рамках идеологии BDI агентов, к Believs у средства связи $netAg_k$ относятся сведения о расположении средств на каналах связи $\{j(netAg) | netAg \in$

$NetAg_k \subset NetAg\}$ и о принадлежности средств связи агентам модели движения, к Desires – $\mathcal{NIL}_k$ и условие максимально быстрой отправке сообщений, к Intentions – дисциплина обслуживания, алгоритмы маршрутизации и т.п.

Группа агентов, двигающихся по пересеченной местности, может рассматриваться как одномерный клеточный автомат (т.н. robot-space cellular automaton, далее – КА пространства робота), каждая клетка которого содержит координаты агента на плоскости, его реальное и желаемое взаимное расположение с соседними агентами или же как двухмерный клеточный автомат (т.н. world-space cellular automaton, далее – КА пространства мира), каждая клетка которого соответствует квадрату местности и набору его свойств и содержит идентификатор находящегося в данной клетки агента [80].

Обозначим как $\mathfrak{M}_0$ глобальную функцию перехода КА пространства мира модели движения, $\mathfrak{M}_1$ – глобальную функцию перехода КА пространства робота модели движения, $\mathfrak{N}_0$ глобальную функцию перехода КА пространства мира модели системы связи, $\mathfrak{N}_1$ – глобальную функцию перехода КА пространства робота модели системы связи.

Можно рассматривать модели движения и системы связи как конечные автоматы, представимые в виде уравнений

$$\mathbf{s}(t+1) = \mathfrak{M}_0(t, \mathbf{s}(t), \mathbf{c}(t)),$$
$$\mathbf{ag}(t+1) = \mathfrak{M}_1(t, \mathbf{ag}(t), \mathbf{netAg}(t), \mathbf{s}(t), \mathbf{c}(t)),$$
$$\mathbf{c}(\tau+1) = \mathfrak{N}_0(\tau, \mathbf{s}(\tau), \mathbf{c}(\tau)),$$
$$\mathbf{netAg}(\tau+1) = \mathfrak{N}_1(\tau, \mathbf{ag}(\tau), \mathbf{netAg}(\tau), \mathbf{s}(\tau), \mathbf{c}(\tau)),$$
$$\tau = Kt,$$

где $\mathbf{s}(t)$ – вектор состояний клеток КА модели движения, $\mathbf{c}(\tau)$ – вектор состояний клеток КА модели системы связи, $\mathbf{ag}(t)$ – вектор состояний агентов, $\mathbf{netAg}(\tau)$ – вектор состояний средств связи, $K \in \mathbb{N}$.

5.3 Соединение моделей движения и связи в среде «Психоход»

Программная среда «Психоход» реализована на языке C++14 с применением библиотек Qt 5.9.1. В программе (см. рис. 5.2) присутствует интерфейсный

класс `IAutomaton`, который представляет собой абстрактный клеточный автомат и унаследованные от него классы автоматов разного типа, например, `Bokohod` – класс модели движения или `AutoNet` – класс модели связи, `Life` – демонстрационный класс «Жизни» Конвея. Также в модели есть классы агентов: `Agent` – реализация множества Ag и `NetAgent` – реализация множества $netAg$.

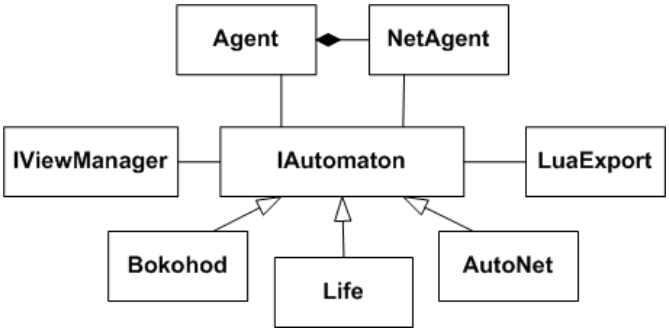


Рисунок 5.2 — Иерархия классов системы моделирования «Психоход»

Помимо этого, конкретные эксперименты, связанные с движением и обменом сообщениями между агентами, производятся либо с графического интерфейса, либо из скрипта Lua 5.2, обрабатываемого с помощью класса `LuaExport`. Визуализация результатов эксперимента осуществляется в классе `ViewManager`.

Технически, слои $\mathcal{B}_i$ реализованы в виде контейнеров STL типа `std::vector<StateLine>`, где `StateLine` – это `std::vector<Cell>`, а `Cell`, в свою очередь, это структура произвольной длины, хранящая состояние клетки из $\mathcal{S}$.

Поскольку изменения в модели системы связи и в модели движения происходят с существенно разными скоростями, то имеет смысл их реализовывать либо в разных потоках одного приложения, либо вообще в разных процессах, и автор выбрал последнее, для облегчения последующего подключения своего программного обеспечения к существующим сетевым симуляторам, таким как omnet++ или ns-3. В программе «Психоход» модели системы связи и движения представляют собой два независимых процесса операционной системы (см. рис. 5.3), взаимодействие которых происходит через разделяемую память `QSharedMemory` с синхронизацией тактов через стандартные семафоры `lock()` и `unlock()`.

Обработка состояний $\mathbf{ag}(t)$ и актуализация $\mathbf{ag}(t+1)$ происходит в двух разных циклах для обеспечения параллельных вычислений с помощью OpenMP. При этом дополнительная логика поведения агентов может быть

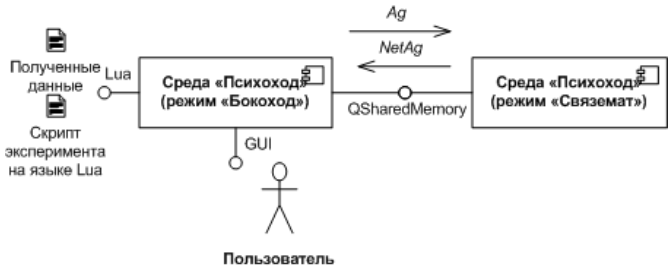


Рисунок 5.3 — Взаимодействие модели связи и движения в среде «Психоход»

добавлена как непосредственно внутрь обработчика состояний агентов (функции `Automaton::next()`), так и загружена извне в вектор объектов типа `void*(*)(void*)` (которые впоследствии приводятся к нужному типу), который последовательно исполняется на каждом шаге функционирования КА.

Заключение

Основные результаты работы заключаются в следующем. В диссертации были получены следующие результаты.

1. Разработана формализованная модель управления подвижными агентами (роботами, БПЛА, солдатами), обменивающихся информационными сообщениями и командами по мобильной сети связи и стремящихся двигаться оптимальным образом в соответствии с заданными критериями. Управление в данной модели заключается либо в начальном задании целей и параметров агента, таких как радиус обзора, точка назначения, склонность к организации групп с другими агентами, скорость агента и т.п., либо в относительно нечастом изменении этих параметров в процессе функционирования системы агентов. Агенты сконструированы так, что их внутренняя система принятия решения позволяет им работать продолжительное время и достигать поставленных им целей без внешнего вмешательства. Предлагаемая в диссертации система связи между агентами может также функционировать без внешнего вмешательства и адаптироваться к изменениям помеховой обстановки, исчезновению отдельных агентов из сети и т. д.

