Министерство науки и высшего образования РФ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт математики и фундаментальной информатики
Кафедра высшей и прикладной математики

УТВЕРЖДАЮ
Заведующий кафедрой
/С.Г. Мысливец
«___» _____ 2022 г.

БАКАЛАВРСКАЯ РАБОТА

Направление 01.03.02 Прикладная математика и информатика

ЗАДАЧА MARL ДЛЯ ОПТИМИЗАЦИИ ПЛАНОВ КООРДИНАЦИЙ СВЕТОФОРНЫХ ОБЪЕКТОВ УЧАСТКА ДОРОЖНОЙ СЕТИ

Руководитель — доцент, кандидат физико-математических наук — Д.В. Семенова

Выпускник — Т.И. Тисленко

Нормоконтролер — Т.Н. Шипина

Красноярск 2022

РЕФЕРАТ

Бакалаврская работа по теме «Задача MARL для оптимизации планов координаций светофорных объектов участка дорожной сети» содержит 53 страницы текста, 1 приложение, 16 использованных источников.

ОБРАТНАЯ ЗАДАЧА, MDP, MARL, QLEARNING, СИСТЕМА АДАПТИВНОГО УПРАВЛЕНИЯ СВЕТОФОРАМИ.

Цель работы — разработать и исследовать математические модели мультиагентной системы для задачи сововокупного управления светофорными объектами дорожной сети.

Основные результаты дипломной работы представлены ниже.

1. Построены математические модели процесса управления для одного и для сети светофорных объектов, отличающиеся учетом текущего расположения светофорных объектов и их загруженности и позволяющие сформулировать оптимизационные задачи, целью которых является минимизация задержки трафика автомобилей.
2. Разработан и теоретически обоснован алгоритм управления участком дорожной сети для одного двухфазного светофорного объекта.
3. Разработан и теоретически обоснован алгоритм управления участком дорожной сети для двух двухфазных светофорных объектов.
4. Разработан и теоретически обоснован алгоритм управления участком реальной дорожной сети из двух многофазных светофорных объектов.
5. Создан комплекс программ, реализующий разработанные алгоритмы, для проверки их результативности применительно к дорожным сетям.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Актуальность и степень разработанности темы исследования.

Информационные технологии дают людям подходящие инструменты для удовлетворения их различных потребностей. Потребность эффективно распределять свое время появилась из-за необходимости выживать в условиях жесткого рынка труда. Одно из направлений для удовлетворения данной потребности — уменьшение времени простоя в пробках. Возможна ситуация, когда ряд происшествий частично или полностью блокирует движение на артериях города. Например, некоторые происшествия, парализовавшие или полностью остановившие движение на проспекте Свободном в городе Красноярске:

– в районе «Космоса», 12 февраля 2020 года движение под мостом было заблокировано из-за упавшей балки (у железнодорожного моста есть ограничение на высоту транспорта);
– второго августа 2013 года недалеко от Алексеевского путепровода, по нечётной стороне, произошло обрушение части подпорной стены;
– бесчисленные мелкие аварии на выезде из торгового комплекса и при повороте на улицу Курчатова.

Речь идет о потерянных часах для отдельно взятого индивидуума или о тысячах часов для всех, кто стоял в образовавшихся пробках. Проблему пробок решают с помощью настройки системы управления светофорными объектами.

Системы управления светофорными объектами подразделяют на адаптивные и неадаптивные. Неадаптивные системы светофоров переключают фазы светофоров через заранее заданное фиксированное время.

В таблице 1 представлены наиболее известные адаптивные системы управления светофорами.

Для работы адаптивных систем первого поколения задается наперед план координации. Основным приложением таких систем является регулирование фаз светофоров для разгрузки дороги в определенный период времени: утренний, дневной, вечерний, выходной день. Это требует сбора информации. Наиболее распространенная модель такого типа — Urban Traffic Control System-

Таблица 1 — Модели адаптивных систем светофоров

Критерий	**UTCS-1**	**SCOOT**	**OPAC**	**MARL**	**АСУДД «Микро»**
город	Вашингтон	Лондон	Арлингтон, Тускон	Торонто	Красноярск
временной период	1970е	1995	1983,1989	2010	1993
длительность фаз	фиксированная		переменная		
оптимизация	офлайн	онлайн			
предсказание	нет	есть		нет	есть
устройство	централизованная		децентрализованная		
основные ограничения	постоянный сбор данных	сенсоры далеко	только для 8 фаз	«ПРОКЛЯТИЕ РАЗМЕРНОСТИ»	находится в разработке

First Generation (UTCS-1) [12], разработанная в 1970-х на языке Fortran. Для работы подобной модели требуется информация об объеме пропускаемого трафика в час для обрабатываемого периода времени. Входными данными модели являются: средняя скорость машин; среднее время, затраченное на преодоление перекрестка; максимальное время для полной загрузки перекрестка. Принцип работы заключается в подборе параметров длительности желтой фазы, стрелки поворота и стоп-фазы для разных распределений трафика. Понятно, что такая модель чувствительна к объему входных данных. Отсюда следует недостаток системы — необходимость в постоянном сборе и обновлении входных данных.

Еще одна адаптивная система светофоров — Split, Cycle, and Offset Optimisation Technique (SCOOT) [13]. Перечислим основные отличия от предыдущей модели:

– непрерывный сбор и анализ информации с детекторов;
– преобразовании информации с предыдущего этапа в информацию о потоке машин;
– принятие решения о длительности фазы: исключение или увеличение времени полного цикла светофора, если у потока большая насыщенность.

Данная система не может быть применена к слишком коротким улицам и к дорогам длинной более одного километра [13].

Optimised Policies for Adaptive Control(OPAC) [14] — более совершенная система. Она отличается от систем предыдущего типа децентрализованностью и переходом к полностью переменным длительностям фаз.

На момент написания работы многие зарубежные компании одна за другой уходят с российского рынка, что делает использование из программных продуктов сложным и небезопасным. Об остановке деятельности в РФ уже известили Microsoft, Oracle, Autodesk, SAP, Poster и многие другие крупные издатели профессионального программного обеспечения. Поэтому, как отдельный класс стоит рассмотреть отечественные продукты.

Автоматизированная Система Управления Дорожным Движением (АСУДД «Микро») — наиболее широко используемая в России система. Она успешно работает в следующих городах и регионах:

- Ангарск – 32 перекрестка;
- Белгород – 122 перекрестков;
- Воронеж – 137 перекрестков;
- Иркутск – 146 перекрестков;
- Иркутская область – 15 перекрестков;
- Красноярск – 362 перекрестка;
- Красноярский край – 7 перекрестков;
- Хабаровск – 168 перекрестков;
- Московская область (ГБУ МО «МосАвтоДор») – 512 перекрестков;
- Федеральные автомагистрали (ФУАД «ЦентрАвтоМагистраль») – 264 перекрестков.

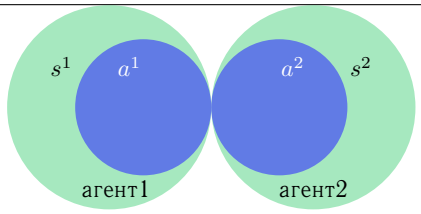
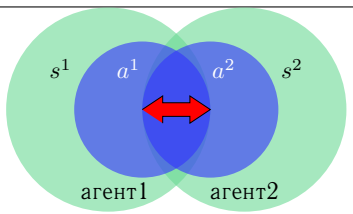
АСУДД «Микро» является децентрализованной системой и поддерживает до шести GPRS-серверов, позволяющих подключить 250 перекрестков. Отечственные видеодетекторы серии «Инфопро», используемые в АСУДД «Микро», предназначены для сбора ститистических данных о транспортном потоке и данных реального времени для актуального управления и работают на расстоянии до 70-ти метров. По дальности распознавания данные датчики не уступают их Бельгийскому аналогу TrafiCam. В данный момент основным недостатком рассматриваемой системы является тот факт, что реализация адаптивных

алгоритмов находится в разработке. В таблице 1 приведены параметры, по которым она сравнивается с другими решениями.

В последние годы с ростом вычислительных возможностей ЭВМ и развитием программных инструментов в теории мультиагентного программирования возрос интерес к изучению моделей управляемых сетей светофоров, в которых объектом интереса является уменьшение времени простоя. Задачи с такими моделями возникают в процессе работы у управляющих служб. Как альтернативный способ координации управления светофорными объектами были предложены модели мультиагентного обучения с подкреплением (MARL), основанные на марковских процессах принятия решений, которые могут не оперировать интенсивностью потока.

Задачи MARL для их эффективного применения требуют разрешения проблем совокупного управления агентов и роста размерности матриц при расширении покрытия дорожной сети. При совокупном управлении разрешаются взаимоблокировки и конфликты управления агентов по отдельности. В таблице 2 представлены схемы и описание различий моделей MARL (Multiagent Reinforcement Learning) и ее координированной модификации MARLIN (Multiagent Reinforcement Learning for Integrated Network). Также в процессе необходимых при поиске управления вычислений создаются матрицы размерности пропорциональной квадрату мощности множества всех совокупных действий агентов.

Таблица 2 — Рассматриваемые задачи мультиагентного обучения с подкреплением

Задача	Схема	Описание
MARL		Действия и состояния агентов рассматриваются по отдельности.
MARLIN		Рассматриваются допустимые совместные действия и состояния агентов.

Цели и задачи исследования. Целью работы является разработка и исследование математических моделей мультиагентной системы для задачи совокупного управления светофорными объектами дорожной сети.

Для достижения цели были поставлены и решены следующие задачи.

1. Построить математическую модель и разработать алгоритмы для управления модельным участком дорожной сети из одного светофорного объекта.
2. Построить математическую модель и разработать алгоритмы координированного и некоординированного управления модельным участком дорожной сети из двух двухфазных светофорных объектов.
3. Построить математическую модель и разработать алгоритмы координированного и некоординированного управления реальным участком дорожной сети из двух многофазных светофорных объектов.
4. Создать комплекс программ, реализующий разработанные алгоритмы, для проверки их результативности на модельных и реальных данных применительно к дорожным сетям.

Результаты, представленные в работе, имеют практическое применение.

Практическая значимость работы. Предложенный метод позволяет решить проблемы пробок на реальном участке дорожной сети города Красноярска, содержащем перекресток проспект Свободный — улица Михаила Годенко и перекресток проспект Свободный — улица Высотная. Разработанный комплекс алгоритмов и программ может быть использован в целях улучшения дорожной обстановки города Красноярска.

Апробация. Результаты работы обсуждались и докладывались на научных семинарах кафедры высшей и прикладной математики СФУ и конференциях:

- XVII Международная конференция студентов, аспирантов и молодых ученых Проспект Свободный (Красноярск, 2021; диплом III степени);
- XX Международная конференция имени А.Ф. Терпугова «Информационные технологии и математическое моделирование» ИТММ (Томск, 2021; диплом победителя);
- VIII Международная молодежная научная конференция «Математиче-

ское и программное обеспечение, технических и экономических систем» МПОиТЭС (Томск, 2021; диплом).

– XVIII Международная конференция студентов, аспирантов и молодых ученых Проспект Свободный (Красноярск, 2022; диплом II степени);
– IV Всероссийская с международным участием научно-практическая конференция студентов, аспирантов и работников образования и промышленности «Системы управления, информационные технологии и математическое моделирование» СУИТиММ (Омск, 2022; диплом).

Публикации. По тематике работы опубликовано **3** работы [10], [11], [9].

Краткое содержание работы. Бакалаврская работа состоит из введения, трех глав, заключения, списка литературы и приложения. Общий объем работы составляет 53 страницы; иллюстративный материал представлен 23 рисунками и 6 таблицами; список литературы содержит 16 наименований.

В главе 1 рассмотрены задача подсчета функции оценки эффективности принятого управления и задача поиска оптимального управления одним светофорным объектом. Описание обеих задач приведено в параграфе 1.2. В параграфе 1.3 приведены математическая постановка и решение первой задачи. В параграфах 1.4, 1.5 приведены постановка и решение второй задачи, а также в параграфе 1.5 доказан критерий единственности точного решения и представлен алгоритм поиска приближенного решения. В параграфе 1.6 была произведена серия вычислительных экспериментов, в результате которой получено численное выражение того, насколько управляемая марковским процессом модель эффективнее справляется с пробками по сравнению с фиксированным планом смены фаз светофорных объектов.

В главе 2 рассмотрена задача поиска оптимального управления модельной дорожной сетью, состоящей из двух светофорных объектов. В параграфе 2.1 приведено описание рассматриваемой задачи. Алгоритм совокупного управления светофорными объектами, позволяющий свести задачу совокупного управления несколькими светофорными объектами к задаче управления одним светофорным объектом, был предложен в параграфе 2.2. Также в параграфе 2.3 были произведены серии вычислительных экспериментов для до-

рожной сети из двух двухфазных светофорных объектов. Основным результатом главы 2 стало подтверждение эффективности применяемого управления.

В главе 3 поставлена и решена задача поиска оптимального управления для реальной дорожной сети, содержащей многофазные светофорные объекты. Описание данной задачи приведено в параграфе 3.1. Разработан комплекс программ, связывающих показания оптического датчика и переменные моделируемой среды и проведены серии вычислительных экспериментов для реальной дорожной сети, содержащей многофазные светофорные объекты. Код представлен в Приложении А. Результаты и подробное описание вычислительных экспериментов приведены в параграфе 3.2. Основным результатом главы 3 стало подтверждение эффективности применяемого управления для реальных участков дорожной сети.

Глава 1. Задача управления одним светофорным объектом

В качестве математической модели сети светофоров в работе рассматривается управляемый марковский процесс с конечным числом действий и состояний. Каждый агент (светофорный объект) не располагает ресурсами и решает задачу целесообразности выбора того или иного сигнала. Среда — перекресток с машинами, где на отрезках дорог за сто метров до стоп-линий засекается время. Состояние среды отражает активность фазы светофора. Тогда проблема управления светофорными объектами сводится к задаче мультиагентного обучения с подкреплением (Multiagent Reinforcement Learning). Как правило, обучение с подкреплением используется для одного агента в среде, чтобы максимизировать его долгосрочную награду. Модель среды — марковский процесс принятия решения. В работе исследуется модель обучения с подкреплением (Q-обучения) одного и нескольких агентов (светофорных объектов) в стационарной среде. Обучение с подкреплением сводится к оптимальному сопоставлению агентом действия a состоянию среды s. Предложенная математическая модель процесса выбора фазы светофора учитывает текущее расположение светофоров и их загрузки и позволяет сформулировать оптимизационные задачи, целью которых является минимизация задержки трафика автомобилей. Отметим, что структура мультиагентной системы обеспечивает наиболее эффективное распараллеливание всей задачи на подзадачи, которые будут решены агентами.

1.1 Основные определения и обозначения

Приведем основные определения и обозначения, которые используются в работе, согласно [16].

Определение 1.1. Интеллектуальным агентом называется метаобъект, наделенный долей субъектности, взаимодействующий с другими агентами и средой, выполняющий определенные функции для достижения поставленных целей.

Определение 1.2. Средой называется множество объектов, не принадлежащих агенту.

Определение 1.3. Мультиагентная система (Multy Agent System, MS) — совокупность взаимодействующих агентов и среды.

Определение 1.4. Мультиагентное обучение с подкреплением (Multy Agent Reinforcement Learning, MARL) — процесс подкрепленного обучения агентов в стохастической среде, где в роли учителя выступает среда.

В работе используются следующие определения и обозначения из теории марковских процессов [4].

Определение 1.5. Случайный процесс $X(t), t \in \mathbb{R}$, называют марковским процессом [4], если $\forall n \in \mathbb{N}$, $\forall t_1 < t_2 < \cdots < t_n < t_{n+1}$, $\forall x \in \mathbb{R}$, выполняется:

$$\begin{aligned} &p(\{X(t_{n+1}) \mid X(t_n) = x_n, X(t_{n-1}), \dots X(t_1)\}) = \\ &= p(\{X(t_{n+1}) \mid X(t_n) = x_n\}). \end{aligned} \tag{1.1}$$

Определение 1.6. Марковской цепью называется пара $\langle \mathcal{S}, \mathbb{P} \rangle$, где $\mathcal{S}$ — дискретное множество состояний; $\mathbb{P}$ — матрица вероятностей переходов из одного состояния в другое.

Определение 1.7. Марковская цепь называется однородной или стационарной, если вероятности переходов не зависят от времени:

$$\forall t \in \mathbb{N}, \forall s^{(k)}, s^{(j)} \in \mathcal{S} : p\left(s_{t+1} = s^{(j)} \mid s_t = s^{(k)}\right) = p\left(s_1 = s^{(j)} \mid s_0 = s^{(k)}\right).$$

Как и в оптимальном управлении [2], для моделирования влияния агента на среду в вероятности переходов достаточно добавить зависимость от выбираемых агентом действий.

Рассматриваемая далее модель среды — «управляемая» однородная марковская цепь с дискретным временем и конечным числом действий и состояний.

