ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ ПРОЦЕССЫ

УДК 004.852 Научная статья
DOI: 10.35330/1991-6639-2025-27-2-86-102
EDN: NZSEKR

О применении обучения с подкреплением в задаче выбора оптимальной траектории движения

М. Г. Городничев

Московский технический университет связи и информатики
111024, Россия, Москва, ул. Авиамоторная, 8А

Аннотация. В данной статье рассматриваются современные методы обучения с подкреплением, с акцентом на их применение в динамичных и сложных средах. Исследование начинается с анализа основных подходов к обучению с подкреплением, таких как динамическое программирование, методы Монте-Карло, методы временной разницы и градиенты политики. Особое внимание уделяется методологии Generalized Adversarial Imitation Learning (GAIL) и ее влиянию на оптимизацию стратегий агентов. Приведено исследование безмодельного обучения и выделены критерии выбора агентов, способных работать в непрерывных пространствах действий и состояний. Экспериментальная часть посвящена анализу обучения агентов с использованием различных типов сенсоров, включая визуальные, и демонстрирует их способность адаптироваться к условиям среды, несмотря на ограничения разрешения. Представлено сравнение результатов на основе кумулятивной награды и длины эпизода, выявляющее улучшение производительности агентов на поздних этапах обучения. Исследование подтверждает, что использование имитационного обучения значительно повышает эффективность агента, сокращая временные затраты и улучшая стратегии принятия решений. Настоящая работа открывает перспективы для дальнейшего изучения механизмов улучшения разрешающей способности сенсоров и тонкой настройки гиперпараметров.

Ключевые слова: обучение с подкреплением, интеллектуальные агенты, оптимальная траектория, высокоавтоматизированные транспортные средства, обучение на основе политик, архитектуры «актер-критик», имитационное обучение, сенсоры, непрерывные состояния, дискретные состояния, PPO, SAC

Поступила 25.03.2025, одобрена после рецензирования 26.03.2025, принята к публикации 09.04.2025

Для цитирования. Городничев М. Г. О применении обучения с подкреплением в задаче выбора оптимальной траектории движения // Известия Кабардино-Балкарского научного центра РАН. 2025. Т. 27. № 2. С. 86–102. DOI: 10.35330/1991-6639-2025-27-2-86-102

MSC: 68T07 Original article

On the application of reinforcement learning in the task of choosing the optimal trajectory

M.G. Gorodnichev

Moscow Technical University of Communications and Informatics
111024, Russia, Moscow, 8A Aviamotornaya street

Abstract. This paper reviews state-of-the-art reinforcement learning methods, with a focus on their application in dynamic and complex environments. The study begins by analysing the main approaches to reinforcement learning such as dynamic programming, Monte Carlo methods, time-difference methods

and policy gradients. Special attention is given to the Generalised Adversarial Imitation Learning (GAIL) methodology and its impact on the optimisation of agents' strategies. A study of model-free learning is presented and criteria for selecting agents capable of operating in continuous action and state spaces are highlighted. The experimental part is devoted to analysing the learning of agents using different types of sensors, including visual sensors, and demonstrates their ability to adapt to the environment despite resolution constraints. A comparison of results based on cumulative reward and episode length is presented, revealing improved agent performance in the later stages of training. The study confirms that the use of simulated learning significantly improves agent performance by reducing time costs and improving decision-making strategies. The present work holds promise for further exploration of mechanisms for improving sensor resolution and fine-tuning hyperparameters.

Keywords: reinforcement learning, intelligent agents, optimal trajectory, highly automated vehicles, policy-based learning, actor-critic architectures, simulated learning, sensors, continuous states, discrete states, PPO, SAC

Submitted 25.03.2025, approved after reviewing 26.03.2025, accepted for publication 09.04.2025

For citation. Gorodnichev M.G. On the application of reinforcement learning in the task of choosing the optimal trajectory. *News of the Kabardino-Balkarian Scientific Center of RAS.* 2025. Vol. 27. No. 2. Pp. 86–102. DOI: 10.35330/1991-6639-2025-27-2-86-102

Введение

Основной задачей для лиц, ответственных за организацию дорожного движения, является обеспечение безопасности и оптимальности передвижения на дорогах общего пользования. Для решения данной задачи повсеместно внедряются системы помощи водителям (ADAS) различного уровня. Основной причиной аварий является человеческий фактор, который можно снизить за счет ADAS. В последние годы исследования, разработки и внедрение систем автопилотирования приобрели особенно высокий потенциал. Однако ограничениями для внедрения автоматизированных транспортных средств на дорогах общего пользования являются факторы, связанные с законодательством, сертификацией и стандартизацией.