2. Предложены дискретные модели движения, конфликта и связи агентов на основе клеточного автомата. Была также предложена непрерывная модель движения агентов, соответствующая дискретному аналогу. Оказалось, что исследуемый клеточный автомат соответствует конечно-разностной схеме для приближенного решения непрерывной модели движения агентов.

3. Разработано специальное программное обеспечение «Психоход», реализующее вышеуказанную модель и обеспечивающее визуализацию и анализ соответствующей информации. С помощью указанного программного обеспечения разработаны методы выявления зависимости эффективности поведения агентов (скорости движения, качества организации строя, скорости организации сети связи) от количества агентов и свойств местности, имеющихся каналов связи.

4. Разработанная автором многоагентная система может применяться для моделирования поведения различных сложных систем, в том числе движения

насекомых, микроорганизмов, системы кровообращения и т. п. Например, агентам могут соответствовать эритроциты, перемещающиеся по загроможденному разными препятствиями руслу кровеносной системы.

5. Исследованы зависимости скорости самоорганизации децентрализованной сети связи на основе *когнитивного радио* от количества агентов, подсетей и каналов связи и выявлено, что такая зависимость описывается известной из социологии моделью сегрегации Шеллинга тип II. Были разработаны алгоритмы, необходимые для автономного движения робота в соответствии с поставленными целями и для организации сети связи на основе *когнитивного радио*.

6. На основе проведенных вычислительных экспериментов были получены зависимости применяемых в ландшафтной экологии характеристик случайных *ландшафтов*, таких как «общий край» и конфигурационная энтропия. Вычислительные эксперименты показали зависимости времени движения агентов по пересеченной местности от свойств *ландшафта*,

- таких как «общий край» и конфигурационная энтропия;
- зависимость параметров интегральных характеристик движущейся системы агентов (например, концентрации агентов) от свойств *ландшафта*;
- характер распределения координат агента при движении по кратчайшему по времени маршруту и характер распределения времени прихода в целевую точку.

Перспектива исследования.

Дальнейшим развитием исследований может быть применение полученных результатов для создания проблемно-ориентированных систем управления, принятия решений и оптимизации.

Список литературы

1. *Кузнецов А.В.* Упрощенная модель боевых действий на основе клеточного автомата // *Известия РАН. Теория и системы управления*. — 2017. — Т. 56, № 3. — С. 59–71. — URL: https://elibrary.ru/item.asp?id=29369822.

2. *Кузнецов А.В.* Модель совместного движения агентов с трехуровневой иерархией на основе клеточного автомата // *Журнал вычислительной математики и математической физики*. — 2017. — Т. 57, № 2. — С. 339–349. — URL: https://elibrary.ru/item.asp?id=28918677.

3. *Kuznetsov A., Shishkina E., Sitnik S.* Probabilistic Properties of Near-Optimal Trajectories of an Agent Moving Over a Lattice // *Journal of Optimization Theory and Applications*. — 2018. — Aug. — URL: https://doi.org/10.1007/s10957-018-1374-6.

4. *Kuznetsov A. V.* On the Motion of Agents across Terrain with Obstacles // *Computational Mathematics and Mathematical Physics*. — 2018. — Jan. — Vol. 58, no. 1. — Pp. 137–151. — URL: https://doi.org/10.1134/S0965542518010098.

5. *Schumann A., Kuznetsov A. V.* Foundations of mathematics under neuroscience conditions of lateral inhibition and lateral activation // *International Journal of Parallel, Emergent and Distributed Systems*. — 2018. — Vol. 33, no. 3. — Pp. 237–256. — URL: https://doi.org/10.1080/17445760.2018.1439490.

6. *Kuznetsov A. V.* Model of the motion of agents with memory based on the cellular automaton // *International Journal of Parallel, Emergent and Distributed Systems*. — 2018. — Vol. 33, no. 3. — Pp. 290–306. — URL: https://doi.org/10.1080/17445760.2017.1410819.

7. *Kuznetsov A. V.* Segregation model for dynamic frequency allocation // *Advances in Systems Science and Applications*. — 2018. — Vol. 18, no. 2. — Pp. 84–92. — URL: http://ijassa.ipu.ru/index.php/ijassa/article/view/542.

8. *Kuznetsov A.* Generation of a Random Landscape by given Configuration Entropy and Total Edge // *Computational Technologies*. — 2017. — Vol. 22, no. 4. — Pp. 4–10. — URL: https://elibrary.ru/item.asp?id=30053459.

9. *Кузнецов А.В.* Моделирование системы связи агентов движущихся по пересечённой местности // *Челябинский физико-математический журнал*. — 2018. — Т. 3, № 2. — С. 237–248. — URL: http://cpmj.csu.ru/index.php/cpmj/article/view/178/127.

10. *Кузнецов А. В.* Динамическая модель системы связи группы агентов // *Управление большими системами: сборник трудов*. — 2018. — № 75. — С. 6–29.

11. *Кузнецов А.В.* Мера несходства на множестве графов и ее приложения // *Вестник ВГУ. Серия: Системный анализ и информационные технологии*. — 2017. — Т. 1. — С. 125–131. — URL: https://elibrary.ru/item.asp?id=29185115.

12. *Кузнецов А. В.* Организация строя агентов с помощью клеточного автомата // *Управление большими системами: сборник трудов*. — 2017. — № 70. — С. 136–167.

13. *Кузнецов А.В., Бессонов В.В.* Описание математической модели системы связи // *Теория и техника радиосвязи*. — 2010. — № 2. — С. 58–64.

14. *Кузнецов А.В., Бессонов В.В., Кручинин С.В. и др.* Объектно-лингвистическая модель единой среды визуального проектирования и формат хранения и переноса данных о системе связи на основе XML // *Вестник Воронежского Государственного Технического Университета*. — 2011. — Т. 7, № 8. — С. 56–60.

15. *Кузнецов А.В., Жарков С.Н.* Настройка беспроводной сети специального назначения по защищенному радиоканалу // *Электросвязь*. — 2016. — № 12. — С. 28–35.

16. *Кузнецов А. В.* Распределение ограниченных ресурсов в системе с устойчивой иерархией (на примере перспективной системы военной связи) // *Управление большими системами*. — 2017. — Т. 66. — С. 68–93.

17. *Кузнецов А. В.* Краткий обзор многоагентных моделей // *Управление большими системами*. — 2018. — Т. 71. — С. 6–44.

18. Квазиоптимальное распределение авионики на основе клеточного автомата и муравьиного алгоритма / А.В. Кузнецов, Н.И. Сельвесюк, М.Е. Семенов,

Г.А. Платошин // *Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии*. — 2017. — № 4. — С. 38–45. — URL: https://elibrary.ru/item.asp?id=32497378.

19. *Кузнецов А.В*. Модель движения и взаимодействия системы интеллектуальных агентов // *Вестник ВГУ, Серия: системный анализ и информационные технологии*. — 2018. — № 2. — С. 130–138. — URL: http://www.vestnik.vsu.ru/pdf/analiz/2018/02/2018-02-16.pdf.

20. Application of cellular automatons and ant algorithms in avionics / A. V. Kuznetsov, N. I. Selvesiuk, G. A. Platoshin, E. V. Semenova // *Journal of Physics: Conference Series*. — 2018. — Vol. 973, no. 1. — P. 012062. — URL: http://stacks.iop.org/1742-6596/973/i=1/a=012062.