Определение 1.8. Средой называется тройка$\langle \mathcal{S}, \mathcal{A}, \mathbb{P} \rangle$, где $\mathcal{S}$ — дискретное множество состояний среды; $\mathcal{A}$ — дискретное множество действий агентов; $\mathbb{P}$ — общая матрица переходов.

Определение 1.9. Вознаграждение в момент времени t в состоянии $s_t \in \mathcal{S}$ после действия $a_t \in \mathcal{A}$ есть функция $r(s_{t+1} \mid s_t, a_t)$, которая всецело определяется текущим состоянием s_t, выбранным действием a_t и состоянием $s_{t+1} \in \mathcal{S}$, в которое перейдет процесс на следующем шаге.

Определение 1.10. Марковский процесс принятия решений представляется в виде кортежа $(\mathcal{S}, \mathcal{A}, \mathbb{P})$, где $(\mathcal{S}, \mathcal{A}, \mathbb{P}, r)$ — среда; $r(s_t, a_t)$ — функция вознаграждения.

Определение 1.11. Семейство выбранных решений $a_t \in \mathcal{A}$, принятых в момент времени t в состоянии $s_t \in \mathcal{S}$, образует управление $\delta = \{a_t\}_{0 \le t < \infty}$. Принято обозначать решение a_t как $\delta(s_t)$.

Определение 1.12. Политикой агента называется условное распределение $\pi(a \mid s) = p(a \mid s)$ дискретной случайной величины (s, a), с законом распределения $\mathrm{P}\colon\ \mathcal{S} \times \mathcal{A} \to [0; 1]$.

Определение 1.13. Набор $\mathcal{T} = \{s_0, a_0; s_1, a_1; s_2, a_2; \dots\}$, где $\forall j$ выполнено $a_j \in \mathcal{A}, s_j \in \mathcal{S}$, называется траекторией.

Определение 1.14. Для данной среды $\langle \mathcal{S}, \mathcal{A}, \mathbb{P} \rangle$, политики π и начального состояния $s_0 \in \mathcal{S}$ распределение, из которого строятся траектории $\mathcal{T}$, называется распределением по траектории и записывается в виде

$$p(\mathcal{T}) = \prod_{t>0} \pi(a_t \mid s_t) p(s_{t+1} \mid a_t, s_t). \tag{1.2}$$

Поведение агента будем описывать марковским процессом принятия решения $\langle \mathcal{S}, \mathcal{A}, \mathbb{P}, r \rangle$. Процесс принятия решения агентом будет выглядеть следующим образом [5]. В момент времени t агент наблюдает состояние среды $s_t \in \mathcal{S}$ и выбирает действие $a_t \in \mathcal{A}$. Среда отвечает генерацией награды $R_t = r(s_t, a_t)$ и переходит в следующее состояние $s_{t+1} = s'$ с вероятностью $p(s' \mid s_t, a_t)$. Следует отметить, что процесс выбора действия агентом в текущем состоянии может быть как детерминированным, так и стохастическим. Эффективность управления (поведения агента) $\delta = \{a_t, 0 \le t < \infty\}$ оценивается с помощью некоторой кумулятивной функции вознаграждения агента $V : \langle \mathcal{S}, \mathcal{A}, \mathbb{P}, r \rangle \to \mathbb{R}$, определяемую по формуле

$$V(\delta) = \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) = \sum_{t=0}^{\infty} \gamma^t R_t, \tag{1.3}$$

где величина γ, $0 < \gamma < 1$, называется коэффициентом переоценки и показывает во сколько раз уменьшается отложенное вознаграждение за один временной шаг [5]. Переоценка задает приоритет получения награды в ближайшее

время перед получением той же награды через некоторое время. Математически смысл условия $0 < \gamma < 1$ состоит в том, чтобы гарантировать ограниченность функционала V. Предпочтительным управлением δ будет то, значение целевой функции $V(\delta)$ для которого будет больше.

1.2 Описание задачи

Рассмотрим регулируемый перекресток, на который с четырех примыкающих к нему дорог заезжают автомобили. На каждой из дорог размещены детекторы. Схема описываемого перекрёстка представлена на рисунке 1.1.

Построим модель дорожной сети, состоящей из одного светофорного объекта. Считаем, что светофорный объект — агент, не располагающий ресурсами. В качестве среды будем рассматривать перекресток с машинами, где на отрезках дорог за 100 метров до стоп-линий засекается время проезда через него. Состояние среды отражает активность фазы светофора.

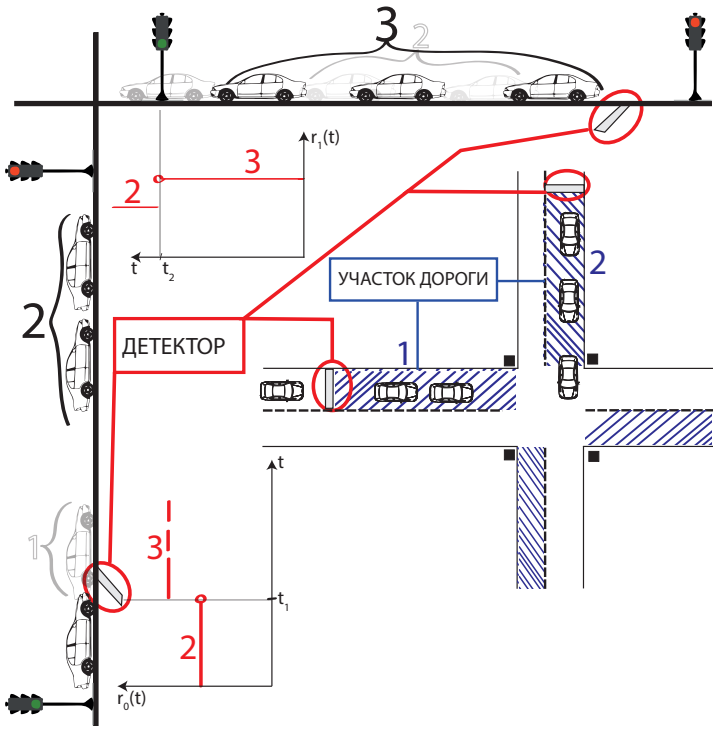


Рисунок 1.1 — Схема перекрестка с одним управляемым светофорным объектом

Введём следующие обозначения: $\mathcal{S}$ — дискретное пространство состояний агента, $\mathcal{A}$ — дискретное пространство решений агента, $\delta\colon \mathcal{S} \to \mathcal{A}$ —

управление сменами фаз светофора. Фазы светофора меняются последовательно. Полагаем, что множество $\mathcal{S}$ есть кольцо классов вычетов целых чисел $\mathbb{Z}_n = \langle \mathbb{Z}, +, \cdot \rangle$, где $|\mathcal{S}| = n$ — количество классов $\mathbb{Z}_n$ [3]. Характеристика кольца вычетов $\mathbb{Z}_n$ равна n и в данной модели отражает число смен фазы светофора до её возвращения к начальному значению, следовательно, число действий $|\mathcal{A}|$ до возвращения в начальное состояние светофора равно n. Если агент находится в состоянии s, то при выборе действия $a = \delta(s)$ новое состояние s' определяется формулой

$$s' = (a + s) \mod |\mathcal{S}|. \tag{1.4}$$

Далее будем считать, что среда, находясь в состоянии $s \in \mathcal{S}$, ожидает от агента действия $a \in \mathcal{A}$, после чего совершает шаг, переходя в состояние s' согласно формуле (1.4) с вероятностью $p(s'|s, a)$. Условные вероятности $p(s'|s, a)$, для всех $s, s' \in \mathcal{S}$ и действий $a \in \mathcal{A}$ образуют матрицы переходов цепи Маркова $\mathbb{P}$. При фиксированном действии a выполняются следующие свойства

– условие нормировки $\sum_{s' \in \mathcal{S}} p(s' \mid s, a) = 1$;

– свойство марковости (независимость переходов от истории):

$$p\left(s_{t+1} \mid s_t, \ldots, s_0\right) = p\left(s_{t+1} \mid s_t\right);$$

– свойство стационарности (независимость от времени) — вероятности переходов между состояниями не меняются со временем

$$p\left(s_{t+1} \mid s_t\right) = p\left(s_1 \mid s_0\right), \quad \forall t = 0, 1, 2, \ldots;$$

– предполагается, что множество действий не меняется со временем и не зависит от состояния.

Таким образом, в качестве модели среды будем рассматривать управляемую марковскую цепь с дискретным временем $\langle \mathcal{S}, \mathcal{A}, \mathbb{P} \rangle$ [5].

Пример 1.1. *Рассмотрим приведённую модель на примере двухфазного светофора. Предполагается, что светофорный объект может находиться в двух состояниях $s^{(0)}$ и $s^{(1)}$ и менять их при помощи действий $a^{(0)}$ и $a^{(1)}$, т.е.*

$\mathcal{S} = \left\{s^{(0)} = 0, s^{(1)} = 1\right\}$ *и* $\mathcal{A} = \left\{a^{(0)} = 0, a^{(1)} = 1\right\}$. *Здесь состояние* $s^{(i)}$ *интерпретируется как «активна фаза* i*»,* $i = \{0, 1\}$*; действие* $a^{(0)}$ *интерпретируется как «оставить фазу»,* $a^{(1)}$ *— «сменить фазу». Согласно (1.4) состояние* s'*, в которое перейдёт система из состояния* s *при решении* a*, однозначно определяется по формуле* $s' = (a + s) \bmod 2$*. На рисунке 1.2 представлен стохастический граф управляемого процесса смены фазы для двухфазного светофорного объекта.*

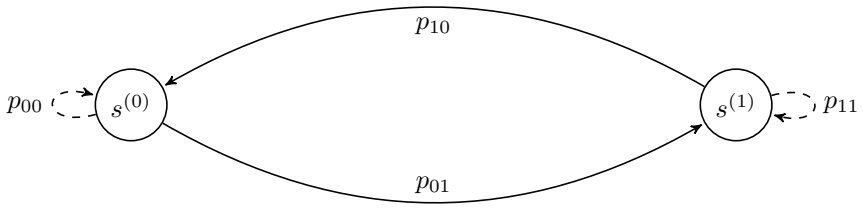


Рисунок 1.2 — Стохастический граф управляемого процесса смены фаз для двухфазного светофорного объекта. Здесь пунктиром обозначены переходы при действии $a^{(0)}$, а сплошной линией переходы при решении $a^{(1)}$.

Динамика среды $\mathbb{P}$ *определена вероятностями* $p_{00}, p_{01}, p_{10}, p_{11}$ *следующим образом*

$$\begin{aligned} p_{00} &= p\left(s^{(0)}|s^{(0)}, a^{(0)}\right), \quad p_{01} = p\left(s^{(1)}|s^{(0)}, a^{(1)}\right), \\ p_{10} &= p\left(s^{(0)}|s^{(1)}, a^{(1)}\right), \quad p_{11} = p\left(s^{(1)}|s^{(1)}, a^{(0)}\right). \end{aligned} \tag{1.5}$$

Такое определение динамики среды приводит к вырожденным распределениям $p(s'|s)$ *при фиксации действия. В таблице 1.1 представлены условные вероятности перехода из состояния* s *в состояние* s' *с учетом выбора действия* a*. В связи с полной определенностью следующего состояния* s' *выбранными действием* a *и текущим состоянием* s*, возникает вопрос о случайности процесса. По самому смыслу управления естественно было бы считать, что стохастика заложена в выборе текущего действия.*

Таблица 1.1 — Распределение $p(s' \mid s, a)$

a	s \ s'	$s^{(0)}$	$s^{(1)}$
$a^{(0)}$	$s^{(0)}$	1	0
	$s^{(1)}$	0	1
$a^{(1)}$	$s^{(0)}$	0	1
	$s^{(1)}$	1	0

Заметим, что для случая, когда рассматривается сеть светофоров с двумя состояниями, следующую фазу светофора можно найти с помощью операции «исключающего или»:

$$\boldsymbol{s}' = \boldsymbol{a} \oplus \boldsymbol{s}.$$

Далее, состояние в момент времени t будем обозначать s_t, а действие — a_t.

Приведем описание, исходя из которого считается функция вознаграждения. Для каждой полосы определено число машин на отрезке дороги, начинающегося с детектора и заканчивающегося стоп-линией перекрестка (рис. 1.1). Пусть $r : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ — функция вознаграждения агента при изменении состояния s при действии $a = \delta(s)$. В момент времени t значение функции $r(s_t, a_t) = R_t$ определяется для следующей активной полосы и пропорционально времени, затраченному всеми машинами на преодоление детектируемых участков дороги.

Управление светофорным объектом будем описывать марковским процессом принятия решения $\langle \mathcal{S}, \mathcal{A}, \mathbb{P}, r \rangle$. Процесс принятия решения будет выглядеть следующим образом.В момент времени t активна фаза светофорного объекта $s_t \in \mathcal{S}$ и выбрано управление $a_t \in \mathcal{A}$. Далее идет подсчет награды $R_t = r(s_t, a_t)$, и происходит смена активной фазы светофорного объекта на $s_{t+1} = s'$ с вероятностью $p(s' \mid s_t, a_t)$. Следует отметить, что процесс подбора управления светофорным объектом при активной фазе может быть как детерминированным, так и стохастическим. Эффективность управления (поведения агента) $\delta = \{a_t, 0 \leq t < \infty\}$ оценивается с помощью некоторой кумулятивной функции вознаграждения агента $V : \langle \mathcal{S}, \mathcal{A}, \mathbb{P}, r \rangle \to \mathbb{R}$, определяемую по

формуле

$$V(\delta) = \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) = \sum_{t=0}^{\infty} \gamma^t R_t, \tag{1.6}$$

где величина γ, $0 < \gamma < 1$, называется коэффициентом переоценки и показывает во сколько раз уменьшается отложенное вознаграждение за один временной шаг [5]. Под отложенным до момента времени t вознаграждением принято понимать число $\gamma^t R_t$.

Предложение 1.2.1. *Функция оценки эффективности управления светофорным объектом ограничена.*

Доказательство. Рассмотрим функцию $V(\delta) = \sum\limits_{t=0}^{\infty} \gamma^t R_t$. Заметим, что значения вознаграждений R_t неотрицательны и ограничены сверху некоторой константой, поскольку они пропорциональны времени, затраченному всеми машинами на преодоление детектируемых участков дороги, и зависят от числа машин на участке дорожной сети в момент времени t. Следовательно, $\sum\limits_{t=0}^{\infty} \gamma^t R_t$ – положительный числовой ряд. Покажем, что ряд $\sum\limits_{t=0}^{\infty} \gamma^t R_t$ сходится.

Обозначим $r_{\max} = \max\limits_t r_t$. Учитывая, что $0 < r_{\max} < \infty$ и $0 < \gamma < 1$, получаем следующую оценку

$$0 \leqslant \sum_{t=0}^{\infty} \gamma^t R_t \leqslant \sum_{t=0}^{\infty} \gamma^t r_{\max}. \tag{1.7}$$

Для ряда $\sum\limits_{t=0}^{\infty} \gamma^t r_{\max}$ выполнен радикальный признак Коши для положительных числовых рядов [6]. Действительно, предел t-й степени из общего члена ряда есть

$$\lim_{t\to\infty} \sqrt[t]{\gamma^t r_{\max}} = \lim_{t\to\infty} \gamma \sqrt[t]{r_{\max}} = \gamma \lim_{t\to\infty} \sqrt[t]{r_{\max}} = \gamma < 1.$$

Тогда из сходимости ряда $\sum\limits_{t=0}^{\infty} \gamma^t r_{\max}$ и оценки (1.7) следует сходимость ряда $\sum\limits_{t=0}^{\infty} \gamma^t r_t$ по признаку сравнения положительных числовых рядов [6]. В силу ограниченности сходящегося положительного числового ряда, функция $V(\delta)$ ограничена. Сумму ряда $\sum\limits_{t=0}^{\infty} \gamma^t r_{\max}$ можно вычислить по формуле бесконечно убывающей геометрической прогрессии. Итак, для функции V выполнено

двойное неравенство

$$0 \leqslant V \leqslant \frac{1}{1-\gamma} r_{\max}.$$

Что и требовалось доказать. $\diamond$

Изучение поведения агента при различных условиях является предметом теории исследования операций [2]. Задачи в теории исследования операций принято разделять на прямые и обратные. Прямые задачи исследования операций отвечают на вопрос: чему будет равна целевая функция V, если в заданных условиях будет принято управление δ? Обратные задачи отвечают на вопрос: как выбрать управление δ, максимизирующее целевую функцию V?

Заметим, что для решения обратной задачи прежде всего необходимо уметь решать прямую.