Проблемами при разработке таких систем являются сложность сбора и недостаточность данных. Уровень развития техники на данный момент позволяет создавать сложные и реалистичные симуляции с многообразием параметров. В связи с этим применяются различные виды виртуальных 2D и 3D-симуляторов сложных социально-технических систем. Данный подход позволяет проводить научные исследования, испытания готовых образцов различных систем помощи водителям. Тем самым это позволяет повысить экономическую и ресурсную эффективность.

Цель данного исследования заключается в исследовании эффективности применения обучения с подкреплением (RL) в задаче выбора оптимальной траектории движения высокоавтоматизированных транспортных средств с учетом условий безопасности исходя из продольного и поперечного динамического габарита.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Сравнение обучения агентов RL на основе различной информации.

2. Создание окружения среды и обучение агентов RL на данных для оптимального выбора траектории движения

3. Сравнение производительности различных подходов к обучению.

В рамках данного исследования постараемся ответить на следующие вопросы, которые встают перед исследователями и инженерами в области разработки высокоавтоматизированных транспортных средств:

1. Насколько хорошо обученные агенты выполняют задачу выбора оптимальной траектории движения?

2. Каковы отличия использования RL по сравнению с базовыми моделями?

3. Можно ли выделить класс агентов, которые лучше справляются с задачей выбора оптимальной траектории?

В данной статье под оптимизацией траектории движения будем понимать процесс выбора траектории движения транспортного средства в конкретный момент времени и состоянии с учетом заданной цели.

Высокоавтоматизированное транспортное средство (агент) должно перманентно решать задачу выбора оптимальной траектории движения посредством построения объективной функции, выражающейся в минимизации или максимизации целевой переменной, в первую очередь для поддержания должного уровня безопасности за счет контроля продольного и поперечного динамического габарита. Однако стоит отметить, что нельзя забывать про оптимальность распределения транспортных средств в улично-дорожной сети.

В каждый момент времени *t* агент наблюдает совокупность векторов из множества *T*. *T* – это обратная связь с учетом контекста окружения среды, получаемая с различных типов источников, например, лидаров, сонаров, видеокамер и т.д. Исходя из типа источника информации формируются векторы *T*. Агент должен уметь обрабатывать данные, поступаемые с различных источников. Вектор (P_t) информации о возможных траекториях с учетом состояний (препятствий) определяется следующим образом:

$$P_t = \left[\log \frac{P_{1,t}}{P_{1,t-1}}, \log \frac{P_{2,t}}{P_{2,t-1}}, ..., \log \frac{P_{M,t}}{P_{M,t-1}} \right]^T \forall P_t \in R^M,$$

где M – общее число возможных траекторий в настоящий момент с учетом состояния.

Как правило, агент включает в себя один или несколько из трех составляющих: политику, модель и функцию вознаграждения. Политика – это набор правил, которыми руководствуется агент в каждом состоянии, т.е. это отношение между набором состояний и набором действий. Состояние S(t) – это некоторая позиция в среде, в которой агент может оказаться с учетом ограничений. Функция вознаграждения определяет качество каждого состояния или пары «состояние – действие». Модель – это представление агента об окружающей среде, с помощью которого агент предсказывает изменение среды [1–2].

Исходя из используемых составляющих алгоритмы обучения с подкреплением можно классифицировать по различным критериям:

1. Политика

Основанные на политиках: алгоритмы, которые обучаются на основе действий, текущих в данной политике: A2C, A3C, PPO и REINFORCE.

Не привязаны к политике: алгоритмы, которые могут обучаться на основе действий, полученных с помощью другой политики: DDPG, DQN, NAF, Q-learning, SAC и TD3.

Другие: для алгоритма Monte Carlo, который может использовать как основанные на политиках, так и наоборот.

2. Пространство действий

Непрерывные: алгоритмы, работающие с непрерывными действиями, включают A2C, A3C, DDPG, NAF, PPO, REINFORCE, SAC, TD3 и TRPO.

Дискретные: алгоритмы, работающие с дискретными действиями, включают DQN, Q-learning, SARSA, SARSA-Лямбда.

3. Пространство состояний

Непрерывные: алгоритмы, которые имеют непрерывное пространство состояний: DDPG, NAF, PPO, REINFORCE, SAC и TD3.

Дискретные: алгоритмы с дискретным пространством состояний: DQN, Monte Carlo, Q-learning, SARSA, SARSA-Лямбда.

4. Оператор

Q-value: алгоритмы, использующие значения Q для обновления политики: DDPG, DQN, Q-learning, Q-learning - Lambda, SAC, SARSA, SARSA - Lambda, TD3.