21. *Kuznetsov A.V.* Cellular automata-based model of group motion of agents with memory and related continuous model // Proceedings of the Mathematical Modeling Session at the International Conference Information Technology and Nanotechnology (MM-ITNT 2017) / Ed. by Sergei Sazhin, Elena Shchepakina, Vladimir Sobolev, Denis Kudryashov. — CEUR Workshop Proceedings no. 1904. — Aachen: 2017. — Pp. 223–231. — URL: http://ceur-ws.org/Vol-1904/paper38.pdf.

22. *Schumann A., Kuznetsov A.V.* Talmudic foundations of mathematics // BICT 2017 – 10th EAI International Conference on Bio-Inspired Information and Communications Technologies 10. — 2017. — Pp. 7–74.

23. *Kuznetsov A.* Automatic role-oriented assignment of channels in the ad hoc network of hierarchically organized agents // В сборнике: BICT 2017 – 10th EAI International Conference on Bio-Inspired Information and Communications Technologies 10. — 2017. — Pp. 75–82.

24. *Кузнецов А.В., Бессонов В.В.* Технология создания самоорганизующейся радиосети с функциями когнитивного радио на основе принципов программно-определяемого радио // Кибернетика и высокие технологии XXI века. XV Международная научно-техническая конференция. Воронеж: НПФ «Саквоее» ООО. — № 2. — 2014. — С. 176–187.

25. *Кузнецов А.В.* Построение непрерывной модели направленного движения агентов по пересеченной местности // Математические методы в технике и технологиях – ММТТ. — Т. 2. — Саратов: 2017. — С. 46–51.

26. *Кузнецов А.В.* Построение непрерывной модели направленного движения агентов по пересеченной местности // Актуальные проблемы прикладной математики, информатики и механики сборник трудов Международной научно-технической конференции. — Воронеж: 2017. — С. 747–749.

27. *Кузнецов А.В.* Децентрализованное управление строем агентов, основанное на клеточном автомате // В сборнике: Современные сложные системы управления. Материалы XII международной научно-практической конференции. — 2017. — С. 91–95.

28. *Кузнецов А.В.* Анализ совместного движения системы агентов // В сборнике: Воронежская зимняя математическая школа С. Г. Крейна - 2016 материалы международной конференции. Под редакцией В. А. Костина. — 2016. — С. 242–245.

29. *Кузнецов А.В.* Адаптивная иерархическая система связи, управления и поддержки принятия решений // В сборнике: РАДИОЛОКАЦИЯ, НАВИГАЦИЯ, СВЯЗЬ XXII Международная научно-техническая конференция. — 2016. — С. 269–277.

30. *Кузнецов А.В.* Моделирование системы совместно движущихся агентов // Актуальные проблемы прикладной математики, информатики и механики Сборник трудов Международной научно-технической конференции, посвященный 70-летию Победы в Великой отечественной войне. — 2015. — С. 287–289.

31. *Кузнецов А.В.* Модель системы связи мобильных агентов, соединенная с моделью движения, и ее реализация в среде «Психоход» // В сборнике: XXIV Международная научно-техническая конференция «Радиолокация, навигация, связь» (RLNC*2018). — Т. 5. — Воронеж: ООО «ВЭЛБОРН», 2018. — С. 351–360.

32. *Кузнецов А.В.* Вероятностные свойства квазиоптимальных траекторий агента, движущегося по решетке // СБОРНИК ТРУДОВ ИТНТ-2018. IV

международная конференция и молодежная школа «Информационные технологии и нанотехнологии» 24 – 27 апреля. — Самара: Новая техника, 2018. — С. 2001–2009.

33. *Обельченко М.В., Бессонов В.В., Кузнецов А.В., Мохряков В.Б.* Средство передачи данных телекоммуникационной сети и телекоммуникационная сеть // Патент России № 2549120, опубл. 20.04.2015.

34. *Кузнецов А.В., Нестеренко А.В., Богатырев М.В., Пустовит В.П.* Способ организации защищенной системы связи // Патент России № 2644523, опубл. 2018/2/12.

35. *Кузнецов А.В.* Программа моделирования движения и боевых действий иерархически организованных агентов «Бокоход»; пр. для ЭВМ 2016615934. Заяв. и правообладатель Акционерное общество «Концерн «Созвездие». 2016613058 06.04.2016; опубл. 20.07.2016 // *Программы для ЭВМ. Базы данных. Топологии интегральных микросхем*. — 2016. — № 7. — 1 с.

36. *Кузнецов А.В., Лещев А.С.* Программная среда многоагентного моделирования «Психоход». Пр. для ЭВМ № 2017619605. Дата регистрации: 28.08.2017, номер и дата поступления заявки: 2017616880 11.07.2017. Правообл. А.В. Кузнецов, А.С. Лещев // *Программы для ЭВМ. Базы данных. Топологии интегральных микросхем*. — 2017. — № 9.

37. *Iglesias Carlos A., Garijo Mercedes, González José C.* A Survey of Agent-Oriented Methodologies // Intelligent Agents V: Agents Theories, Architectures, and Languages: 5th International Workshop, ATAL'98 Paris, France, July 4–7, 1998 Proceedings / Ed. by J. P. Müller, A. S. Rao, M. P. Singh. — Berlin, Heidelberg: Springer Berlin Heidelberg, 1999. — Pp. 317–330. — URL: http://dx.doi.org/10.1007/3-540-49057-4_21.

38. *Тарасов В.Б.* Нетрадиционные и гибридные логики в моделировании интеллектуальных агентов. I. Искусственные деятели, интенциональные характеристики и пути моделирования // *Известия РАН. Теория и системы управления*. — 2000. — no. 5. — Pp. 5–17.

39. *Stone P., Veloso M.* Multiagent Systems: A survey from a machine learning perspective // *Autonomous Robots*. — 2000. — July. — Vol. 8, no. 3. — Pp. 345–383.

40. *Borshchev A., Filippov A.* From System Dynamics and Discrete Event to Practical Agent Based Modeling: Reasons, Techniques, Tools // The 22nd International Conference of the System Dynamics Society. July 25 - 29, 2004, Oxford, England / Ed. by editor. — 2004.

41. A Survey of Programming Languages and Platforms for Multi-Agent Systems. / R. H. Bordini, L. Braubach, M. Dastani et al. // *Informatica (Slovenia)*. — 2006. — Vol. 30, no. 1. — Pp. 33–44. — URL: http://dblp.uni-trier.de/db/journals/informaticaSI/informaticaSI30.html.

42. A Survey on Parallel and Distributed Multi-Agent Systems / A. Rousset, B. Herrmann, C. Lang, L. Philippe // Euro-Par 2014: Parallel Processing Workshops - Euro-Par 2014 International Workshops, Porto, Portugal, August 25-26, 2014, Revised Selected Papers, Part I. — 2014. — Pp. 371–382. — URL: https://doi.org/10.1007/978-3-319-14325-5_32.

43. *Kravari K., Bassiliades N.* A Survey of Agent Platforms // *Journal of Artificial Societies and Social Simulation*. — 2015. — Vol. 18, no. 1. — URL: http://EconPapers.repec.org/RePEc:jas:jasssj:2014-71-2.