1.3 Задача вычисления функции оценки эффективности управления

Рассмотрим решение прямой задачи для двухфазного светофора из примера 1.1, описываемого марковского процесса принятия решения $\langle \mathcal{S}, \mathcal{A}, \mathbb{P}, r \rangle$. Согласно примеру 1.1 множество состояний есть $\mathcal{S} = \left\{ s^{(0)} = 0, s^{(1)} = 1 \right\}$, множество действий $\mathcal{A} = \left\{ a^{(0)} = 0, a^{(1)} = 1 \right\}$. Динамика среды $\mathbb{P} = \{p_{00}, p_{01}, p_{10}, p_{11}\}$ задается формулой (1.5). Считаем, что процесс выбора агентом действия $a \in \mathcal{A}$ при фиксированном состоянии $s \in \mathcal{S}$ подчиняется условному распределению $\pi(a|s)$. Награды $r_t = r(s_t, a_t)$ для каждого момента времени t рассчитываются как суммарное время машин, находящихся на активируемом фазой $a \oplus s$ детектируемом участке дорожной сети. Например, на рисунке 1.1 переменная датчика до шага t_1 полосы 1 хранила значение $r_0(t_1) = 2$. Так как, на детектируемом участке в течение $\Delta t = t_1 - t_0$ секунд находилось $r_0(t_1)$ машин, награда равна $r(s, a) = 2\Delta t$. Функция оценки эффективности управления светофорным объектом $V(\delta)$ определяется по формуле (1.6).

Заметим, что выбор стратегии $\delta = \{a_0, a_1, \ldots\}$ определяет траекторию процесса $\mathcal{T} = (a_0, s_0, a_1, s_1, \ldots)$. На рисунке 1.3 показаны возможные траектории рассматриваемого марковского процесса принятия решения и вознаграждения на каждом шаге по t. На шаге t_0 вознаграждение за смену активной фазы $s^{(0)}$ на $s^{(1)}$ будет равно 2.

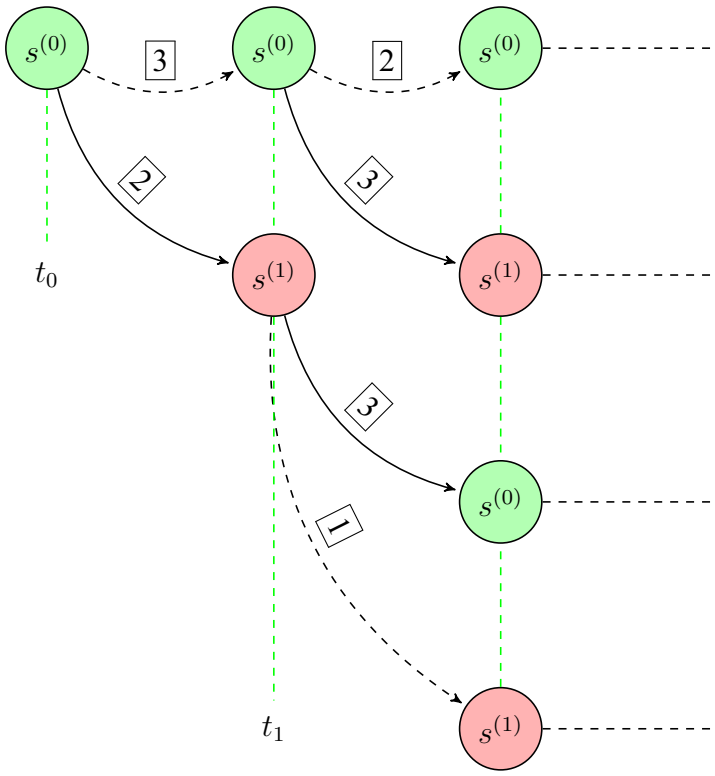


Рисунок 1.3 — Траектории марковского процесса принятия решения для примера 1.1

Исходя из вышесказанного, можно дать формальную постановку задачи вычисления оценки эффективности управления светофорным объектом в следующем виде.

Заданы: марковский процесс принятия решения $\langle \mathcal{S}, \mathcal{A}, \mathbb{P}, r \rangle$ для управления одним светофорным объектом, стратегия δ.

Требуется найти: функцию оценки эффективности управления двухфазным светофорным объектом V_δ на шаге T.

1.4 Задача поиска оптимального управления

При решении обратной задачи ищется такое управление δ, доставляющее максимум целевой функции V. Управление определяется политикой π агента в каждом состоянии s. В параграфе 1.3 была рассмотрена прямая задача исследования операций в случае, когда целевая функция V зависит только от заранее известных параметров. Предполагается, что в случае обратной задачи неизвестно управление δ.

В качестве метода поиска управления в работе используется алгоритм мультиагентного обучения с подкреплением (Multiagent Reinforcement Learning, MARL) [5,16]. Данный алгоритм разработан для математической модели марковского процесса принятия решений, задающегося кортежем $(\mathcal{S}, \mathcal{A}, \mathbb{P}, r)$.

Определения и обозначения задачи MARL для одного светофорного объекта даны согласно [4].

Пусть в момент времени t активна фаза светофора $s \in \mathcal{S}$ и управление $a = \delta(s) \in \mathcal{A}$, задержка на фазе $r(s,a)$ — значение, хранящееся в переменной детектора R. Свойство переменной R быть стохастической величиной является вполне естественным: оно отражает тот факт, что в реальных условиях при неизвестной траектории процесса её невозможно наперед задать. Функция $r(s,a)$ определяется как математическое ожидание случайной величины

$$r(s,a) = \mathbb{E}R = \sum_r r \sum_{s' \in \mathcal{S}} p(s', r \mid s, a). \tag{1.8}$$

Из равенства $p(s', r \mid s, a) = p(s' \mid s, a)p(r \mid s, a, s')$ следует

$$r(s,a) = \sum_r r \sum_{s' \in \mathcal{S}} p(r \mid s, a, s')p(s' \mid s, a) = \sum_r rp(r \mid s, a). \tag{1.9}$$

Функция наград всецело определяется текущим состоянием s и выбранным действием a, в силу того что они полностью определяют s':

$$r(s,a,s') = \sum_r r \sum_{s' \in \mathcal{S}} p(r \mid s, a, s') = \sum_r rp(r \mid s, a) = r(s,a). \tag{1.10}$$

Эффективность управления (поведения агента) $\delta = \{a_t, 0 \le t < \infty\}$ получается из формул (1.8) – (1.10) в виде кумулятивной функции вознаграждения агента:

$$V(s) = \mathbb{E} \sum_{t=0}^{\infty} \gamma^t r(s_t, \delta(s_t)), \tag{1.11}$$

где коэффициент переоценки $0 < \gamma < 1$.

Управление фазами одного светофорного объекта находится при решении задачи обучения с подкреплением одного агента в среде (RL — Reinforcement Learning) и формулируется следующим образом:

Заданы: марковский процесс принятия решения $\langle \mathcal{S}, \mathcal{A}, \mathbb{P}, r \rangle$ для управления светофорным объектом,
активная в начальный момент времени фаза светофорного объекта.

Требуется найти: управление светофорного объекта $\delta^* = \{a_t^*\}_{0 \le t < \infty}$, которое доставит максимум функции оценки его эффективности (1.11).

1.5 Решение задачи поиска оптимального управления

Решение для обратной задачи находится методом динамического программирования на основе принципа оптимальности Беллмана [16].

Предложение 1.5.1. *В задаче управления фазами светофорного объекта уравнение Вальда-Беллмана имеет вид*

$$V^*(s) = \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} p(s' \mid s, a)(r(s, a) + \gamma V^*(s')). \tag{1.12}$$

Доказательство. Раскроем математическое ожидание в формуле (1.11)

$$\begin{aligned} V(s) = \mathbb{E} \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) = \sum_r \sum_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} p(r, s', a \mid s) \Big(r + \gamma V(s') \Big) = \\ = \sum_r \sum_{a \in \mathcal{A}} p(a \mid s) \sum_{s' \in \mathcal{S}} p(r, s' \mid s, a) \Big(r + \gamma V(s') \Big). \end{aligned} \tag{1.13}$$

По свойству условной вероятности

$$p(s' \mid s, a) = \sum_r p(s', r \mid s, a). \tag{1.14}$$

Подставляя (1.14) в (1.13) получим:

$$\begin{aligned} V^*(s) = \max_{\delta} V(s) = \max_{a \in \mathcal{A}} \sum_r \sum_{s' \in \mathcal{S}} p(r, s' \mid s, a) \Big(r + \gamma V(s') \Big) = \\ = \max_{a \in \mathcal{A}} \Big(r(s, a) + \sum_{s' \in \mathcal{S}} p(s' \mid s, a) \gamma V(s') \Big) = \\ = \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} p(s' \mid s, a) \Big(r(s, a) + \gamma V(s') \Big). \end{aligned}$$

Что доказывает справедливость (1.12). $\diamond$

Утверждение об единственности решения задачи для управления одним агентом в среде схематично доказано в работе [4], приведем подробное доказательство утверждения о единственности решения задачи управления светофорным объектом с любым количеством фаз.

Предложение 1.5.2. *Для задачи поиска оптимального управления светофорным объектом с любым количеством фаз справедливы следующие утверждения*

– *существует единственное точное решение;*

– *оценка точности приближенного решения на n-ом шаге итерации*

$$\rho(Q_n, Q_0) \leqslant \frac{\gamma^n \rho(Q_1, Q_0)}{1-\gamma},$$

где $Q_t \in \mathbb{R}_\infty^{|\mathcal{A}|+|\mathcal{S}|}$ — вектора значений функции $Q(s,a)$ на шаге t,
$\forall q, w \in \mathbb{R}_\infty^{|\mathcal{A}|+|\mathcal{S}|}$ определена функция $\rho(q,w) = \max\limits_{1 \leqslant j \leqslant |A|+|S|} |q_j - w_j|$;

– *приближенное решение находится согласно формулам*

$$V^*(s) = \max_{a \in \mathcal{A}} \lim_{t \to +\infty} Q_t(s,a), \tag{1.15}$$

$$a_t(s) = \arg\max_{a' \in \mathcal{A}} Q_t(s,a'). \tag{1.16}$$

Доказательство. Введя обозначение

$$Q(s,a) = \sum_{s' \in \mathcal{S}} p(s' \mid s,a) r(s' \mid s,a) + \gamma V^*(s'), \tag{1.17}$$

перепишем задачу о поиске управления в следующем виде

$$V^*(s) = \max_{a \in \mathcal{A}} Q(s,a). \tag{1.18}$$

Подставив в уравнение (1.17) выражение для поиска управления (1.18), получим:

$$Q(s,a) = \sum_{s' \in \mathcal{S}} p(s' \mid s,a)(r(s' \mid s,a) + \gamma \max_{a' \in \mathcal{A}} Q(s',a')). \tag{1.19}$$

Запись (1.19) задает функцию Q итеративно, перепишем ее в следующем виде

$$Q_{k+1}(s,a) = \sum_{s' \in \mathcal{S}} p(s' \mid s,a)(r(s' \mid s,a) + \gamma \max_{a' \in \mathcal{A}} Q_k(s',a')). \tag{1.20}$$

Из курса функционального анализа [6] известно, что между множеством операторов $Q_k : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ и множеством векторов вида $\{Q_k(s,a)\}_{s\in\mathcal{S},a\in\mathcal{A}}$ есть взаимно-однозначное соответствие (биекция) f. Для доказательства данного факта воспользуемся определением биекции [1] и проверим, что данное соответствие f инъективно и сюръективно.

Покажем, что для отображения f выполнено свойство инъективности. Пусть операторы Q_m и Q_n не равны, т.е. $\forall(s,a) \in \mathcal{S}\times\mathcal{A}$ выполнено условие $\exists(st,ac) \in \mathcal{S}\times\mathcal{A}$ такая, что $Q_m(st,ac) \neq Q_n(st,ac)$. Полагая по определению f, что $f(Q_m) = \{Q_m(s,a)\}_{s\in\mathcal{S},a\in\mathcal{A}}$ и $f(Q_n) = \{Q_n(s,a)\}_{s\in\mathcal{S},a\in\mathcal{A}}$, получаем неравенство для (st,ac)-ых компонент векторов. Действительно, в силу неравенства $\{Q_m(s,a)\}_{s\in\mathcal{S},a\in\mathcal{A}} \neq \{Q_n(s,a)\}_{s\in\mathcal{S},a\in\mathcal{A}}$ имеем $f(Q_m) \neq f(Q_n)$. Таким образом, свойство инъективности для отображения f доказано.

Свойство сюръективности выполнено для f в силу того, что $\mathcal{S},\mathcal{A}$ — конечные дискретные множества и для каждого вектора $\{Q(s,a)\}_{s\in\mathcal{S},a\in\mathcal{A}}$ можно таблично задать $Q(s,a)$. Действительно, если известен вектор $\{Q(s,a)\}_{s\in\mathcal{S},a\in\mathcal{A}}$, то известно бинарное отношение $Q \subset (\mathcal{S}\times\mathcal{A})\times\mathbb{R}$, которое по определению и есть оператор Q.

Итак, $Q = \{Q(s,a)\}_{s\in\mathcal{S},a\in\mathcal{A}}$, можно записать итеративно $Q_{t+1} = A(Q_t)$, где $A\colon \mathbb{R}_\infty^{|\mathcal{A}|+|\mathcal{S}|} \to \mathbb{R}_\infty^{|\mathcal{A}|+|\mathcal{S}|}$ — сжимающее отображение, определяемое по формуле (1.20).

Известно [6, стр. 74], что метрическое пространство со стандартной метрикой $\left\langle \mathbb{R}_\infty^{|\mathcal{A}|+|\mathcal{S}|}, \rho(\cdot,\cdot) \right\rangle$ является полным.

Для отображения A выполнено неравенство

$$\begin{aligned}
\rho\left(A\circ Q_1, A\circ Q_2\right) = \max_{a\in A, s\in\mathcal{S}} \Big| \sum_{s'\in\mathcal{S}} p(s' \mid s,a)(r(s,a) + \gamma \max_{a'\in\mathcal{A}} Q_1(s',a')) - \\
- \sum_{s'\in\mathcal{S}} p(s' \mid s,a)(r(s,a) + \gamma \max_{a'\in\mathcal{A}} Q_2(s',a')) \Big| \leqslant \\
\leqslant \max_{a\in\mathcal{A}, s\in\mathcal{S}} \left|\gamma \max_{a'\in\mathcal{A}} Q_1(s',a') - \gamma \max_{a'\in\mathcal{A}} Q_2(s',a')\right| = \\
= \gamma\rho(Q_1,Q_2), \qquad \gamma\in(0;1).
\end{aligned}$$

Согласно принципу сжимающих отображений [6], в полном метрическом пространстве существует одна и только одна неподвижная точка отображения, т.е.

решение $A \circ Q = Q$ существует и единственно. Что доказывает первое утверждение.

Последовательность $\{Q_t\}$ представляет собой приближенное решение уравнения $A \circ Q = Q$, эффективный способ оценки точности которого:

$$\begin{aligned}\rho(Q_{t_n}(s,a), Q_{t_0}(s,a)) = \rho((A^n \circ Q_{t_0})(s,a), Q_{t_0}(s,a)) \leqslant \\ \leqslant \frac{\gamma^n \rho(Q_{t_1}(s,a), Q_{t_0}(s,a))}{1-\gamma} = \\ = \frac{\gamma^n \rho((A \circ Q_{t_0})(s,a), Q_{t_0}(s,a))}{1-\gamma}.\end{aligned}$$

Что доказывает второе утверждение.

Идея обучения с подкреплением заключается в оценке невычислимой правой части:

$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha_t(s,a)\left(r(s,a) + \gamma \max_{a' \in \mathcal{A}} Q_t(s',a') - Q_t(s,a)\right). \quad (1.21)$$

где s' — фаза светофорного объекта в момент времени $t+1$, если в момент времени t была активна фаза светофорного объекта s и было выбрано действие a. Если в момент времени t была активна фаза светофорного объекта s и было выбрано действие a, то $0 < \alpha_t(s,a) \leqslant 1$, иначе $\alpha_t(s,a) = 0$.

$$V_t^*(s) = \max_{a \in \mathcal{A}} Q_t(s,a).$$

Что доказывает третье утверждение.

Заметим, что если используемая стратегия $\delta(s)$ приводит к тому, что с вероятностью 1 каждая пара (s,a) будет бесконечное число раз встречаться на бесконечном горизонте наблюдения, то из отмеченного выше условия сжимаемости при оценках вероятности

$$\sum_{t=0}^{\infty} \alpha_t(s,a) = \infty, \qquad \sum_{t=0}^{\infty} \alpha_t(s,a)^2 \leqslant \infty$$

будет следовать сходимость (с вероятностью 1) процесса (1.21), откуда следует справедливость формул (1.15), (1.16). $\diamond$

1.6 Вычислительные эксперименты

Вычислительные эксперименты проводились на персональном компьютере с процессором Intel® Core™ i7-10510U CPU @1.80ГГц и оперативной памятью объемом 8ГБ.

1.6.1 Вычисление функции оценки эффективности управления для двухфазного светофорного объекта

Для поиска решения прямой задачи управления одним двухфазным светофором из примера 1.1 был разработан комплекс программ в интегрированной среде разработки Visual Studio Community 2019 на языке программирования C++11 и проведена серия вычислительных экспериментов.