Advantage: алгоритмы, основанные на преимуществе действия: A2C, A3C, PPO, NAF, TRPO.

Выборочное среднее: используется в Monte Carlo подходах.

5. Класс

Actor-Critic: алгоритмы, сочетающие политику (Actor) и оценку ценности (Critic): A2C, A3C, NAF, PPO, REINFORCE, SAC и TRPO.

Основанные на данных: алгоритмы, основывающиеся преимущественно на оценке ценности: DQN, Monte Carlo, Q-learning, SARSA, SARSA - Lambda, TD3.

Основанные на политиках: метод, основанный на прямом управлении политикой, представлен алгоритмом REINFORCE.

Создание окружения

Под средой может выступать любая симуляция, которая обрабатывает действия агента и последствия этих действий. На вход поступает действие агента A(t) в состоянии S(t). После обработки получаем переход в следующее состояние S(t) с вознаграждением R(t+1). Вознаграждение R(t) возвращает числовое значение за нахождение агента в том или ином состоянии. Таким образом вознаграждение показывает, насколько данная совокупность ценна. Цель для агента описывается максимизацией прогнозирования кумулятивного вознаграждения. Под действием будем понимать разрешенные перемещения в конкретной среде [3].

Траектория представляет собой весовые коэффициенты в наборе всех допустимых в каждый момент времени t с учетом контекста окружения:

$$A_t = \left[A_{t,1}, A_{t,2}, \ldots, A_{t,M} \right]^T \forall A_t \in R^M$$

$$\sum_{i=1}^{M} A_{i,t} = 1$$

$$0 \le A_{i,t} \le 1 \forall i, t,$$

где i – одна из возможных траекторий.

Для достижения цели по направлению агента к заданной точке в трехмерной симуляции с учетом обхода препятствий был разработан симулятор маршрута [4]. Он состоит из модульных сегментов, которые могут случайным образом комбинироваться в ходе каждой тренировки. Генерация случайных маршрутов является необходимой для предотвращения обучения агента исключительно одному конкретному маршруту в процессе дальнейших тренировок [5].

Для создания виртуального окружения использовался симулятор MATSim [6]. Данный симулятор является открытым, что позволяет дописывать необходимые модули для проведения исследований. К преимуществам данного симулятора можно отнести: возможность создания большого числа агентов, работа с большими картами с высокой степенью детализации, моделирования различных типов транспорта и загрузки ранее полученных результатов моделирования.

Все эксперименты проводились на вычислительном сервере МТУСИ, который имеет следующие характеристики: CPU AMD EPYC 7742, 64 ядра, 128 потоков, GPU 8 x NVIDIA Tesla A100-SXM4-40GB, RAM 16 x Samsung DDR4 32 GB 3200 MT/s.

Исследуемая конфигурация представляет собой дискретное поле с сеточной структурой, на котором располагаются элементы трассировки. Выделяются три категории элементов: начальный, угловой и прямой блоки. Процедура генерации карты инициируется установкой стартового блока, который позиционируется в одном из четырех направлений: вверх, вправо, вниз или влево. В процессе генерации к стартовому блоку последовательно присоединяются случайным образом выбранные блоки из доступных категорий. Эта процедура продолжается итерационно, до того момента, пока трассировка либо не замкнется, либо не достигнет предустановленного количества элементов.

В процессе формирования трассы каждое последующее звено и его ориентация определяются случайным образом, однако при этом соблюдается условие, что генеральный план должен напоминать непрерывную дорогу. Для этого первоначально создается список блоков, каждый из которых содержит информацию о направлениях входа и выхода. На основе этих данных выполняются вставка и вращение визуальных образов блоков.

После достижения агентом всех заданных целей на трассе инициируется генерация новой карты. В случае столкновения агента с физическим барьером он перемещается в инициальную позицию на трассе, а все цели восстанавливаются. С целью предотвращения избыточного времени простоя агента внедрен 30-секундный таймер. По истечении этого времени симуляция запускается заново. Обнуление таймера производится исключительно при достижении агентом ближайшей цели.

В ходе исследования было выявлено, что имеющиеся стимулы оказались недостаточно эффективными, так как агент испытывал затруднения в понимании предписанных действий в отсутствие прямого достижения цели. Для повышения ясности и эффективности стимулирующей структуры системы были внедрены дополнительные штрафные и входные механизмы:

- Уменьшение расстояния между агентом и ближайшей целью сопровождалось применением штрафа в размере 0,05 балла.
- Увеличение расстояния до цели влекло за собой применение отрицательного штрафа, равного -0,05 балла.
- В систему входных данных был интегрирован вектор, отражающий расстояние до ближайшей цели, что служило дополнительным ориентиром для агента.