44. *Проскурников А.В., Фрадков А.Л.* Задачи и методы сетевого управления // *Автоматика и телемеханика*. — 2016. — № 10. — С. 3–39.

45. *Antonelli G.* Interconnected dynamic systems: An overview on distributed control // *IEEE Control Systems*. — 2013. — Feb. — Vol. 33, no. 1. — Pp. 76–88.

46. *Wieland P.* From Static to Dynamic Couplings in Consensus and Synchronization among Identical and Non-Identical Systems: Ph.D. thesis / Universität Stuttgart. Institut für Systemtheorie und Regelungstechnik. — 2010.

47. *Bullo F.* Lectures on Network Systems. — Version 0.95, 2017. — With contributions by J. Cortes, F. Dorfler, and S. Martinez. URL: http://motion.me.ucsb.edu/book-lns.

48. *Агаев Р.П., Чеботарев П.Ю.* Модели латентного консенсуса // *Автоматика и телемеханика*. — 2017. — № 1. — С. 106–120.

49. *Квинто Я.И.* Линейные алгоритмы управления геометрическим расположением объектов в многоагентной системе // *Управление большими системами*. — 2010. — Pp. 665–680.

50. *Квинто Я.И., Парсегов С.Э.* Равноудаленное расположение агентов на отрезке: анализ алгоритма и его обобщения // *Автоматика и телемеханика*. — 2012. — no. 11. — Pp. 30–41.

51. Real-time task scheduling for multi-agent control system of UAV's group based on network-centric technology / D. Budaev, K. Amelin, G. Voschuk et al. // 2016 International Conference on Control, Decision and Information Technologies (CoDIT). — 2016. — April. — Pp. 378–381.

52. *Sörensen K., Sevaux M., Glover F.* A history of metaheuristics // OR2016: Annual International Conference of the German Operations Research Society. — Hamburg, Germany: 2016. — URL: http://leeds-faculty.colorado.edu/glover/468%20-%20A%20History%20of%20Metaheuristics%20w%20Sorensen%20&%20Sevaux.pdf.

53. Journal of Heuristic Policies on Heuristic Search Research. — URL: http://www.springer.com/cda/content/document/cda_downloaddocument/Journal+of+Heuristic+Policies+on+Heuristic+Search.pdf?SGWID=0-0-45-1483502-p35487524.

54. Метод подвижных клеточных автоматов как новое направление дискретной вычислительной механики. I. Теоретическое описание / С.Г. Псахье, Г.П. Остермайер, А.И. Дмитриев и др. // *Физическая мезомеханика*. — 2000. — Т. 3, № 2. — С. 5–13. — URL: http://www.ispms.ru/ru/journals/290/459/.

55. *Wolfram S.* Cellular automaton fluids 1: Basic theory // *Journal of Statistical Physics*. — 1986. — Nov. — Vol. 45, no. 3. — Pp. 471–526. — URL: http://dx.doi.org/10.1007/BF01021083.

56. *Succi S.* The Lattice Boltzmann Equation: For Fluid Dynamics and Beyond. Numerical Mathematics and Scientific Computation. — Clarendon Press, 2001. — P. 288. — URL: https://books.google.ru/books?id=OC0Sj_xgnhAC.

57. *Alarcón T., Byrne H.M., Maini P.K.* A cellular automaton model for tumour growth in inhomogeneous environment // *Journal of Theoretical Biology*. — 2003. — Vol. 225, no. 2. — Pp. 257 – 274. — URL: http://www.sciencedirect.com/science/article/pii/S0022519303002443.

58. *Monteagudo Á., Santos J.* A Cellular Automaton Model for Tumor Growth Simulation // 6th International Conference on Practical Applications of Computational Biology & Bioinformatics / Ed. by Miguel P. Rocha, Nicholas Luscombe, Florentino Fdez-Riverola, Juan M. Corchado Rodríguez. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. — Pp. 147–155. — URL: http://dx.doi.org/10.1007/978-3-642-28839-5_17.

59. *Narváez C. A., Tovar A., Garzón D. A.* Topology Synthesis of Compliant Mechanisms Using the Hybrid Cellular Automaton Method with an Efficient Mass Control Strategy // III European Conference on Computational Mechanics: Solids, Structures and Coupled Problems in Engineering: Book of Abstracts / Ed. by C. A. Motasoares, J. A. C. Martins, H. C. Rodrigues et al. — Dordrecht: Springer Netherlands, 2006. — Pp. 490–490. — URL: http://dx.doi.org/10.1007/1-4020-5370-3_490.

60. *Gordon Th. J.* A simple agent model of an epidemic // *Technological Forecasting and Social Change*. — 2003. — Vol. 70, no. 5. — Pp. 397 – 417. — URL: http://www.sciencedirect.com/science/article/pii/S0040162502003232.

61. *Frias-Martinez E., Williamson G., Frias-Martinez V.* An Agent-Based Model of Epidemic Spread Using Human Mobility and Social Network Information // 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing. — 2011. — Oct. — Pp. 57–64.

62. Modeling transport through an environment crowded by a mixture of obstacles of different shapes and sizes / A. J. Ellery, R. E. Baker, S. W. McCue, M. J. Simpson // *Physica A: Statistical Mechanics and its Applications*. — 2016. — Vol. 449, no. Supplement C. — Pp. 74 – 84. — URL: http://www.sciencedirect.com/science/article/pii/S0378437115011607.

63. *Jones J., Mayne R., Adamatzky A.* Representation of shape mediated by environmental stimuli in Physarum polycephalum and a multi-agent model // *IJPEDS*. — 2017. — Vol. 32, no. 2. — Pp. 166–184. — URL: https://doi.org/10.1080/17445760.2015.1044005.

64. *Jones J.* Applications of multi-agent slime mould computing // *IJPEDS*. — 2016. — Vol. 31, no. 5. — Pp. 420–449. — URL: https://doi.org/10.1080/17445760.2015.1085535.

65. Cellular automaton model of crowd evacuation inspired by slime mould / V. S. Kalogeiton, D. P. Papadopoulos, I. Georgilas et al. // *Int. J. General Systems*. — 2015. — Vol. 44, no. 3. — Pp. 354–391. — URL: https://doi.org/10.1080/03081079.2014.997527.

66. Bio-Inspired Game Theory: The Case of Physarum Polycephalum / A. Schumann, K. Pancerz, A. Adamatzky, M. Grube // 8th International Conference on Bio-inspired Information and Communications Technologies, BICT 2014, Boston, MA, USA, December 1-3, 2014. — 2014. — URL: https://doi.org/10.4108/icst.bict.2014.257869.

67. *Schumann A.* Go games on plasmodia of physarum polycephalum // 2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015, Lódz, Poland, September 13-16, 2015. — 2015. — Pp. 615–626. — URL: https://doi.org/10.15439/2015F236.

68. *Schumann A.* Towards Context-Based Concurrent Formal Theories // *Parallel Processing Letters*. — 2015. — Vol. 25, no. 1. — URL: https://doi.org/10.1142/S0129626415400083.