Целью эксперимента был расчет показателей эффективности V при различном задаваемом управлении δ для прямой задачи исследования операций.

Для моделирования использовались наблюдения об интенсивности трафика и планов координации двухфазного светофора с перекрёстка пр. Свободный — ул. Лесопарковая, г. Красноярска за период с 2017 по 2018 год.

На основании этих данных сформирована двумерная выборка $\mathcal{X} = \{(s_i, a_i)\}_{i=1}^{N}$ объемом N порядка 10^6. В результате получена несмещенная оценка распределения $\mathcal{P} = \{p(s,a), s \in \mathcal{S}, a \in \mathcal{A}\}$ двумерной случайной величины (s,a), где $p(s,a)$ — вероятность того, что в состоянии s агент принял решение a. На основании выборочных вероятностей $\hat{p}(s,a)$ рассчитаны политики агента $\hat{\pi}(a|s)$ для каждого $s \in \mathcal{S}$

$$\hat{\pi}(a|s) = \frac{\hat{p}(s,a)}{\sum\limits_{a \in \mathbb{A}} \hat{p}(s,a)} = \frac{\hat{p}(s,a)}{\hat{p}(s)}.$$

Наряду с политиками агента, при обработке интенсивностей получены массивы $r_{a^{(k)}} = \left\{r(s_0, a^{(k)}), r(s_1, a^{(k)}), r(s_2, a^{(k)}), \dots\right\}$, $k = 0, 1$. Элементы этих массивов $r\left(s_t, a^{(k)}\right)$ вычислены как время нахождения машин на активируемых фазой $a^{(k)} \oplus s_t$ полосах. Отсчет времени начинается с момента обнаружения машины датчиками. Результат обработки интенсивности трафика представлен на рисунке 1.4.

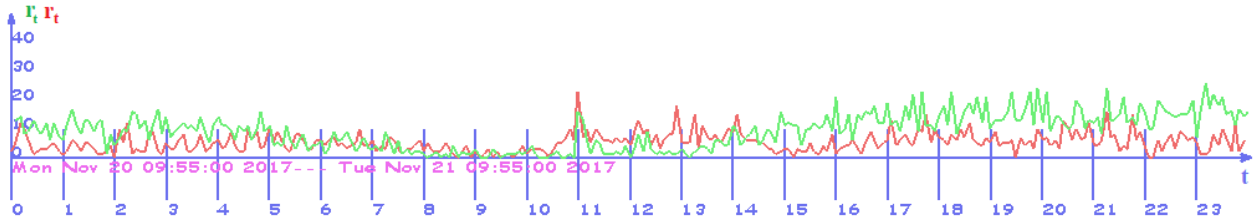


Рисунок 1.4 — Интенсивность трафика на перекрестке в течение 24 часов, зеленый цвет соответствует фазе $s^{(0)}$, красный — $s^{(1)}$

Решение задачи находится в три этапа, изображенных на рисунке 1.6.

Этап 1. Обработка данных об интенсивности трафика через перекресток и получение политик $\pi(a \mid s)$.

Этап 2. Создание массивов $r_{a^{(k)}}$, $k = 0, 1$.

Этап 3. Моделирование процесса выбора решения с подсчетом функции V.

Вычислительный эксперимент проводился с учётом условий:

– коэффициент переоценки $\gamma = 0.9$;

– задан целочисленный массив со средним количеством машин на участке дорожной сети в течение 24 часов с частотой дискретизации 300 секунд;

– задан критерий останова $t \leqslant T$ или $|V(t + \Delta t) - V(t)| < \varepsilon$, где $T = 86400$ секунд, $\varepsilon = e^{-308}$;

– шаг итерации Δt равен 4 секундам;

– управление светофорным объектом задается вручную в соответствии с политикой агента.

Пример политики, используемой при решении задачи представлен на рисунке 1.5.

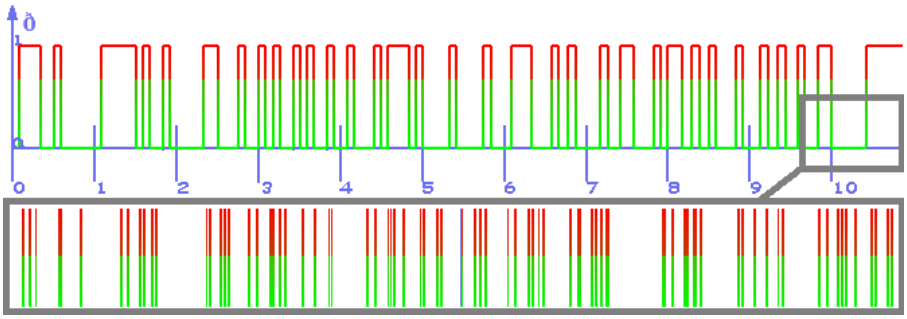


Рисунок 1.5 — График значений функции управления $\delta(s_t)$, зеленым цветом выделено действие $a^{(0)}$, красным — $a^{(1)}$

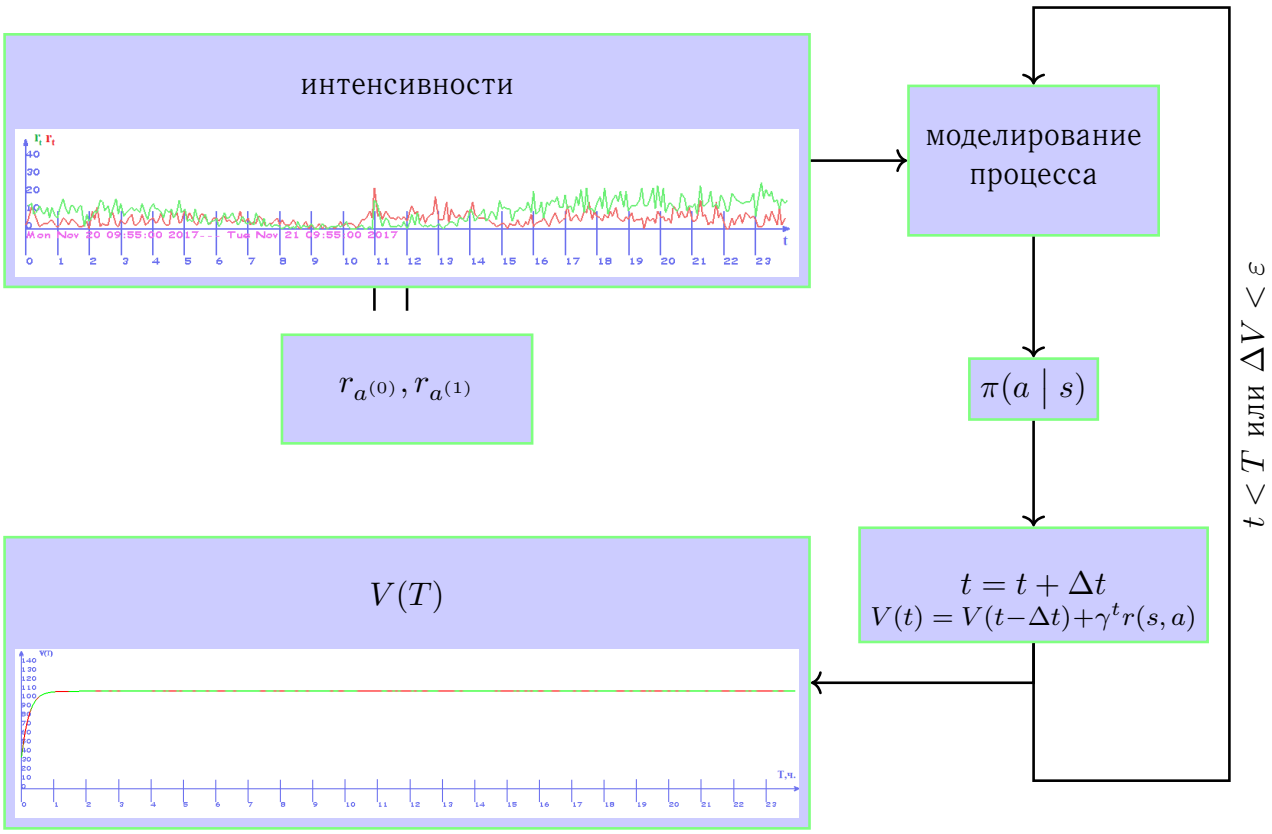


Рисунок 1.6 — Схема поиска решения прямой задачи

В процессе эксперимента при управлении δ из массивов $r_{a^{(0)}}$ и $r_{a^{(1)}}$ считывается с шагом Δt информация о вознаграждении. Далее на шаге t, согласно управлению δ и формуле перехода $s_{t+1} = \delta(s_t) \oplus s_t$, производится смена активной фазы и к переменной-счетчику V прибавляется вознаграждение $r(\delta(s_t), s_t)$, домноженное на коэффициент γ^t. Ниже на рисунке 1.7 приводится график функции $V(T)$ в масштабе тысячи единиц на час. Интересно отметить, что функция V стабилизируется, и подтверждается предложение 1.2.1 о ее ограниченности.

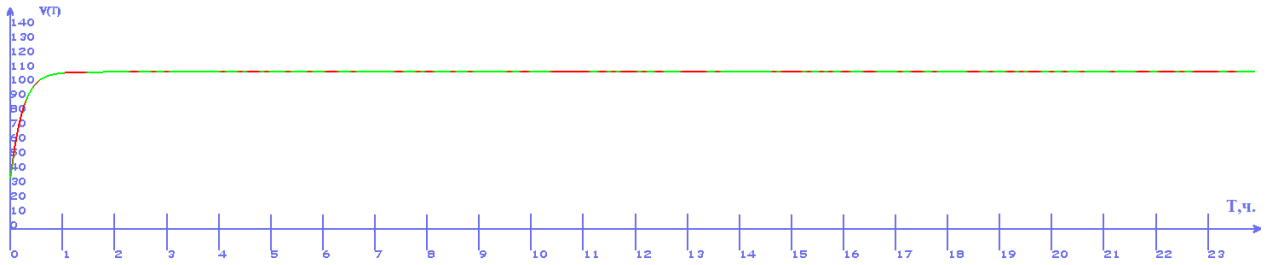


Рисунок 1.7 — График значений целевой функции, зеленым цветом выделена фаза $s^{(0)}$, красным — $s^{(1)}$

1.6.2 Задача поиска оптимального управления для двухфазного светофорного объекта

В мультиагентном подходе актуальны задачи нахождения результатов взаимодействия агентов со средой. Одним из возможных подходов к решению этих задач является имитационное моделирование.

Для исследования представленной модели была разработана программа имитационного моделирования в системе AnyLogic 8 Personal Learning Edition 8.7.12 на языке программирования Java 15.0.2 (см. Приложение А) и проведены серии вычислительных экспериментов.

Целью экспериментов было сравнение времени задержки машин для моделей системы управления светофоров, длительность фаз которой получена перебором и управляемой марковским процессом. Здесь $p0$ — длительность фазы $s^{(0)}$, $p1$ — длительность фазы $s^{(1)}$. Первая модель изображена на рисунке 1.8, вторая — на рисунке 1.9.

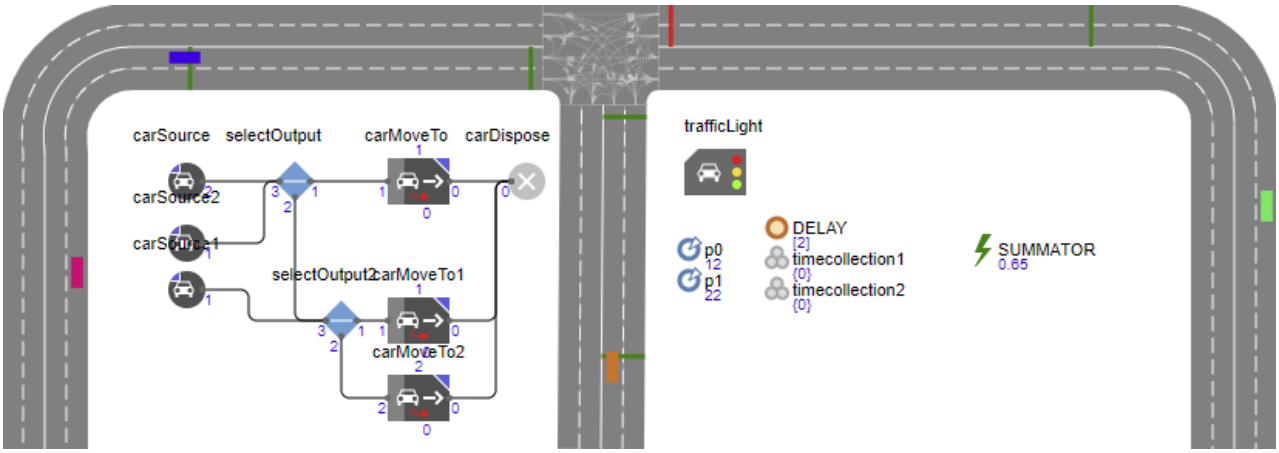


Рисунок 1.8 — Визуализация модели, длительность фаз которой получена перебором, в среде AnyLogic

Имитационное моделирование процесса управления светофором проводилось с учётом следующих условий.

– машины прибывают на перекресток с каждого из трех направлений: $carSource$, $carSource1$, $carSource2$, с интенсивностью 1000 $\frac{\text{машин}}{\text{час}}$;
– коэффициенты скидки α и переоценки γ подобраны эмпирически;
– дискретное время $period$ составляет 5 секунд.

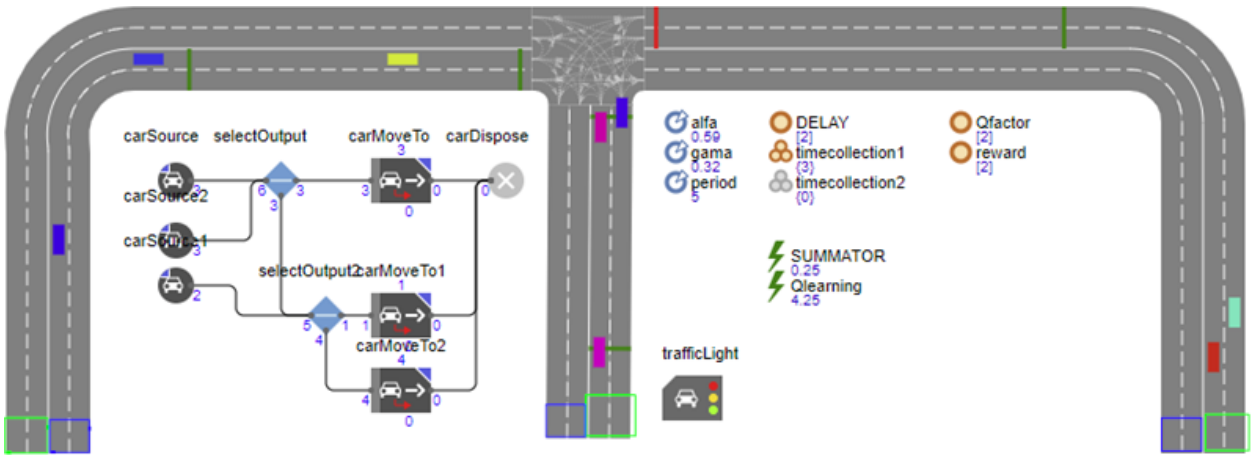


Рисунок 1.9 — Визуализация модели, управляемой марковским процессом, в среде AnyLogic

Переменные и процедуры, далее по тексту именуемые $timecollection$, $SUMMATOR$, $Qlearning$, $DELAY$, $trafficLight$, $alfa$, $gamma$, $period$ указаны на рисунках 1.8 и 1.9. Также на рисунке 1.9 отмечены эмпирически подобранные коэффициенты скидки γ и переоценки α. Длительность фаз $p0$ и $p1$ изображена на рисунке 1.8. При имитационном моделировании машины создаются на одной из позиций, отмеченных зеленым цветом, и перемещаются в направлении позиций, отмеченных синим, после чего удаляются. Параметры моделируемых машин задаются блоками $carSource$. Блок $carMoveTo$ содержит программную часть, ответственную за их пункт назначения. При пересечении полосы, на расстоянии 100 метров до стоп-линии, пары, состоящие из указателей на объект машины и текущего времени модели, добавляются в одну из коллекций $timecollection$, $timecollection1$, $timecollection2$. Далее, машины удаляются из коллекции, при проезде через перекресток. Каждую секунду модельного времени вызывается процедура $SUMMATOR$, прибавляющая суммарное время задержки машин, находящихся в коллекции, к текущей задержке $DELAY$. В течении периода времени $period$ вызывается событие $Qlearning$, реализующее решение задачи MARL. Псевдокод алгоритма $Qlearning$ представлен в приложении А. На основе события $Qlearning$ принимается решение об остановке или продлении фазы светофора $trafficLight$. Диаграма процесса управления светофорным объектом представлена на рисунке 1.10.