Была произведена модификация мишеней с целью обеспечения их видимости в виде отчетливых прямоугольных форм, расположенных на каждом элементе трассы, что способствует четкой идентификации корректных точек входа. Кроме того, была разработана и внедрена дополнительная функциональность, обеспечивающая ротацию целевых объектов на поворотных участках дороги, с целью их ориентации под оптимальным углом в 45 градусов относительно блока поворота.

Сценарный модуль агента включает в себя код, предназначенный для выполнения симуляции. В данном модуле устанавливаются условия, при которых агент получает стимулы для выполнения конкретных действий, а также определяется последовательность операций в рамках симуляционного процесса. С целью интенсификации процесса обучения на начальных этапах была применена методология симуляционного обучения. Для реализации этого подхода была создана демонстрационная версия, состоящая из ряда тестовых испытаний.

Выбор базовой модели обучения

Существуют четыре ключевых подхода к обучению с подкреплением: динамическое программирование, методы Монте-Карло, методы временной разницы и методы градиента политики. Динамическое программирование представляет собой двухэтапный процесс, применяющий уравнение Беллмана для оптимизации политики после проведения ее оценки [7]. Этот метод находит применение в случаях, когда модель среды заранее известна в полном объеме.

Методы Монте-Карло, напротив, изучают опыт эпизодов без учета динамики среды, вычисляя наблюдаемый средний выигрыш как приближенную оценку прогнозируемой траектории. Таким образом, обучение возможно только по завершении всех эпизодов [8–9].

Методы временной разницы обучаются на незавершенных эпизодах, используя бутстреппинг для оценки выигрыша, что делает их своеобразным гибридом динамического программирования и методов Монте-Карло. В отличие от методов временной разницы, которые полагаются на оценку выигрыша для выбора оптимальной политики, методы градиента политики исходят непосредственно из оценки самой политики.

Обучение политике может происходить как с использованием текущей политики (on-policy), так и вне ее пределов (off-policy). В случае on-policy обучения агенты анализируют ту же политику, которая привела к выполнению действия, тогда как в off-policy обучении агенты рассматривают политику, которая может не совпадать с той, которая непосредственно используется [10].

Выбор агентов для обучения с подкреплением (RL) осуществлялся на основе трех ключевых критериев. Во-первых, агент RL должен быть безмодельным, так как данная работа акцентирует внимание на методах безмодельного обучения. Во-вторых, выбранный агент должен быть ранее использован в аналогичных исследованиях, зафиксированных в научной литературе. В-третьих, агент должен поддерживать непрерывные пространства действий и состояний.

В итоге в данной работе для исследования были выбраны два агента: PPO и SAC.

Proximal Policy Optimization (PPO) – это набор алгоритмов, применяемых в обучении с подкреплением без использования модели. Алгоритмы PPO являются методами политико-градиентного типа, что означает, что они осуществляют поиск в пространстве политик, а не присваивают значения парам «состояние – действие». PPO обладает преимуществами алгоритмов оптимизации политики доверительного региона (TRIPOD), но при этом они проще в реализации, более универсальны и имеют лучшую сложность выборки [11]. PPO использует нейронные сети для аппроксимации идеальных функций, которые сопоставляют наблюдения агента с оптимальными действиями в данных состояниях [12].

Soft-Actor Critic (SAC) работает в режиме off-policy, что позволяет ему обучаться на основе опыта, накопленного в прошлом [13]. Накопленный опыт помещается в буфер воспроизведения и используется случайным образом во время тренировок. Это делает

SAC значительно более эффективным с точки зрения выборки, часто требуя в 5–10 раз меньше данных для выполнения той же задачи, что и PPO. Однако SAC обычно требует больших обновлений модели. SAC является оптимальным выбором для более сложных или медленных условий (примерно 0,1 секунды на шаг или более) [14]. Также SAC работает по принципу «максимальной энтропии», что позволяет проводить исследования более естественным образом.

Обучение модели на базе сенсоров

В начальной версии обучающего алгоритма применяется компонент, имитирующий работу лидара. Данный компонент представляет собой датчик, который поддерживает наблюдения на основе излучения лучей, исходящих из центральной точки агента. Этот датчик способен фиксировать расстояния до окружающих объектов, обеспечивая тем самым расширенное восприятие окружающей среды [15].