69. *Schumann A.* Payoff Cellular Automata and Reflexive Games // *J. Cellular Automata*. — 2014. — Vol. 9, no. 4. — Pp. 287–313. — URL: http://www.oldcitypublishing.com/journals/jca-home/jca-issue-contents/jca-volume-9-number-4-2014/jca-9-4-p-287-313/.

70. *Schumann A., Fris V.* Swarm Intelligence among Humans - The Case of Alcoholics // Proceedings of the 10th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2017) - Volume 4: BIOSIGNALS, Porto, Portugal, February 21-23, 2017. — 2017. — Pp. 17–25. — URL: https://doi.org/10.5220/0006106300170025.

71. Введение в математическое моделирование транспортных потоков / А.В. Гасников, С. Л. Кленов, Е. А. Нурминский и др. — Москва: МЦНМО, 2013.

72. *Kesting A., Treiber M., Helbing D.* Agents for Traffic Simulation. — URL: https://arxiv.org/pdf/0805.0300.pdf.

73. *Reynolds C. W.* Flocks, Herds and Schools: A Distributed Behavioral Model // *SIGGRAPH Comput. Graph.* — 1987. — aug. — Vol. 21, no. 4. — Pp. 25–34. — URL: http://doi.acm.org/10.1145/37402.37406.

74. *Jones M.P., Dudenhoeffer D.D.* A formation behavior for large-scale micro-robot force deployment // Winter Simulation Conference. — Vol. 01. — Los Alamitos, CA, USA: IEEE Computer Society, 2000. — Pp. 972–982. — URL: doi.ieeecomputersociety.org/10.1109/WSC.2000.899900.

75. *Малинецкий Г. Г., Степанцов М. Е.* Применение клеточных автоматов для моделирования движения группы людей // *Журнал вычислительной математики и математической физики*. — 2004. — Т. 44, № 11. — С. 2094–2098.

76. *Бреер В.В., Новиков Д.А., Рогаткин А.Д.* Управление толпой: математические модели порогового коллективного поведения. — Москва: Ленанд, 2016.

77. *Beer B., Mead R., Weinberg J. B.* A Distributed Method for Evaluating Properties of a Robot Formation // Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010. — 2010. — URL: http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/view/1677.

78. *Long R. L., Mead R., Weinberg J. B.* Distributed Auction-Based Initialization of Mobile Robot Formations // Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010. — 2010. — URL: http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/view/1673.

79. *Mead R., Weinberg J. B.* 2-Dimensional Cellular Automata Approach for Robot Grid Formations // Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008. — 2008. — Pp. 1818–1819. — URL: http://www.aaai.org/Library/AAAI/2008/aaai08-299.php.

80. *Mead R., Weinberg J. B.* A Single- and Multi-Dimensional Cellular Automata Approach to Robot Formation Control // Proceedings of IEEE International Conference on Robotics and Automation (ICRA-08). — 2008. — URL: http://robotics.usc.edu/~rossmead/docs/2008/2008WeinbergMead_ICRA08.pdf.

81. Graph Matching-Based Formation Reconfiguration of Networked Agents With Connectivity Maintenance / Z. Kan, L. Navaravong, J. M. Shea et al. // *IEEE Transactions on Control of Network Systems*. — 2015. — March. — Vol. 2, no. 1. — Pp. 24–35.

82. *Морозова Н.С.* Децентрализованное управление движением строя роботов при динамически изменяющихся условиях // *Искусственный интелект и принятие решений*. — 2015. — Т. 1. — С. 65 – 74.

83. A cooperative formation control strategy maintaining connectivity of a multi-agent system / R. Dutta, L. Sun, M. Kothari et al. // 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, September 14-18, 2014. — 2014. — Pp. 1189–1194. — URL: https://doi.org/10.1109/IROS.2014.6942708.

84. *Oh Kwang-Kyo, Park Myoung-Chul, Ahn Hyo-Sung*. A survey of multi-agent formation control // *Automatica*. — 2015. — Vol. 53. — Pp. 424–440. — URL: http://www.sciencedirect.com/science/article/pii/S0005109814004038.

85. *Tsatsoulis Costas, Soh Leen-Kiat*. Intelligent Agents in Telecommunication Networks // in Computational Intelligence in Telecommunications Networks. — CRC Press, 2000. — Pp. 23–28.

86. *Merghem L., Lecarpentier H.* Agents: A Solution for Telecommunication Network Simulation // Network Control and Engineering for QoS, Security and Mobility: IFIP TC6 / WG6.2 & WG6.7 Conference on Network Control and Engineering for QoS, Security and Mobility (Net-Con 2002) October 23–25, 2002, Paris, France / Ed. by Dominique Gaïti, Nadia Boukhatem. — Boston, MA: Springer US, 2003. — Pp. 165–176. — URL: http://dx.doi.org/10.1007/978-0-387-35620-4_14.

87. Mobile Agents for Integration of Internet of Things and Wireless Sensor Networks / T. Leppänen, M. Liu, E. Harjula et al. // 2013 IEEE International Conference on Systems, Man, and Cybernetics. — 2013. — Oct. — Pp. 14–21.

88. *Batool K., Niazi M. A.* Modeling the internet of things: a hybrid modeling approach using complex networks and agent-based models // *Complex Adaptive Systems Modeling*. — 2017. — Mar. — Vol. 5, no. 1. — P. 4. — URL: http://dx.doi.org/10.1186/s40294-017-0043-1.

89. *Ilachinski A.* Artificial War: Multiagent-Based Simulation of Combat. — Singapore: World Scientific Publishing Company, 2004. — P. 747.

90. Recent Developments in theMANA Agent-based Model / G. C. McIntosh, D. P. Galligan, M. A. Anderson, M. K. Lauren // *Scythe : Proceedings and Bulletin of the International Data Farming Community*. — 2003. — no. 1. — Pp. 38 – 39. — URL: https://harvest.nps.edu/scythe/Issue1/IDFW13-Scythe-Mana.pdf.

91. *Ross J. L.* A Comparative Study of Simulation Software for Modeling Stability Operations // Proceedings of the 2012 Symposium on Military Modeling and Simulation. — 2012. — URL: https://ccl.northwestern.edu/papers/2012/ross.pdf.

92. *Kuznetsov A. V.* A simplified combat model based on a cellular automaton // *Journal of Computer and Systems Sciences International*. — 2017. — May. — Vol. 56, no. 3. — Pp. 397–409. — URL: http://dx.doi.org/10.1134/S106423071703011X.

93. *Hamill L., Gilbert N.* Agent-Based Modelling in Economics. — 1st edition. — Wiley Publishing, 2016. — P. 256.

94. *Pyka A., Fagiolo G.* Agent-based Modelling: A Methodology for Neo-Schumpetarian Economics // Elgar Companion to Neo-Schumpeterian Economics. — Edward Elgar Publishing, 2007. — URL: http://EconPapers.repec.org/RePEc:elg:eechap:2973_29.

95. *Abbas S. M. A.* An agent-based model of the development of friendship links within Facebook // *Computational and Mathematical Organization Theory*. — 2013. — Jun. — Vol. 19, no. 2. — Pp. 232–252. — URL: http://dx.doi.org/10.1007/s10588-013-9156-z.