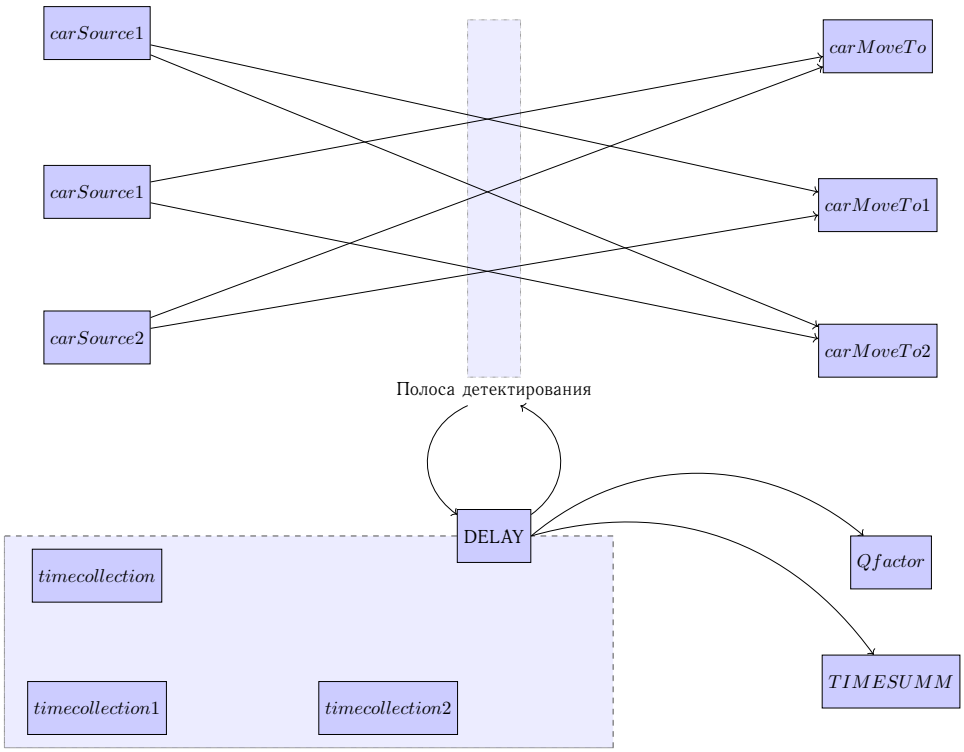


Рисунок 1.10 — Диаграмма процесса управления светофором

Модель, управляемая марковским процессом, при 300 независимых испытаниях показала уменьшение суммарной задержки в среднем в 1.5 раза по сравнению с системой управления светофором, длительность фаз которой подобрана перебором от 5 секунд до 30 секунд с шагом 1.

Интересно отметить, что вектор всевозможных значений функции $Q(s, a)$ стабилизируется, и подтверждается утверждение о его сходимости к неподвижной точке сжимающего отображения, построенного в предложении 1.5.2. График сходимости вектора всевозможных значений функции $Q(s_t, a_t)$ по метрике из $\mathbb{R}_{\infty}^{|\mathcal{S}|+|\mathcal{A}|}$ от времени t показан на рисунке 1.11.

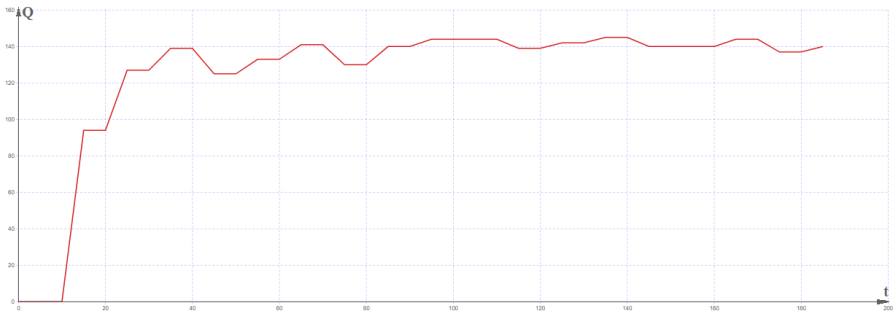


Рисунок 1.11 — График сходимости итерационного процесса для модели, управляемой марковским процессом

1.6.3 Задача поиска оптимального управления для трехфазного светофорного объекта

Целью экспериментов было сравнение времени задержки машин в среде AnyLogic для модели системы управления светофор, длительность фаз которой получена перебором, и управляемой марковским процессом. В качестве модели используется реальный перекресток города Красноярска пр. Свободный — ул. Годенко. Здесь $p0$ — длительность фазы $s^{(0)}$, $p1$ — длительность фазы $s^{(1)}$, $p2$ — длительность фазы $s^{(2)}$. В данной модели $s^{(0)}$ запускает движение по ул. Годенко, $s^{(1)}$ — движение по пр. Свободному в обоих направлениях, $s^{(2)}$ — движение по ул. Высотной. Первая модель изображена на рисунке 1.12, вторая — на 1.13. Имитационное моделирование процесса управления светофором проводилось с учётом следующих условий:

- машины прибывают на перекресток с интенсивностью 1000 $\frac{\text{машин}}{\text{час}}$ с каждого из четырех направлений: $carSource$, $carSource1$, $carSource2$, $carSource3$;
- коэффициенты скидки α и переоценки γ подобраны эмпирически;
- дискретное время $period$ составляет 4 секунды.

Переменные и процедуры, далее по тексту именуемые tcf, $SUMMATOR$, $Qlearning1$ и т.д., указаны на рисунках 1.12 и 1.13. При имитационном моделировании машины создаются на одной из позиций, отмеченных зеленым цветом, и перемещаются в направлении позиций, отмеченных синим, после чего удаляются. При пересечении полосы, на расстоянии 100 метров до стоплинии, пары, состоящие из указателей на объект машины и текущего времени модели, добавляются в одну из коллекций tcf, $tcf1$, $tcf2$, $tcf3$, $tcf4$ (time collection forward). Далее, машины удаляются из коллекции, при проезде через перекресток. Каждую секунду модельного времени вызывается процедура $SUMMATOR$, которая прибавляет количество машин, проехавших участок дорожной сети, к переменной $MCOUNT$ (machine count). Также событие $SUMMATOR$, прибавляющее суммарное время задержки машин, находящихся в коллекции, к текущей задержке $TIMESUMM$. В течении периода времени $period$ вызывается событие $Qlearning1$, реализующее решение зада-

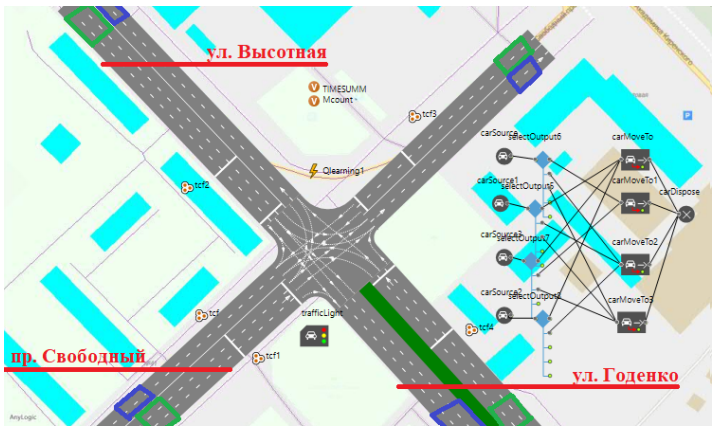


Рисунок 1.12 — Визуализация модели, управляемой марковским процессом, в среде AnyLogic

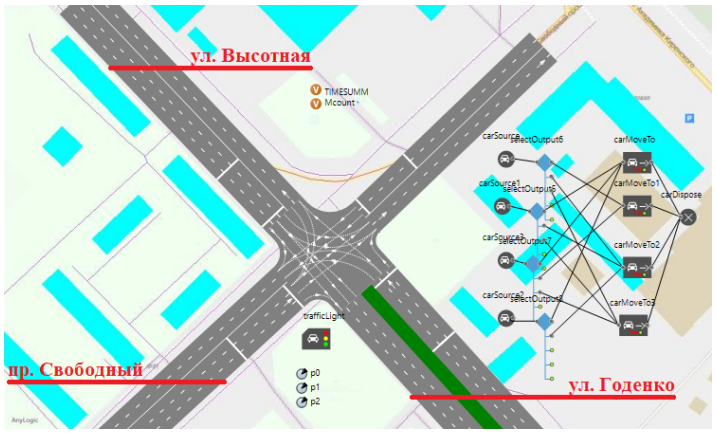


Рисунок 1.13 — Визуализация модели, длительность фаз которой получена перебором, в среде AnyLogic

чи MARL. На основе события $Qlearning1$ принимается решение об остановке или продлении фазы светофора $trafficLight$.

Модель, управляемая марковским процессом, показала уменьшение суммарной задержки в 1.2 раза по сравнению с системой управления светофором, длительность фаз которой подобрана перебором от 4 секунд до 32 секунд с шагом 4.

1.7 Выводы по главе 1

В данной главе поставлены и решены задача поиска оптимального управления для одного светофорного объекта и задача о вычислении функциии оценки эффективности управления.

Предложен алгоритм поиска приближенного решения задачи поиска опти-

мального управления для одного светофорного объекта. Для него сформулированны и доказаны критерий оптимальности Вальда–Беллмана 1.5.1, а также утверждение о виде приближенного решения 1.5.2.

Разработан комплекс программ и проведены серии вычислительных экспериментов для одного двухфазного, и одного трехфазного светофорного объекта. Произведена серия вычислительных экспериментов, в результате которой получено численное выражение того, насколько управляемая марковским процессом модель эффективнее справляется с пробками по сравнению с фиксированным планом смены фаз светофорных объектов.

Результаты главы 1 опубликованы в работе [10].

Глава 2. Сеть из двух светофорных объектов

Для нескольких агентов координация осуществляется при помощи рассмотрения стохастической игры [7]. В данной игре агенты в начальном состоянии получают вознаграждения, зависящие только от выбранных действий. Распределение вероятности переходов меняется и на следующем шаге оно зависит уже от предыдущего состояния и действий агентов в совокупности. Состояния и действия агентов, соответствующие такой задаче, графически изображены на рисунке 2.1а. Задача каждого агента выбрать действие, максимизирующее ожидаемую совокупную награду, учитывая возможный выбор действий соседнего. Схема координации агентов представлена на рисунке 2.1б. При конечном числе игроков, конечных множествах действий и состояний игра с конечным числом повторений всегда имеет равновесие Нэша [8]. Это справедливо также для игр с бесконечным числом повторений, если выигрыши участников представляют собой дисконтированную сумму.

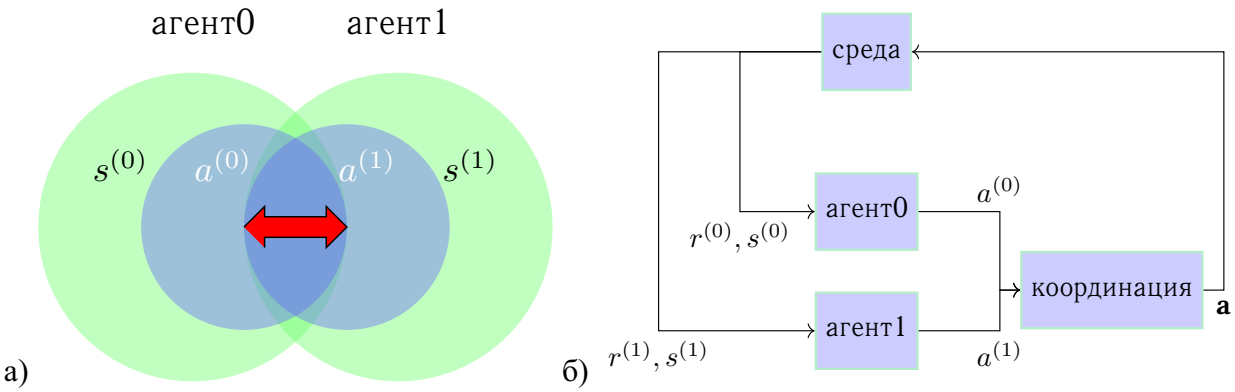


Рисунок 2.1 — а) Координированная задача MARL; б) Схема координации агентов

2.1 Описание задачи

Расширим задачу, рассмотренную в главе 1, на сеть из двух светофорных объектов. Множество из двух агентов, для каждого из которых описаны множество состояний $\mathcal{S}^0, \mathcal{S}^1$ и множество решений $\mathcal{A}^0, \mathcal{A}^1$. Обозначим $\mathbf{s}_t = \{s_t^1, s_t^2\} \in \mathcal{S}^0 \times \mathcal{S}^1$ — совокупное состояние среды в момент времени t, а $\mathbf{a}_t = \{a_t^0, a_t^1\} \in \mathcal{A}^0 \times \mathcal{A}^1$ — совокупное управление в момент времени t.

Отметим, что смена фазы любым агентом приводит к изменению общего

состояния среды **s**. В таблице 2.1 представлены условные вероятности перехода из состояния **s** в состояние $\mathbf{s}'$ с учетом выбора действия **a**.

Таблица 2.1 — Распределение $p(\mathbf{s}' \mid \mathbf{s}, \mathbf{a})$

a	**s** \ $\mathbf{s}'$	$\mathbf{s}^{(0)}$	$\mathbf{s}^{(1)}$	$\mathbf{s}^{(2)}$	$\mathbf{s}^{(3)}$
$\mathbf{a}^{(0)}$	$\mathbf{s}^{(0)}$	1	0	0	0
	$\mathbf{s}^{(1)}$	0	1	0	0
	$\mathbf{s}^{(2)}$	0	0	1	0
	$\mathbf{s}^{(3)}$	0	0	0	1
$\mathbf{a}^{(1)}$	$\mathbf{s}^{(0)}$	0	1	0	0
	$\mathbf{s}^{(1)}$	0	0	1	0
	$\mathbf{s}^{(2)}$	0	0	0	1
	$\mathbf{s}^{(3)}$	1	0	0	0
$\mathbf{a}^{(2)}$	$\mathbf{s}^{(0)}$	0	0	1	0
	$\mathbf{s}^{(1)}$	0	0	0	1
	$\mathbf{s}^{(2)}$	1	0	0	0
	$\mathbf{s}^{(3)}$	0	1	0	0
$\mathbf{a}^{(3)}$	$\mathbf{s}^{(0)}$	0	0	0	1
	$\mathbf{s}^{(1)}$	1	0	0	0
	$\mathbf{s}^{(2)}$	0	1	0	0
	$\mathbf{s}^{(3)}$	0	0	1	0

Для случая, когда рассматривается сеть светофорных объектов с двумя состояниями, следующую фазу светофора можно найти с помощью операции «исключающего или»:

$$\mathbf{s}' = \mathbf{a} \oplus \mathbf{s}. \tag{2.1}$$

Также переходы между состояниями, соответствующие рассматриваемой цепи, можно графически изобразить в виде стохастического графа 2.2, где для простоты изложения используется нумерация в двоичной системе счисления. Нумерация задается формулой (2.1).

Введем обозначение двоичного литерала из языка программирования Java, согласно [15]. Для его записи перед номером ставится префикс $0b$ (binary). Например, число 2 в двоичной записи будет представлено как $0b10$. В случае для действия $\mathbf{a} = 2$ можно провести следующие рассуждения: с заменой системы счисления и сопоставлением вектору из $\mathcal{A}^0 \times \mathcal{A}^1$. Действия $a^0 = 1$ и $a^1 = 0$ однозначно задают совместное действие $\mathbf{a} = \left(a^0, a^1\right) = 0b10$.

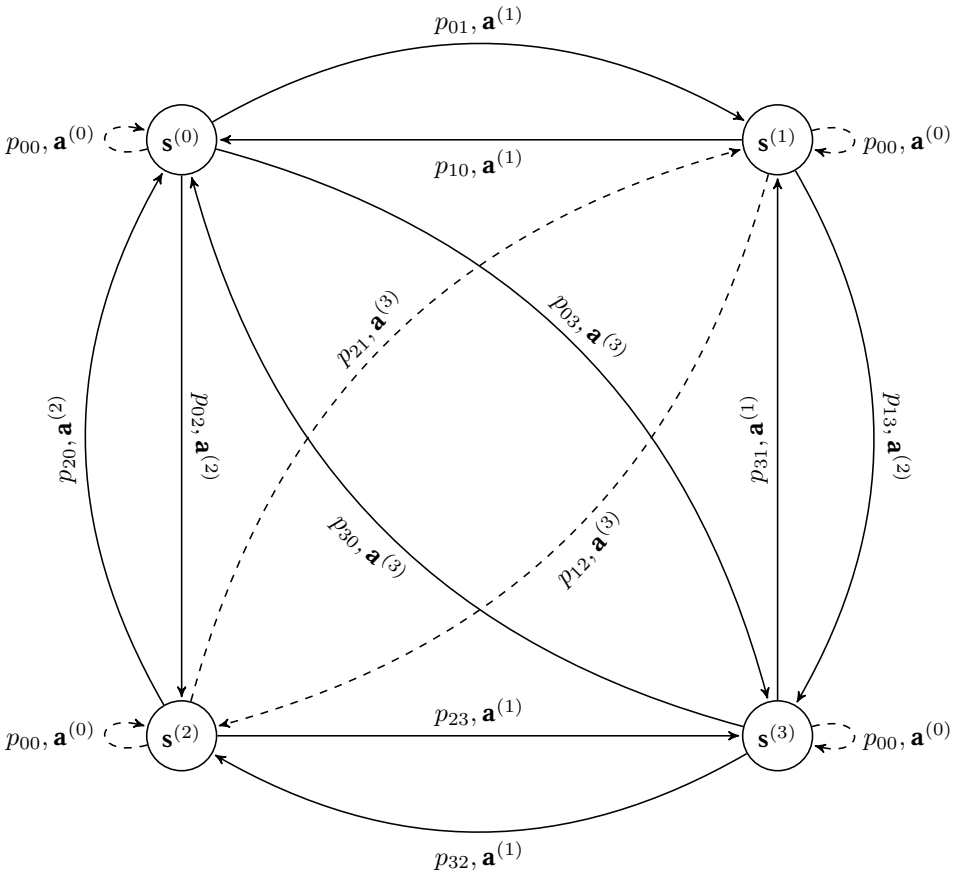


Рисунок 2.2 — Стохастический граф управляемого процесса смены фаз. Здесь s_l отражает совокупное состояние светофоров l (l =0b00, 0b01, 0b10, 0b11). При действии $\mathbf{a}^{(m)}$ (m =0b00, 0b01, 0b10, 0b11) процесс перейдет в состояние $n = l \oplus m$, с вероятностью p_{ln}.

В момент времени t функция наград $r(\mathbf{s}_t, \mathbf{a}_t)$ определяется как сумма функций наград каждого агента по отдельности $r(\mathbf{s}_t, \mathbf{a}_t) = r(s_t^0, a_t^0) + r(s_t^1, a_t^1)$, которые в свою очередь рассчитываются согласно формуле (1.11) из параграфа 1.4. В таком случае функция оценки эффективности управления примет вид

$$V(\mathbf{s}) = \mathbb{E} \sum_{t=0}^{\infty} \gamma^t r(\mathbf{s}_t, \mathbf{a}_t), \tag{2.2}$$

где $0 \leq \gamma \leq 1$ — коэффициент переоценки, $\mathbb{E}$ — оператор математического ожидания.

Управление фазами двух светофорных объектов находится как решение задачи обучения с подкреплением (Multiagent Reinforcement Learning for Integrated Network, MARLIN) и формулируется следующим образом:

Заданы: марковский процесс принятия решения $\left\langle \mathcal{S}^0 \times \mathcal{S}^1, \mathcal{A}^0 \times \mathcal{A}^1, \mathbb{P}, r \right\rangle$ для управления дорожной сетью из двух светофорных объектов,
активные в начальный момент времени фазы светофорных объектов.

Требуется найти: управление светофорными объектами $\delta^* = \{\mathbf{a}_t^*\}_{0 \le t < \infty}$, которое доставит максимум функции оценки его эффективности (2.2).

2.2 Описание механизма координации

Решение задачи поиска оптимального совокупного управления светофорными объектами дорожной сети ищется методом динамического программирования согласно принципу оптимальности Вальда—Беллмана. Функцию суммарных вознаграждений при оптимальном управлении на шаге t можно переписать в следующем виде:

$$V^*\left(\{\mathbf{s}_{t'}, \delta\}_{t'=0}^{t'=t}\right) = \max_{\mathbf{a} \in \mathcal{A}} Q_t(\mathbf{s}_t, \mathbf{a}),$$

где

$$Q_t(\mathbf{s}_t, \mathbf{a}) = \sum_{\mathbf{s}_{t+} \in \mathcal{S}} p(\mathbf{s}_{t+1} \mid \mathbf{s}_t, \mathbf{a}) \Big(r(\mathbf{s}_{t+1} \mid \mathbf{s}_t, \mathbf{a}) + \gamma \max_{\mathbf{a}' \in \mathcal{A}} Q_{t-1}(\mathbf{s}_{t+1}, \mathbf{a}') \Big).$$

Для совместных действий и состояний итерационно заданная функция Q_t имеет вид:

$$Q_t(\mathbf{s}, \mathbf{a}) = \sum_{t'=t}^{t-1} \gamma^{t'} r(\mathbf{s}_{t'}, \mathbf{a}_{t'}) + \gamma \sum_{\mathbf{s}'} p(\mathbf{s}' \mid \mathbf{s}, \mathbf{a}) V^* \left(\{\mathbf{s}_t\}, \delta \setminus \{\mathbf{a}_0 \dots \mathbf{a}_{t-1}\} \right).$$

Запись $Q_t(\mathbf{s}, a, a') = Q_t(\mathbf{s}, \mathbf{a})$ верна для случая двух агентов.

Сведем задачу поиска максимизирующего совместного действия нескольких агентов к случаю для одного агента k, совершающего действие a. Функция Q для агента k определяется следующим образом

$$\begin{aligned} Q_t^k(\mathbf{s}, a) &= \sum_{a' \in \mathcal{A}^j} p(a' \mid \mathbf{s}) Q_t(\mathbf{s}, a, a') = \\ &= \sum_{t'=0}^{t-1} \gamma^{t'} r(\mathbf{s}_{t'}, \mathbf{a}_{t'}) + \gamma \sum_{\mathbf{s}'} p(\mathbf{s}' \mid \mathbf{s}, a) \max_{\mathbf{a}'} Q_{t-1}(\mathbf{s}', \mathbf{a}'). \end{aligned}$$

Оптимальное управление для фиксированного агента k будем искать как решение задачи MARL в виде

$$a_t = \arg\max_{\mathbf{a}^k \in \mathcal{A}^k} \sum_{\mathbf{a}^j \in \mathcal{A}^j} Q_t^k(\mathbf{s}, \mathbf{a}^{kj}) p(\mathbf{s}' \mid \mathbf{s}, \mathbf{a}^{kj}). \tag{2.3}$$

Вероятность $p\left(\mathbf{s}' \mid \mathbf{s}, \mathbf{a}^{kj}\right)$ — это вероятность того, что агент j выберет действие $\mathbf{a}^j$ с учётом текущего совместного состояния $\mathbf{s}$ и выбранного агентом k действия $\mathbf{a}^k$.

Решение задачи поиска оптимального совокупного управления светофорными объектами дорожной сети ищется таким образом, чтобы увеличить максимальное совокупное вознаграждение, путем разбиения задачи на подзадачи одного агента. Для каждого агента по отдельности верны утверждения 1.5.1, 1.5.2 из главы 1, а следовательно, они верны и для нескольких координированных агентов.

Таким образом, задача совокупного управления несколькими агентами сводится к задаче управления одним агентом.

2.3 Вычислительные эксперименты

Решение задачи совокупного управления несколькими светофорными объектами было получено в виде (2.3) , где вероятность $p(\mathbf{s}' \mid \mathbf{s}, \mathbf{a}^{kj})$ находится как статистическая вероятность.

Для исследования представленной математической модели была разработана программа имитационного моделирования в системе AnyLogic (см. Приложение А) и проведены серии вычислительных экспериментов. Эксперименты проводились на ПК с процессором Intel Core i7-10510U CPU @ 1.80ГГц и оперативной памятью объемом 8ГБ. Целью экспериментов было сравнение времени задержки машин для двух моделей. Для первой модели длительность фаз (TrafficLight state) для первого (TLs0, TLs1) и второго светофора (TL1s0, TL1s1) получена перебором. Для второй модели длительность фаз получена как решение задачи MARL на основе обучения с подкреплением.

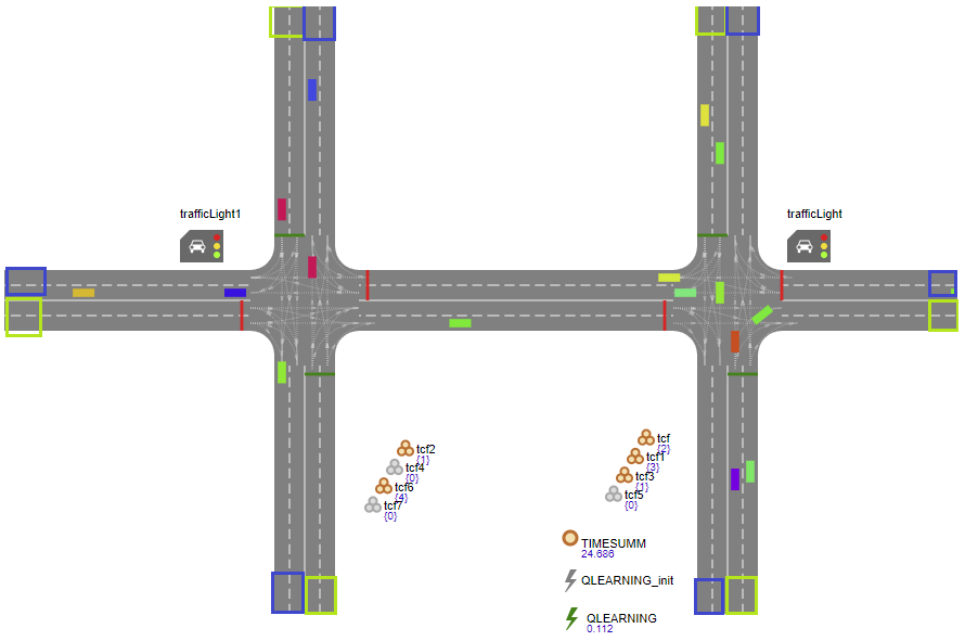


Рисунок 2.3 — Визуализациясети из двух двухфазных светофоров в среде AnyLogic

На рисунке 2.3 представлена модель дорожной сети из двух двухфазных светофорных объектов в среде AnyLogic. Моделирование процесса управления светофором проводилось с учётом следующих условий:

– машины прибывают на каждый из перекрёстков с трёх направлений, отмеченных зеленым цветом, с интенсивностью 1000 в час и удаляются в областях, отмеченных синим;
– направления движения машин указаны белыми стрелками;
– в модели управляемой однородной марковской цепи с конечным числом действий и состояний коэффициенты скидки α и переоценки γ подобраны эмпирическим путём;
– дискретизация по времени равна четырем секундам.

При пересечении полосы, на расстоянии 100 метров до стоп-линии, пары, состоящие из указателей на объект машины и текущего времени модели, добавляются в одну из коллекций $tcf, \ldots, tcf7$ (time collection forward). Далее, машины удаляются из коллекции, при проезде через перекресток и разница времени системы и хранящегося в коллекции прибавляется к величине $TIMESUMM$. Для второй модели в течении 5 секунд вызывается собы-

тие $QLEARNING$, реализующее решение задачи MARLIN. На его основе принимается решение об остановке или продлении фаз светофоров. Модель, управляемая марковским процессом, показала уменьшение суммарной задержки в среднем в 1.2 раза по сравнению с системой управления светофором, длительность фаз которой подобрана перебором.

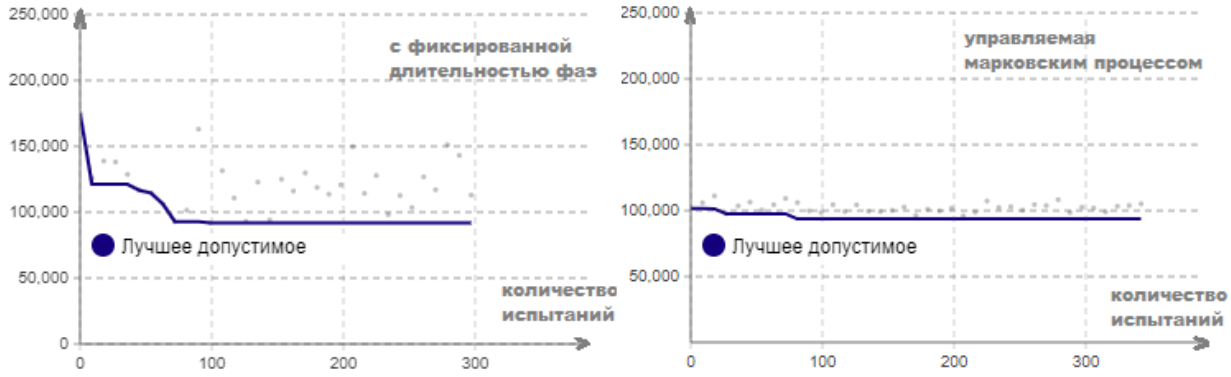


Рисунок 2.4 — Суммарное время, затраченное машинами на преодоление участка дорожной сети, справа на оси ординат показано значение для модели, управляемой марковским процессом, слева показано значение для модели перебора. На оси абсцисс указано количество экспериментов.

На рисунке 2.4 показано различие пропускной способности моделей при 300 независимых испытаниях. Синим цветом указано лучшее допустимое значение $TIMESUMM$. Серые точки показывают некоторые наиболее частые отклонения от лучшего значения.

2.4 Выводы по главе 2

В данной главе поставлена и решена задача поиска оптимального управления для дорожной сети из двух светофорных объектов.

Предложен алгоритм совокупного управления светофорными объектами, для которого в показано, что задача совокупного управления несколькими агентами сводится к задаче управления одним агентом. Разработан комплекс программ и проведены серии вычислительных экспериментов 2.3 для дорожной сети из двух двухфазных светофорных объектов. Основным результатом данной главы стало подтверждение эффективности применяемого управления.

Результаты главы 2 опубликованы в работах [9, 11].

Глава 3. Реальный участок дорожной сети

3.1 Описание задачи

В работе рассматривается участок дорожной сети г. Красноярска, состоящий из двух перекрестков: пр. Свободный — ул. Лесопарковая, пр. Свободный — ул. Высотная. Модель данной дорожной сети, реализованная в среде AnyLogic, изображена на рисунке 3.1а. Рядом с каждым светофорным объектом имеются оптические датчики, которые фиксируют проезд машин через зону дороги, отмеченную желтым цветом и находящуюся на расстоянии 70 метров до стоп-линии. На рисунке 3.1б представлен пример оптического датчика. Требуется уменьшить суммарное затраченное на проезд через перекрестки дорожной сети время.

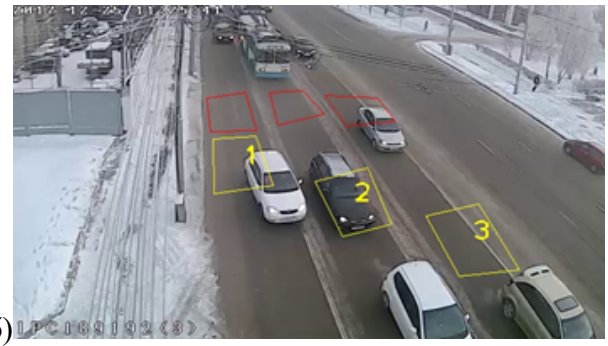
а)  б) 

Рисунок 3.1 — а) Модель рассматриваемого участка дорожной сети, реализованная в системе AnyLogic; б) зоны дороги (жёлтый цвет), фиксируемые оптическими датчиками

Расширим задачу обучения с подкреплением на сеть из двух светофорных объектов с разным количеством фаз. Множество из двух агентов, для которых описаны множество состояний $\mathcal{S}^0 = \left\{s^{(0)}, s^{(1)}\right\}$, $\mathcal{S}^2 = \left\{s^{(0)}, s^{(1)}, s^{(2)}\right\}$. Для перекрестка пр. Свободный — ул. Лесопарковая фаза $s^{(0)} \in \mathcal{S}^0$ активирует движение с ул. Лесопарковая на пр. Свободный, $s^{(1)} \in \mathcal{S}^0$ — движение вверх по пр. Свободному. Для перекрестка пр. Свободный — ул. Годенко фаза $s^{(0)} \in \mathcal{S}^1$ активирует движение по ул. Годенко, $s^{(1)} \in \mathcal{S}^1$ — движение по пр. Свободному в обоих направлениях, $s^{(2)} \in \mathcal{S}^1$ — движение по ул. Высотной. Множество действий для перекрестка пр. Свободный — ул. Лесопарковая $\mathcal{A}^0 = \left\{a^{(0)}, a^{(1)}\right\}$,

где $a^{(0)}$ интерпретируется как «оставить текущую фазу», $a^{(1)}$ — «сменить текущую фазу». Множество действий для перекрестка пр. Свободный — ул. Годенко $\mathcal{A}^1 = \left\{a^{(0)}, a^{(1)}, a^{(2)}\right\}$, где $a^{(k)}$ интерпретируется как «активировать фазу $\mathbf{s}'$», $\mathbf{s}'$ определяется по формуле (1.4). Обозначим $\mathbf{s}_t = \{s_t^0, s_t^1\} \in \mathcal{S}^0 \times \mathcal{S}^1$ — совокупное состояние среды в момент времени t, а $\mathbf{a}_t = \{a_t^0, a_t^1\} \in \mathcal{A}^0 \times \mathcal{A}^1$ — совокупное управление в момент времени t. Отметим, что смена фазы любым агентом приводит к изменению общего состояния среды $\mathbf{s}$. Также переходы между состояниями, соответствующие рассматриваемой цепи, можно графически изобразить в виде стохастического графа 3.2.