96. *Proskurnikov A. V., Tempo R.* A tutorial on modeling and analysis of dynamic social networks. Part I // *Annual Reviews in Control*. — 2017. — Vol. 43, no. Supplement C. — Pp. 65 – 79. — URL: http://www.sciencedirect.com/science/article/pii/S1367578817300172.

97. Network science on belief system dynamics under logic constraints / N. E. Friedkin, A. V. Proskurnikov, R. Tempo, S. E. Parsegov // *Science*. — 2016. — Vol. 354, no. 6310. — Pp. 321–326. — URL: http://science.sciencemag.org/content/354/6310/321.

98. *Ferber J.* Les Systèmes multi-agents: vers une intelligence collective. I.I.A. Informatique intelligence artificielle. — InterEditions, 1995. — URL: https://books.google.ru/books?id=R7VBAAAACAAJ.

99. *Russell S. J., Norvig P.* Artificial Intelligence: A Modern Approach. — 2 edition. — Upper Saddle River, New Jersey: Prentice Hall, 2010. — P. 1132.

100. *Rao A. S., Georgeff M. P.* BDI Agents: From Theory to Practice // Proceedings of the First International Conference on Multiagent Systems. — 1995. — Pp. 312–319.

101. *Кудрявцев В.Б., Подколзин А.С., Болотов А.А.* Основы теории однородных структур. — Москва: Наука, 1990. — С. 296.

102. *Cushman S. A.* Calculating the configurational entropy of a landscape mosaic // *Landscape Ecology*. — 2016. — Vol. 31, no. 3. — Pp. 481–489. — URL: http://dx.doi.org/10.1007/s10980-015-0305-2.

103. FRAGSTATS: Spatial Pattern Analysis Program for Categorical Maps. Documentation. — URL: http://www.umass.edu/landeco/research/fragstats/documents/fragstats_documents.html.

104. *Rodráguez-Iturbe I., D'Odorico P., Rinaldo A.* Configuration entropy of fractal landscapes // *Geophysical Research Letters*. — 1998. — Vol. 25, no. 7. — Pp. 1015–1018. — URL: http://dx.doi.org/10.1029/98GL00654.

105. *Johnson L., Yannakakis G. N., Togelius J.* Cellular Automata for Real-time Generation of Infinite Cave Levels // Proceedings of the 2010 Workshop on Procedural Content Generation in Games. — PCGames '10. — New York, NY, USA: ACM, 2010. — Pp. 10:1–10:4. — URL: http://doi.acm.org/10.1145/1814256.1814266.

106. A Survey of Procedural Noise Functions / A. Lagae, S. Lefebvre, R. Cook et al. // *Computer Graphics Forum*. — 2010.

107. *Agrawal N., Bikineev A., Bristow P. A. et al.* Math Toolkit 2.5.0. — 2014. — URL: http://www.boost.org/doc/libs/1_62_0/libs/math/doc/html/index.html.

108. *Efraimidis P., Spirakis P.* Weighted Random Sampling // Encyclopedia of Algorithms / Ed. by Ming-Yang Kao. — Boston, MA: Springer US, 2008. — Pp. 1–99. — URL: https://doi.org/10.1007/978-0-387-30162-4_478.

109. *Geuvers H.* Inductive and coinductive types with iteration and recursion // Proceedings ofthe 1992 workshop on Types for Proofs and Programs. — Bastad: Chalmers University of Technology, 1992. — Pp. 183–207. — URL: http://www.cs.ru.nl/~herman/PUBS/BRABasInf_RecTyp.pdf.

110. Deriving movement properties and the effect of the environment from the Brownian bridge movement model in monkeys and birds / K. Buchin, S. Sijben, E. E. van Loon et al. // *Movement Ecology*. — 2015. — Jun. — Vol. 3, no. 1. — P. 18. — URL: https://doi.org/10.1186/s40462-015-0043-8.

111. *Тихонов А.Н., Самарский А.А.* Уравнения математической физики. — Москва: Издательство МГУ, 1999.

112. *Annighöfer B.* Model-based architecting and optimization of distributed integrated modular avionics. — Aahen: Shaker Verlag, 2015. — 312 pp.

113. *Sidiropoulos E., Fotakis D.* Spatial optimization and resource allocation in a cellular automata framework // Cellular Automata-Simplicity Behind Complexity. — InTech, 2011.

114. *Dorigo M., Caro G. D., Gambardella L. M.* Ant algorithms for discrete optimization // *Artificial life*. — 1999. — Vol. 5, no. 2. — Pp. 137–172.

115. *Corson S., Macker J.* Request for Comments: 2501. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations // IETF Tools. — 1999. — January. — URL: https://tools.ietf.org/html/rfc2501 (дата обращения: 21.08.2015).

116. *Васильев Андрей*. Концепция автоматизированной системы управления боевыми действиями Сухопутных войск АСУВ 2.0 // Армейский Вестник. — 2013. — Октябрь. — URL: https://goo.gl/wcu3DY (дата обращения: 11.08.2015).

117. *Тимофеев А.В., Юсупов Р.М.* Принципы построения интегрированных систем мультиагентной навигации и интеллектуального управления мехатронными роботами // *International Journal «Information Technologies & Knowledge»*. — 2011. — Т. 5, № 3. — С. 237 – 244. — URL: http://www.foibg.com/ijitk/ijitk-vol05/ijitk05-3-p04.pdf.

118. *Аничкин А.С., Семенов В.А.* Современные модели и методы теории расписаний // *Труды ИСП РАН*. — 2014. — Т. 26, № 3. — С. 5–50. — URL: http://www.ispras.ru/proceedings/docs/2014/26/3/isp_26_2014_3_5.pdf.

119. *Кандауров Дмитрий*. АСУВ в Алабино: желаемое и действительное // Независимое военное обозрение. — 2010. — Ноябрь. — URL: http://nvo.ng.ru/armament/2010-11-19/1_asuv.html (дата обращения: 21.08.2015).

120. *Лазарев А.А., Мусатова Е.Г., Гафаров Е.Р. и др.* Теория расписаний. Задачи железнодорожного планирования. Научное издание. — М.: ИПУ РАН, 2012.

121. *Прилуцкий М.Х., Афраймович Л. Г.* Многоиндексные задачи распределения ресурсов в иерархических системах // *Автоматика и телемеханика*. — 2006. — № 6. — С. 194–205.

122. *Прилуцкий М.Х., Дикарев К.И.* Распределение ресурсов в иерархических системах с активными элементами // *Вестник ННГУ*. — 2012. — № 5-2. — С. 181–189.

123. *Виттих В.А., Скобелев П.О.* Мультиагентные модели взаимодействия для построения сетей потребностей и возможностей в открытых системах // *Автоматика и телемеханика*. — 2003. — С. 177–185.

124. *Быховский М.А., Васильев А.В., Лашкевич А.В.* Основы управления использованием радиочастотного спектра: Частотное планирование сетей телерадиовещания и подвижной связи. Автоматизация управления использованием радиочастотного спектра. — Москва: Красанд, 2012. — Т. 3.

125. *Гениатулин К.А., Носов В.И.* Применение алгоритмов вершина-краска и краска-вершина для модификации метода координационных колец при частотно-пространственном планированиисистемы спутниковой связи с зональным обслуживанием // *Вестник СибГУТИ*. — 2014. — № 3. — С. 23–36.