Для данного случая, когда рассматриваются сеть светофорных объектов с несколькими состояниями, следующую фазу светофора можно найти с помощью операции «остаток от деления»: $\mathbf{s}' = (\mathbf{a} + \mathbf{s}) \mod 6$, где 6 определяется как мощность $\mathcal{S}^0 \times \mathcal{S}^1$. В таблице 3.1 представлены условные вероятности перехода из состояния $\mathbf{s}$ в состояние $\mathbf{s}'$ с учетом выбора действия $\mathbf{a}$.

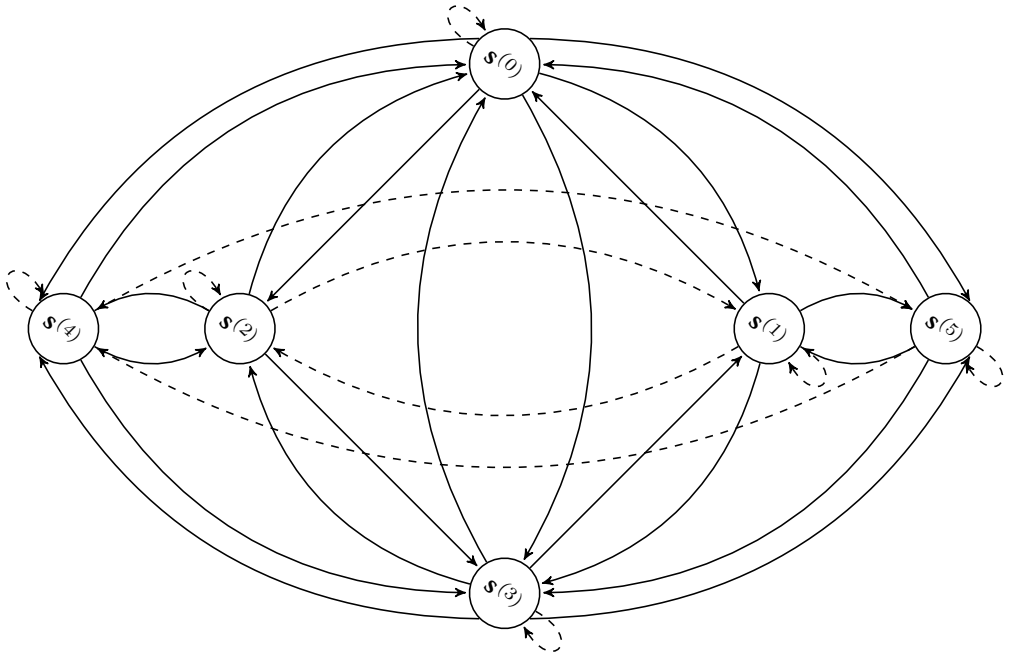


Рисунок 3.2 — Стохастический граф сети из двух многофазных светофорных объектов участка дорожной сети, состоящего из перекрестков: пр. Свободный — ул. Лесопарковая, пр. Свободный — ул. Высотная

Таблица 3.1 — Распределение $p(\mathbf{s}' \mid \mathbf{s}, \mathbf{a})$

$\mathbf{a}$	$\mathbf{s}$ \ $\mathbf{s}'$	$\mathbf{s}^{(0)}$	$\mathbf{s}^{(1)}$	$\mathbf{s}^{(2)}$	$\mathbf{s}^{(3)}$	$\mathbf{s}^{(4)}$	$\mathbf{s}^{(5)}$
$\mathbf{a}^{(0)}$	$\mathbf{s}^{(0)}$	1	0	0	0	0	0
	$\mathbf{s}^{(1)}$	0	1	0	0	0	0
	$\mathbf{s}^{(2)}$	0	0	1	0	0	0
	$\mathbf{s}^{(3)}$	0	0	0	1	0	0
	$\mathbf{s}^{(4)}$	0	0	0	0	1	0
	$\mathbf{s}^{(5)}$	0	0	0	0	0	1
$\mathbf{a}^{(1)}$	$\mathbf{s}^{(0)}$	0	1	0	0	0	0
	$\mathbf{s}^{(1)}$	0	0	1	0	0	0
	$\mathbf{s}^{(2)}$	0	0	0	1	0	0
	$\mathbf{s}^{(3)}$	0	0	0	0	1	0
	$\mathbf{s}^{(4)}$	0	0	0	0	0	1
	$\mathbf{s}^{(5)}$	1	0	0	0	0	0
$\mathbf{a}^{(2)}$	$\mathbf{s}^{(0)}$	0	0	1	0	0	0
	$\mathbf{s}^{(1)}$	0	0	0	1	0	0
	$\mathbf{s}^{(2)}$	0	0	0	0	1	0
	$\mathbf{s}^{(3)}$	0	0	0	0	0	1
	$\mathbf{s}^{(4)}$	1	0	0	0	0	0
	$\mathbf{s}^{(5)}$	0	1	0	0	0	0

$\mathbf{a}$	$\mathbf{s}$ \ $\mathbf{s}'$	$\mathbf{s}^{(0)}$	$\mathbf{s}^{(1)}$	$\mathbf{s}^{(2)}$	$\mathbf{s}^{(3)}$	$\mathbf{s}^{(4)}$	$\mathbf{s}^{(5)}$
$\mathbf{a}^{(3)}$	$\mathbf{s}^{(0)}$	0	0	0	1	0	0
	$\mathbf{s}^{(1)}$	0	0	0	0	1	0
	$\mathbf{s}^{(2)}$	0	0	0	0	0	1
	$\mathbf{s}^{(3)}$	1	0	0	0	0	0
	$\mathbf{s}^{(4)}$	0	1	0	0	0	0
	$\mathbf{s}^{(5)}$	0	0	1	0	0	0
$\mathbf{a}^{(4)}$	$\mathbf{s}^{(0)}$	0	0	0	0	1	0
	$\mathbf{s}^{(1)}$	0	0	0	0	0	1
	$\mathbf{s}^{(2)}$	1	0	0	0	0	0
	$\mathbf{s}^{(3)}$	0	1	0	0	0	0
	$\mathbf{s}^{(4)}$	0	0	1	0	0	0
	$\mathbf{s}^{(5)}$	0	0	0	1	0	0
$\mathbf{a}^{(5)}$	$\mathbf{s}^{(0)}$	0	0	0	0	0	1
	$\mathbf{s}^{(1)}$	1	0	0	0	0	0
	$\mathbf{s}^{(2)}$	0	1	0	0	0	0
	$\mathbf{s}^{(3)}$	0	0	1	0	0	0
	$\mathbf{s}^{(4)}$	0	0	0	1	0	0
	$\mathbf{s}^{(5)}$	0	0	0	0	1	0

Функция суммарных вознаграждений данной задачи принимает вид

$$V(\mathbf{s}) = \mathbb{E} \sum_{t=0}^{\infty} \gamma^t r(\mathbf{s}_t, \mathbf{a}_t), \tag{3.1}$$

где $0 \leq \gamma \leq 1$ — коэффициент переоценки.

Управление фазами двух светофорных объектов находится при решении задачи обучения с подкреплением (Multiagent Reinforcement Learning for Integrated Network, MARLIN) и формулируется следующим образом:

Заданы: марковский процесс принятия решения $\left\langle \mathcal{S}^0 \times \mathcal{S}^1, \mathcal{A}^0 \times \mathcal{A}^1, \mathbb{P}, r \right\rangle$ для управления дорожной сетью из двух светофорных объектов,
активные в начальный момент времени фазы светофорных объектов.

Требуется найти: управление светофорными объектами $\delta^* = \{\mathbf{a}_t^*\}_{0 \leq t < \infty}$, которое доставит максимум функции оценки его эффективности (3.1).

Рассуждения, приведенные в главе 2, применимы и к сети, состоящей из двух многофазных светофорных объектов. В таком случае оптимальное управление для фиксированного агента k будем искать как решение задачи мультиагентного обучения с подкреплением с условием координации агентов MARLIN в виде

$$a_t = \arg\max_{a^k \in \mathcal{A}^k} \sum_{a^j \in \mathcal{A}^j} Q_t^k(\mathbf{s}, \mathbf{a}^{kj}) p(\mathbf{s}' \mid \mathbf{s}, \mathbf{a}^{kj}). \tag{3.2}$$

Вероятность $p\left(\mathbf{s}' \mid \mathbf{s}, \mathbf{a}^{kj}\right)$ — это вероятность того, что агент j выберет действие a^j с учётом текущего совместного состояния $\mathbf{s}$ и выбранного агентом k действия a^k.

3.2 Вычислительные эксперименты

Для исследования представленной модели была разработана программа имитационного моделирования в системе AnyLogic и проведены серии вычислительных экспериментов. Эксперименты проводились на ПК с процессором Intel Core i7-10510U CPU @ 1.80ГГц и оперативной памятью объемом 8ГБ. Целью экспериментов было сравнение результатов имитационного моделирования для моделей FIXED, MARL, MARLIN.

В модели с фиксированнной длительностью фаз FIXED длительность фаз ($TrafficLight\ state$) для первого ($TLs0$, $TLs1$) и второго светофора ($TL1s0$, $TL1s1$) получена перебором всех возможных длин с шагом 4. Для модели MARL с марковским процессом при отсутствии координации агентов управление получено как решение задачи обучения с подкреплением для каждого светофора в отдельности (см. главу 1). Управление модели с координированным марковским процессом MARLIN получено как решение задачи задачи обучения с подкреплением с условием координации светофоров в сети (см. главу 2).

Для решения поставленной задачи были собраны данные об интенсивности трафика за 2017 — 2019 и вычислены среднесуточные интенсивности для рабочих дней.

Этапы подготовки статистических данных:

1. Сбор статистических данных с оптических датчиков дорожного контроля, представленный на рисунке 3.3;
2. Выгрузка из базы данных «Ар Ди Сайнс» файла **year_data.csv**, содержащего годовые интенсивности на различных светофорных объектах;
3. Структурирование данных файла, согласно соответствию устройства связи с дорожным контроллером зоне сбора статистики;
4. Подсчет среднесуточной интенсивности трафика с учетом выходных и нерабочих дней.

Рисунок 3.3 — Зоны сбора статистики

Результат обработки файла с данными указан на рисунке 3.4. В конечном итоге было получено 15 файлов с записями о средней интенсивности трафика для соответствующих устройства связи с дорожным контролем и зоны сбора статистики.

Имитационное моделирование процесса управления светофором проводилось с учётом следующих условий:

– машины прибывают на перекресток с каждой полосы из пяти рассматриваемых направлений;
– интенсивность прибытия машин получена из файла **year_data.csv**;
– коэффициенты скидки α и переоценки γ подобраны эмпирически;
– дискретное время $period$ составляет 8 секунд.

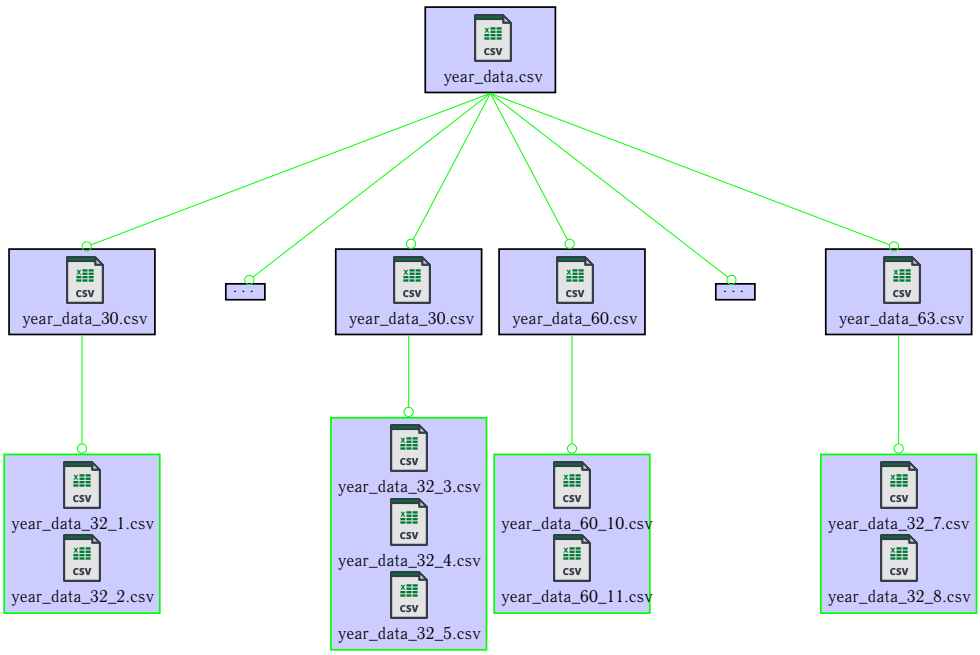


Рисунок 3.4 — Парсинг для файла year_data.csv

Модель, управляемая марковским процессом, показала уменьшение суммарной задержки в 1.5 раза по сравнению с системой управления светофором, длительность фаз которой подобрана перебором. Сравнение показателей эффективности совокупного управления для различных моделей представлено таблице 3.2 и рисунке 3.5.

Таблица 3.2 — Сравнение показателей эффективности совокупного управления для различных моделей

Целевая функция	Ед. изм.	FIXED	MARL	Прирост	MARLIN	Прирост	MARL vs. FIXED	MARLIN vs. FIXED
Средняя задержка	сек./маш.	137.3	45	92.3	44.9	92.4	67.2%	67.3%
Пропускная способность	маш.	1550	2935	1385	3030	1480	47.2%	48%
Суммарная задержка	сек.	213 079	132 367	80712	136 085	76994	37.9 %	36 %

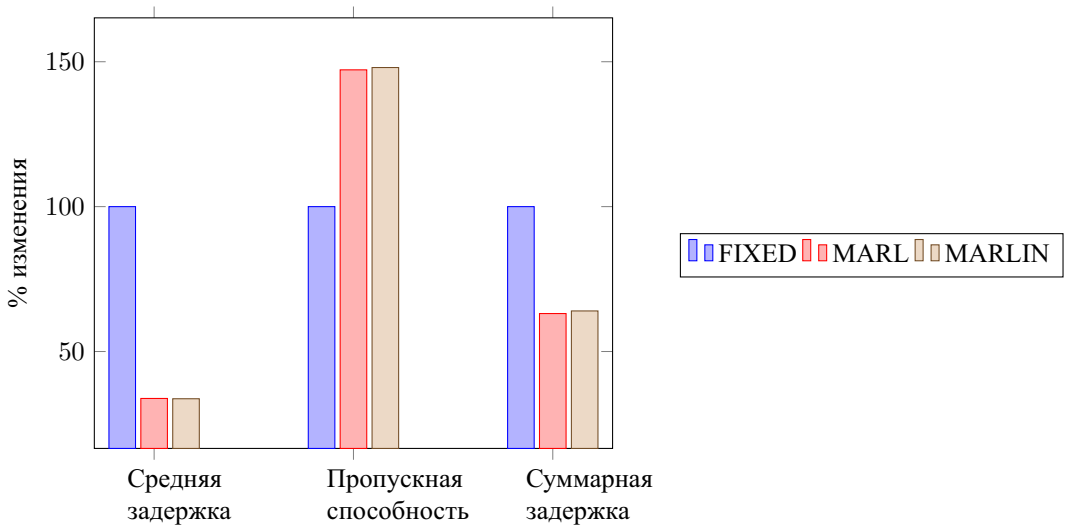


Рисунок 3.5 — Сравнение показателей эффективности совокупного управления для различных моделей

Блок-схема алгоритма поиска решения рассматриваемой задачи представлена на рисунке 3.6. При имитационном моделировании в результате вызова процедуры $time\_collections$ на позициях, обозначенных как $CarSource$, создаются машины в количествах, приближенных к реальным значениям. Далее автомобили перемещаются в дорожной сети пока не выйдут из ее зоны покрытия. При пересечении полосы, на расстоянии 70 метров до стоп-линии, пары, состоящие из указателей на объект машины и текущего времени модели, добавляются в одну из коллекций tcf_улица_полоса (time collection forward). Далее, машины удаляются из коллекции, при проезде через перекресток. В течении периода времени 1 секунда вызывается событие $SUMMATOR$, прибавляющее суммарное время задержки машин, находящихся в коллекции, к текущей задержке $TIMESUMM$. Данное событие прибавляет количество машин, проехавших участок дорожной сети, к переменной $MCOUNT$ (machine count). В течении периода времени $period$ вызывается событие $Qlearning$ и $Qlearning\_init$, реализующее решение задачи MARLIN и выделяющее под вычисления память соответственно. Операция $Qfactor$ соответствует формуле подсчета функции $Q_t^k(\mathbf{s}, a)$, а максимизирующее действие находится по формуле (2.3). На основе события $Qlearning$ принимается решение об остановке или продлении фазы светофоров $trafficLight$ для пр. Свободный — ул.

Высотная. и $trafficLight1$ для пр. Свободный — ул. Лесопарковая.

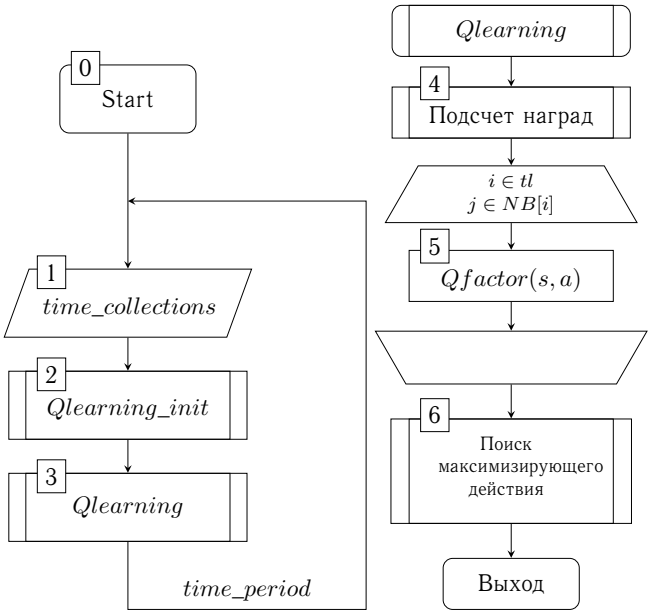


Рисунок 3.6 — Блок-схема алгоритма поиска решения

3.3 Выводы по главе 3

В данной главе поставлена и решена задача поиска оптимального управления для участка реальной дорожной сети, содержащей многофазные светофорные объекты. Разработан комплекс программ, связывающих показания оптического датчика и переменные моделируемой среды и проведены серии вычислительных экспериментов для реальной дорожной сети, содержащей многофазные светофорные объекты. Данный комплекс программ позволяет генерировать автомобили в модели в соответствии с реальной систуацией на дорогах. Основным результатом данной главы стало подтверждение эффективности применяемого управления для реальных участков дорожной сети.

Результаты главы 3 опубликованы в работе [9].

ЗАКЛЮЧЕНИЕ

В бакалаврской работе представлены модель, метод, алгоритмы и программы реализующие мультиагентную систему для задачи совокупного управления светофорными объектами участков реальной дорожной сети.

Основные результаты диссертационной работы представлены ниже.

1. Построены математические модели процесса управления для одного и для сети светофорных объектов, отличающиеся учетом текущего расположения светофорных объектов и их загруженности и позволяющие сформулировать оптимизационные задачи, целью которых является минимизация задержки трафика автомобилей.
2. Разработан и теоретически обоснован алгоритм управления участком дорожной сети для одного двухфазного светофорного объекта.
3. Разработан и теоретически обоснован алгоритм управления участком дорожной сети для двух двухфазных светофорных объектов.
4. Разработан и теоретически обоснован алгоритм управления участком реальной дорожной сети из двух многофазных светофорных объектов.
5. Создан комплекс программ, реализующий разработанные алгоритмы, для проверки их результативности применительно к дорожным сетям.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Биекция [Электронный ресурс]: Википедия. Свободная энциклопедия.— Режим доступа: https://ru.wikipedia.org/wiki/Биекция (дата обращения: 03.04.2022).
2. Венцель, Е. С. Исследование операций: Задачи, принципы, методология / Е. С. Венцель — М. : Дрофа, 2004. — 285 с.
3. Виноградов, И. М. Основы теории чисел / И. М. Виноградов — Регулярная и хаотическая динамика, 2003. — 178 с.
4. Гасников, А. В. Лекции по случайным процессам : учебное пособие / А. В. Гасников, Э. А. Горбунов, и др. — М.: Москва : МФТИ, 2019. — 208 с.
5. Иванов, С. Конспект по обучению с подкреплением. / С. Иванов— Режим доступа: https://arxiv.org/abs/2201.09746 (дата обращения: 03.04.2022).
6. Колмогоров, А. Н. Элементы теории функций и функционального анализа. / А. Н. Колмогоров, С В. Фомин — 7-е изд. — М.: ФИЗМАТЛИТ, 2004. — 572 с.
7. Равновесие Нэша [Электронный ресурс]: Википедия. Свободная энциклопедия.— Режим доступа: hhttps://ru.wikipedia.org/wiki/равновесие_Нэша (дата обращения: 03.04.2022).
8. Стохастическая игра [Электронный ресурс]: Википедия. Свободная энциклопедия.— Режим доступа: https://ru.wikipedia.org/wiki/стохастическая_игра (дата обращения: 03.04.2022).
9. Тисленко, Т. И. Оптимизация планов координаций для светофорных объектов участка дорожной сети / Т. И. Тисленко, Д. В. Семенова, Н. А. Сергеева // Системы управления, информационные технологии и математическое моделирование : материалы IV Всероссийской научно-практической конференции с международным участием (Омск, 19 мая 2022 г.): в 2 т. / Минобрнауки России, Ом. гос. техн. ун-т, Каф. «ММиИТЭ» ; отв. ред. В. Н. Задорожный. — Омск : Изд-во ОмГТУ, 2022. — С. 255-261.

10. Тисленко, Т. И. Задача MARL для светофора на перекрестке / Т. И. Тисленко // Материалы VIII Международной молодежной научной конференции «Математическое и программное обеспечение информационных, технических и экономических систем». — 2021. — С. 144-149.
11. Тисленко, Т. И. Задача MARL для светофорной сети / Т. И. Тисленко, Д. В. Семенова // Информационные технологии и математическое моделирование (ИТММ-2021): Материалы XX Международной конференции имени А.Ф. Терпугова. — Томск: 2022 (принята в печать).
12. Carini, R. N. Application of the UTCS-1 Network Simulation Model to Select Optimal Signal Timings in a Multi-Linear Street System : Interim Report. Publication for Urban Traffic Control System. Joint Highway Research Project, Indiana Department of Transportation and Purdue University, West Lafayette, Indiana, 1977.
13. Chandler, M. J. H. Traffic Control Studies in London: SCOOT and Bus Detection. OTRC Proc. Annual Summer Meeting, University of Sussex, Vol. P269, P. 111-128. 1985.
14. Gartner, N.H. OPAC: Strategy for Demand-responsive Decentralized Traffic Signal Control, IFAC Proceedings Volumes, Volume 23, Issue 2, 1990, p. 241-244.
15. Java syntax [Электронный ресурс]: Википедия. Свободная энциклопедия.— URL: https://en.wikipedia.org/wiki/java_syntax (дата обращения: 03.04.2022).
16. Sutton, R. S. Reinforcement Learning: An Introduction / R. S. Sutton, A G. Barto — 1-st ed. — The MIT Press Cambridge, Massachusetts London, England, 2014. — 352 c.

ПРИЛОЖЕНИЕ А

Алгоритм А.1. Функция $Q - learning$

Вход: $\alpha, \gamma, s, timecollection1, timecollection2, Q1[s][a], Q1max[s]$.

Выход: s'

$double[][]T = newdouble[S][A];$

$double[][]Q2 = newdouble[S][A];$

Процедура Подсчет наград($timecollection1, timecollection2$; **Возвратить** $t, T[\,][\,]$)

$double[]T1 = newdouble[A];$

$doublet = time();$

Для $entry \in Map.Entry < Agent, Double >$ t$timecollection1$

$T[0] += t - entry.getValue()$

Конец цикла

Для $entry \in Map.Entry < Agent, Double >$ t$timecollection2$

$T[1] += t - entry.getValue()$

Конец цикла

$t = T1[0] + T1[1];$

Для $a \in A, s \in S$

$s' = (s + a)\%A;$

$T[s][a] = T1[s'];$

Конец цикла

Конец процедуры

Процедура Подсчет функции $\max\limits_{a \in A} Q(s, a)$(**Возвратить** q_max)

$double[]q\_max = newdouble[S];$

Для $a \in A, s \in S$

$s' = (s + a)\%S;$

$Q2[s][a] = Qfactor(T[s][a], Q1[s][a], alfa, gamma, Q1max[s']);$

Если $q\_max[s] < Q2[s][a]$ **то**

$q\_max[s] = Q2[s][a];$

Конец условия

Конец цикла

Конец процедуры

Для $a \in A, s \in S$

$s' = (s + a)\%S;$

$Q2[s][a] = Qfactor(T[s][a], Q1[s][a], alfa, gamma, Q1max[s']);$

Конец цикла

Процедура Поиск максимизирующего действия($Q2[s][a], q\_max$ **Возвратить** a)

Конец процедуры

$Q1 = Q2;$

$Q1max = q\_max;$

0

```
void main(String[] args) {
//TrafficLight definition%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    trafficLight = new TrafficLight<RoadLanesConnector>();
    trafficLight.phases = 6;
    trafficLight.curind = 3;

    trafficLight1 = new TrafficLight<RoadLanesConnector>();
    trafficLight1.phases = 4;
    trafficLight1.curind = 3;
//%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
//Controller collectors%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    tcf_SVOB_21_1 = new LinkedHashMap<Agent,Double>();
    tcf_SVOB_21_2 = new LinkedHashMap<Agent,Double>();
    tcf_GOD_2_1 = new LinkedHashMap<Agent,Double>();
    tcf_GOD_2_2 = new LinkedHashMap<Agent,Double>();
    tcf_GOD_2_3 = new LinkedHashMap<Agent,Double>();
    tcf_SVOB_1_1 = new LinkedHashMap<Agent,Double>();
    tcf_SVOB_1_2 = new LinkedHashMap<Agent,Double>();
    tcf_SVOB_1_3 = new LinkedHashMap<Agent,Double>();
    tcf_VISOT_1 = new LinkedHashMap<Agent,Double>();
    tcf_VISOT_2 = new LinkedHashMap<Agent,Double>();
    tcf_VISOT_3 = new LinkedHashMap<Agent,Double>();
    tcb_SVOB_1_1 = new LinkedHashMap<Agent,Double>();
    tcb_SVOB_1_2 = new LinkedHashMap<Agent,Double>();
    tcf_SVOB_0_1 = new LinkedHashMap<Agent,Double>();
    tcf_SVOB_0_2 = new LinkedHashMap<Agent,Double>();
    tcf_LESOP_1 = new LinkedHashMap<Agent,Double>();
    tcf_LESOP_2 = new LinkedHashMap<Agent,Double>();
//%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
//tl neighbors and actions%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    Object[] tl = new Object[2];
    NB = new int[][]{
        {1},
        {0}
    };
    a = new int[tl.length];
//\%%%%%%%%%%%%%%%%%%%%%%%%%%%
//CALL Qlearning process%%%%%
    QLEARNING_init();
    QLEARNING();
//%%%%%%%%%%%%%%%%%%%%%%%%%%%%
}
```

1

```
double t = 3600 * 4 + time();
// ЛЕСОПАРКОВАЯ
carSource4_0.rate = selectFrom(year_data_av_31_6)
                .where(year_data_av_31_6.time.goe(t))
                .list(year_data_av_31_6.data).get(0) * 20;

carSource4_1.rate = selectFrom(year_data_av_31_7)
                .where(year_data_av_31_7.time.goe(t))
                .list(year_data_av_31_7.data).get(0) * 20;

// СВОБОДНЫЙ
carSource3_0.rate = selectFrom(year_data_av_32_1)
                .where(year_data_av_32_1.time.goe(t))
                .list(year_data_av_32_1.data).get(0) * 20;

carSource3_1.rate = selectFrom(year_data_av_32_2)
                .where(year_data_av_32_2.time.goe(t))
                .list(year_data_av_32_2.data).get(0) * 20;

// СВОБОДНЫЙ
carSource1_0.rate = selectFrom(year_data_av_60_10)
                .where(year_data_av_60_10.time.goe(t))
                .list(year_data_av_60_10.data).get(0) * 20;

carSource1_1.rate = selectFrom(year_data_av_60_11)
                .where(year_data_av_60_11.time.goe(t))
                .list(year_data_av_60_11.data).get(0) * 20;

// ВЫСОТНАЯ
carSource_0.rate = selectFrom(year_data_av_61_4)
                .where(year_data_av_61_4.time.goe(t))
                .list(year_data_av_61_4.data).get(0) * 20;

carSource_1.rate = selectFrom(year_data_av_61_5)
                .where(year_data_av_61_5.time.goe(t))
                .list(year_data_av_61_5.data).get(0) * 20;

carSource_2.rate = selectFrom(year_data_av_61_6)
                .where(year_data_av_61_6.time.goe(t))
                .list(year_data_av_61_6.data).get(0) * 20;

// ГОДЕНКО
carSource2_0.rate = selectFrom(year_data_av_62_1)
                .where(year_data_av_62_1.time.goe(t))
                .list(year_data_av_62_1.data).get(0) * 20;

carSource2_1.rate = selectFrom(year_data_av_62_2)
                .where(year_data_av_62_2.time.goe(t))
                .list(year_data_av_62_2.data).get(0) * 20;

carSource2_2.rate = selectFrom(year_data_av_62_3)
                .where(year_data_av_62_3.time.goe(t))
                .list(year_data_av_62_3.data).get(0) * 20;
```

2

```
    //tabular calculation units%%
    M = new double[][][][];
    Q = new double[][][][];
    Q2 = new double[][][][];
    MAXQ = new double[][][];
//%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

3

```
//Traffic Light i %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for (int i = 0; i < tl.length; i++) {
    // Traffic light j %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    for (int j = 0; j <  NB[i].length; j++) {
        // ОБНОВЛЯЕМ M[i][j] %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        M[i][j][si][aj]++;
        M[i][j][si][Aj]++;
        //%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        // maximizing action for Q_ij(s,a) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        for(_s = 0; _s < Si*Sj; s++)//      _s is coordinated i, NB[i][j]
            MAXQ[i][j][_s] = FindArrMax(Q[i][j][_s]);
        //%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        // recalculate Q-function %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        for(_s = 0; _s < Si*Sj; s++)//      _s is coordinated i, NB[i][j]
        for(_a = 0; _a < Ai*Aj; a++)//      _a is coordinated i, NB[i][j]
        Q2[i][j][_s][_a] = Qfunction(
                                Q[i][j][_s][_a],
                                r[i][_si][_ai] + r[NB[i][j]][_sj][_aj],
                                MAXQ[i][j][_sn]     );
        //%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    }
    double[] SUMQ_j_aj = new double[Ai];
    for(int _ai = 0; _ai < Ai; _ai++) {
    // chose ai for tl[i] %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    for (int j = 0; j < _j; j++) {
        // summ for all j and aj  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        SUMQ_j_aj[_ai] = 0;
        _s = si * Sj + sj;//                 _s is coordinated i, NB[i][j]
        for (int _aj = 0; _aj < Aj; _aj++) { // for all aj for NB[i][j] tl
            _a = _ai * Aj + _aj;//           _a is coordinated i, NB[i][j]
            SUMQ_j_aj[_ai] += Q2[i][j][_s][_a] * M[i][j][_s][_aj];
        }
    }  }
    //%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    // find maximizing action for i %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    a2[i] = FindArrMaxInd(SUMQ_j_aj);// argmax_of_SUMQ_j_aj
    //%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    }

//%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

4

```
//%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
double[][][] r = {
    CountTimeonTL(
        time(),
        new Object[][]{
            {tcf_SVOB_1_3, tcf_GOD_2_1, tcf_GOD_2_2,
             tcf_GOD_2_3},
            {tcf_SVOB_1_1, tcf_SVOB_1_2, tcf_SVOB_1_3,
             tcf_SVOB_21_1, tcf_SVOB_21_2},
            {tcf_VISOT_1, tcf_VISOT_2, tcf_VISOT_3}    }
    ),
    CountTimeonTL(
        time(),
        new Object[][]{
            {tcf_SVOB_0_1, tcf_SVOB_0_2, tcf_LESOP_1,
             tcf_LESOP_2},
            {tcb_SVOB_1_1, tcb_SVOB_1_2}    }
    )  };
//%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Рисунок А.2 — Расширенная блок-схема

Министерство науки и высшего образования РФ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт математики и фундаментальной информатики
Кафедра высшей и прикладной математики

УТВЕРЖДАЮ
Заведующий кафедрой
С.Г. Мысливец
«19» июня 2022 г.

БАКАЛАВРСКАЯ РАБОТА

Направление 01.03.02 Прикладная математика и информатика

ЗАДАЧА MARL ДЛЯ ОПТИМИЗАЦИИ ПЛАНОВ КООРДИНАЦИЙ СВЕТОФОРНЫХ ОБЪЕКТОВ УЧАСТКА ДОРОЖНОЙ СЕТИ

Руководитель 17.06.2022 доцент, кандидат физико-математических наук Д.В. Семенова

Выпускник 17.06.22 Т.И. Тисленко

Нормоконтролер 23.06.22 Т.Н. Шипина

Красноярск 2022