126. *Носов В.И.* Методы обеспечения электромагнитной совместимости в сетях радиосвязи // *Вестник СибГУТИ*. — 2007. — № 1. — С. 52–56.

127. *Ошмарин Д.В.* Распределение канальных ресурсов в сетях когнитивного радио на основе теории игр // *Бизнес-информатика*. — 2010. — № 4(14). — С. 38–45.

128. *Антонюк Л.Я., Семисошенко М.А.* Адаптивная радиосвязь в системах связи специального назначения // *Электросвязь*. — 2007. — no. 5. — Pp. 261–266.

129. *Беда С.И., Катанович А.А., Пашкевич Л.А., Коваленко Ю.Г.* Патент 2405265 РФ; заявка 2008131085/09. Способ адаптации каналов радиосвязи с использованием искусственного интеллекта и устройство для его осуществления. – Дата начала отсчета срока действия патента 28.07.2008; дата публикации заявки 10.02.2010.

130. *Каплин Е.А., Лебединский Е.В., Егоров В.В.* Современныесистемы передачи данных по КВ радиоканалам // *Электросвязь*. — 2003. — no. 7. — Pp. 261–266.

131. *Redi J., Ramanathan R.* The DARPA WNaN network architecture // 2011 - MILCOM 2011 Military Communications Conference. — 2011. — Nov. — Pp. 2258–2263. — URL: https://www.researchgate.net/publication/241634120_The_DARPA_WNaN_network_architecture.

132. *Sumaiya B. D., Nithya R., Prasanth K.* Energy Efficient Hierarchical Cluster based Routing Protocols in WSN – A Survey // *International Journal for Innovative Research in Science & Technology*. — 2014. — Vol. 1, no. 7. — Pp. 261–266. — URL: http://www.ijirst.org/articles/IJIRSTV1I7098.pdf.

133. *Fadok D.S., Boyd J., Warden J.* Air Power's Quest for Strategic Paralysis. — Maxwell Air Force Base, Alabama: Air University Press, 1995.

134. *Kondo Y., Yomo H., Tang S. et al*. Wake-up radio using IEEE802.11 frame length modulation for radio-on-demand wireless LAN // IEEEInternational Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC. — 2011. — Pp. 869–873.

135. *Kent S., Seo K.* RFC 4301: Security Architecture for InternetProtocol (IPSec) // IETF. — 2005. — URL: https://tools.ietf.org/html/rfc4301.

136. *Freier A.* RFC 6101: The Secure Sockets Layer (SSL) Protocol Version 3.0 // IETF. — 2015. — URL: https://www.ietf.org/rfc/rfc5246.txt.

137. *Dierks T.and Rescorla E.* The transport layer security (TLS) protocol – Version 1.2 // IETF. — 2008. — URL: https://www.ietf.org/rfc/rfc5246.txt.

138. *Pelillo M.* Replicator Equations, Maximal Cliques, and Graph Isomorphism // *Neural Computation*. — 1999. — Vol. 11, no. 8. — Pp. 1933–1955. — URL: https://doi.org/10.1162/089976699300016034.

139. *Jain A. K., Murty M. N., Flynn P. J.* Data Clustering: A Review // *ACM Comput. Surv.* — 1999. — sep. — Vol. 31, no. 3. — Pp. 264–323. — URL: http://doi.acm.org/10.1145/331499.331504.

140. *Horne W. D.* Adaptive Spectrum Access: Using the Full Spectrum Space // Proc. of the International Symposium on Advanced Radio Technologies. — 2003.

141. *Zhao Qing, Tong Lang, Swami A.* Decentralized cognitive mac for dynamic spectrum access // First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. — 2005. — Nov. — Pp. 224–232.

142. *Schelling Th. C.* Dynamic models of segregation // *The Journal of Mathematical Sociology*. — 1971. — Vol. 1, no. 2. — Pp. 143–186. — URL: https://doi.org/10.1080/0022250X.1971.9989794.

143. *Haw D.* Measuring and Understanding Segregation: Ph.D. thesis / Bristol Centre for Complexity Sciences. — University of Bristol, 2016. — URL: https://www.imperial.ac.uk/people/d.haw/research.html.

144. *Haw D. J., Hogan J.* A dynamical systems model of unorganized segregation // *The Journal of Mathematical Sociology*. — 2018. — Vol. 0, no. 0. — Pp. 1–15. — URL: https://doi.org/10.1080/0022250X.2018.1427091.

145. *Шнайер Б.* Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. — М.: Триумф, 2002.

146. *Ögren P.* Formations and Obstacle Avoidance in Mobile Robot Control: Ph.D. thesis / Otimization and Systems Theory department of Mathematics Royal Institute of Technology. — Stockholm, 2003. — 36 pp. URL: https://www.diva-portal.org/smash/get/diva2:9373/FULLTEXT01.pdf.

147. Метод интеллектуальной обработки данных оценки местности в интересах определения районов для передвижения и размещения специальной техники различного назначения / А.А. Ерыгин, С.А. Житенев, В.Н. Коротун и др. // *Теория и техн. радиосвязи*. — 2014. — № 1. — С. 67–74. — URL: http://www.sozvezdie.su/uploads/files/files/ttr/ttr_2014_1.pdf.

148. *Кондрашов В.И.* О некоторых свойствах функций пространства L_p // *Докл. АН СССР*. — 1945. — Т. 48, № 6. — С. 563 – 566.

149. *Колмогоров А.Н., Фомин С.В.* Элементы теории функций и функционального анализа. — М.: Наука, 1976. — С. 101 – 103.

150. *McLeod R. M.* Mean value theorems for vector valued functions // *Proc. Edinburgh Math. Soc.* — 1965. — Vol. 14, no. 2. — Pp. 197 – 209. — URL: https://www.cambridge.org/core/services/aop-cambridge-core/content/view/S0013091500008786.

151. *Филиппов А. Ф.* Дифференциальные уравнения с разрывной правой частью // *Матем. сборник*. — 1960. — Т. 51(93), № 1. — С. 99 – 128.

152. *Никольский С.М.* Приближение функций многих переменных и теоремы вложения. — М.: Наука, 1969. — 480 с.

153. *Карпов В.Э.* Об одной реализации знак-ориентированной системы управления мобильного робота // *Искусственный интеллект и принятие решений*. — 2015. — № 3. — С. 53 – 61.

154. *Бекларян А.Л., Акопов А.С.* Моделирование поведения толпы на основе интеллектуальной динамики взаимодействующих агентов // *Бизнес-информатика*. — 2015. — № 1 (31). — С. 69 – 77. — URL: https://bijournal.hse.ru/2015--1%20(31)/150564297.html.

155. *Матыаш И.* Случайная оптимизация // *Автоматика и телемеханика*. — 1965. — Т. 5, № 2. — С. 246 – 253. — URL: http://mi.mathnet.ru/rus/at/v26/i2/p246.

156. *Ma Hang, Kumar T. K. Satish, Koenig S.* Multi-Agent Path Finding with Delay Probabilities // Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17), San Francisco, California, USA. — 2017. — Pp. 3605–3612. — URL: https://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14984.

157. *Amir O., Sharon G., Stern R.* Multi-agent Pathfinding As a Combinatorial Auction // Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. — AAAI'15. — AAAI Press, 2015. — Pp. 2003–2009. — URL: http://dl.acm.org/citation.cfm?id=2886521.2886599.

Список рисунков

0.1 Схема многоагентной системы 17
0.2 Слои взаимодействия агентов 19

1.1 Куб сложности для связанных динамических систем 38
1.2 Тактовое функционирование клеточного автомата для LBM вида D_2Q_9 41
1.3 Начальное распределение элементов клеточного автомата. Черные клетки пустые, белые заняты сосудами, светло-серые – раковыми клетками и темно-серые – нормальными клетками 42
1.4 *Physarum polycephalum* на стволе дерева (фотография frankenstoen – flickr, CC BY 2.5, https://commons.wikimedia.org/w/index.php?curid=18930526) 45
1.5 World-Space Cellular Automaton и его ограничения: (а) робот между клетками сетки; (б) граница, окружающая автомат; (в) автомат оборачивается вокруг границы; (г) два робота пытаются занять одну и ту же клетку 49
1.6 Роботы как ячейки в одномерном клеточном автомате пространства роботов (Robot-Space Cellular Automaton) 50
1.7 Траектории агентов, занимающих желаемое положение 51
1.8 Пример структуры агента телекоммуникационной сети 53

2.1 Зависимости S, N_1 и β 64
2.2 Вид ландшафтов $\mathcal{L}(48 \times 48{,}10)$, сгенерированных по алгоритму раздела 2.2.2. Слева направо $S = 0.349119$, $TED = 0.40842$, $S = 1.1004$, $TED = 1.05773$, $S = 2.00013$, $TED = 1.63411$ 64
2.3 Зависимость $\overline{TED}$ при разных значениях ε и S, $n = m = 48$, $l = 152$. 66
2.4 Ландшафты при $S = 1.90026$, $\overline{TED} = 0.728733$, $\varepsilon = 0.05$, $S = 2.19722$, $\overline{TED} = 1.03125$, $\varepsilon = 0.2$, $S = 1.90026$, $\overline{TED} = 1.57552$, $\varepsilon = 0.9$ 67
2.5 Примерные зависимости конфигурационной энтропии, TED и N_{obst}. . 67
2.6 Примеры ландшафтов, чем темнее клетка, тем труднее она проходится 68
2.7 Зависимость $\overline{TED}$ от S/S_{max} и количества классов клеток l ($n = m = 48$) 69
2.8 Две конфликтующие группы агентов 74

2.9 Боковые (показаны сплошными стрелочками) и основные (показаны пунктиром) входы и выходы автоматов. 77
2.10 Пример клеточного автомата для агентов с памятью 88
2.11 Обход небольшого одиночного препятствия строем из шести агентов. 103
2.12 Зависимость среднего отклонения позиции агентов от предусмотренной строем и количества препятствий 104
2.13 Проигрыш времени (круглые маркеры) и отклонение от траектории (квадратные) в процентах по сравнению с оптимальным маршрутом для равномерного случайного ландшафта 109
2.14 Проигрыш времени (круглые маркеры) и отклонение от траектории (квадратные) в процентах по сравнению с оптимальным маршрутом для «естественного» случайного ландшафта 109
2.15 Относительный выигрыш времени 110
2.16 Гистограммы распределения времени прибытия в конечную точку маршрута τ для равномерного (а) и «естественного» (б) ландшафтов . 117
2.17 Функции плотности, полученные эмпирически и аналитически 118
2.18 Распределение координаты i агента для $t = 12, 52, 132, 200$ 119
2.19 Зависимость p-значения для различных времен 120
2.20 Вид гистограмм вблизи конца маршрута, $t = 304, 404, 504$ 120
2.21 Поведение функции $\varphi(t,x)$, $x = 94{,}96{,}98$ 121
2.22 Зависимость среднего количества агентов u от дискретного времени t, $N_{obst} = 20$, $x = 2$, $x = 12$, $x = 24$, $x = 36$, $x = 48$ слева направо . . . 124
2.23 Зависимость u от энтропии S, слева направо $N_{obst} = 20$, $N_{obst} = 40$, $N_{obst} = 60$, $N_{obst} = 80$, $N_{obst} = 100$, $N_{obst} = 120$, $N_{obst} = 140$ 128
2.24 Функции u для ландшафтов, сгенерированных разными способами, $x = 26$, $S = 1.84877$ 129
2.25 Пример назначения кабельных трасс и мест установки 130
2.26 Разбиение U на выпуклые подмножества 131

3.1 Пример функции видимости. 141
3.2 Поведение модельной системы связи. Штриховкой обозначена степень проницаемости клетки для радиоволн, чем гуще штриховка, тем хуже проницаемость 148
3.3 Граф потребностей Req 158
3.4 Граф связи Γ 159

3.5 Структура Req' графа потребностей из рис. 3.3 171
3.6 Схема иерархической самоорганизующейся сети 172
3.7 Обмен маячками 174
3.8 Модель сегрегации Шеллинга 178
3.9 Финальная конфигурация автомата 180
3.10 Программа «Психоход» в процессе сегрегации агентов 181
3.11 Среднее время завершения сегрегации сети (50 агентов в сети, 5 сетей) 182
3.12 Гистограммы времен сегрегации 182
3.13 Среднее время завершения сегрегации сети (25 агентов в сети, 5 сетей) 183
3.14 Графики результатов численного эксперимента по сравнению с решениями динамической системы Шеллинга 185
3.15 Изменение матрицы прямой видимости агентов со временем для различных ландшафтов 187
3.16 Формат маячка настройки и маячка ответа 189
3.17 Формат промежуточного маячка настроечных данных (а) и маячка завершения (б) 191
3.18 Последовательность настройки радиосети специального назначения . 192
3.19 Модель системы связи 194
3.20 Зависимость продолжительности 2-го этапа настройки t_1^k от скорости передачи данных v при использовании 50 частотных каналов (а) и зависимость продолжительности 2-го этапа настройки t_1^k от количества частотных каналов F при скорости передачи данных $v = 16$ кбит/с (б) 196
3.21 Автоматическая настройка радиостанции в сети специального назначения 197
3.22 Зависимость времени автоматической настройки T от вероятности подавления используемой частоты P_i (а) и зависимость вероятности автоматической настройки P_r от вероятности подавления используемой частоты P_i (б) 197

4.1 Движение агента около предварительно заданного маршрута ρ. Линии уровня функции u обозначены как u_1, u_2, u_3, u_4. 199
4.2 Маршрут и его приближение 210
4.3 Пример маршрута r и $l(t) = x(l(t))$ 223
4.4 Кривые r_n и r_0 228

5.1 Взаимодействие агентов и средств связи при наличии препятствия . . 237
5.2 Иерархия классов системы моделирования «Психоход» 241
5.3 Взаимодействие модели связи и движения в среде «Психоход» 242

Список таблиц

1 Значения констант для зависимости энтропии и TED 66

2 Значения функции com 148
3 Значения функции q 149
4 Значения функции φ_1 150
5 Значения функции **I** для агентов из рис. 3.3 170
6 Конструкция изоморфизма $\mathcal{H} : Ag'(2) \to Ag''(2)$ 171
7 Процент неудачных сегрегаций 183