В процессе обучения были выявлены различные проблемы, одной из которых стало то, что агент на некоторых участках трассы не мог продвигаться дальше из-за частых столкновений с барьерами на поворотах после достижения цели на длинном прямом участке. Основной причиной этого было превышение скорости агентом, что препятствовало своевременной остановке на повороте, так как барьеры оставались вне поля зрения из-за нахождения цели на их пути. Для решения данной проблемы в агенте был добавлен дублирующий компонент сенсора, в результате чего каждый сенсор специализировался на взаимодействии с конкретным типом объектов, также была увеличена дальность их действия. Таким образом, один сенсор отслеживал исключительно барьеры, в то время как другой – только цели, и оба сенсора обеспечивали более дальнюю видимость.

Это изменение позволило устранить проблему на поворотах после разгона, так как каждый сенсор функционировал независимо, предотвращая возникновение слепых зон. Однако это вызвало новый эффект: сенсор, взаимодействующий с целями, начал подмечать цели за барьерами, что иногда приводило к намеренным столкновениям агента с препятствиями в попытке достичь цели на соседних участках трассы. Для устранения этой проблемы было решено визуализировать только одну цель за раз. Переключение между целями реализовывалось по мере прохождения трассы.

На начальном этапе агент демонстрировал произвольные действия, исследуя окружающую среду, – это можно охарактеризовать как фазу исследования. Под влиянием стимулов агент начинал постепенно осознавать, что его основной задачей является приближение к целям и избегание препятствий, хотя этот процесс обучения происходил довольно медленно.

Были замечены случаи, когда агент застревал на определенных участках трассы, а также, овладев навыками на одной трассе, испытывал затруднения на другой, в нетипичных для него ситуациях. Это обусловлено тем, что агент привыкал к элементам одного маршрута и, попадя в неизвестные условия, не сразу находил оптимальную стратегию действий, что можно отнести к явлению переобучения. Однако данное явление не представляет собой серьезной проблемы, требующей внешнего вмешательства, так как при предоставлении агенту большего количества времени на тренировку он самостоятельно преодолевает подобные препятствия.

В ходе применения имитационного обучения агент не демонстрировал мгновенное выполнение маршрута. Поведение агента стало более уверенным, и благодаря интеграции имитационного моделирования с традиционным подходом обучения действия агента были

более эффективными. При соответствующей настройке интенсивности воздействия имитационного компонента положительные результаты становились очевидными. Даже при ограниченном времени на обучение, эквивалентном 250 000 действиям, наблюдалась значительная разница в скорости освоения заданий. В частности, в рамках стандартного обучения с подкреплением агент мог не завершить ни одного маршрута. Однако при использовании имитационного обучения агент демонстрировал способность прохождения нескольких трасс, достигая до 17 маршрутов в установленные временные рамки.

Для сравнения результатов была составлена таблица 1 по каждому из агентов. В таблице представлены результаты обучения агентов по взаимодействию со средой. Основные параметры – это кумулятивная награда и длина эпизода.

Таблица 1. / Table 1.

Epochs	Cumulative reward (PPO)	Cumulative reward (SAC)	Episode length (PPO)	Episode length (SAC)
0k	0	0	1600	1600
40k	2	1	1400	1500
80k	5	3	1200	1300
120k	7	6	1000	1100
160k	8	7	900	1000
200k	10	9	800	900

Метрика кумулятивной награды демонстрирует способность агента принимать эффективные решения в процессе обучения. Награда служит индикатором успешности выполнения задач агентом. Данный параметр при обучении агента PPO постепенно увеличивается, что свидетельствует об улучшении стратегии агента. На последних этапах обучения (160k до 200k) награда достигает максимума.

Показатели длины эпизода указывают, сколько шагов необходимо агенту для завершения эпизода. На ранних этапах обучения (0k – 80k) длина эпизода была высокой, но постепенно снижается, что показывает адаптацию агента к среде. К 200k шагам длина стабилизируется на уровне 800–900 шагов, демонстрируя, что агент научился эффективно достигать целей.

Исходя из данных, можно сделать вывод, что агент успешно адаптируется к условиям обучения, улучшая производительность и взаимодействие с окружающей средой. Также были получены данные, показывающие динамику потерь в процессе обучения агента. Эти данные подчеркивают прогресс агента и его взаимодействие с окружающей средой.

Как следует из данных, представленных в таблице 2, показатель GAIL Loss (потери, связанные с методом Generalized Adversarial Imitation Learning [16]) демонстрирует устойчивую тенденцию к снижению. На начальном этапе обучения потери составляли 1.10, однако к 200 тысячам итераций этот показатель снизился до 0.60. Такая динамика свидетельствует о том, что агент успешно адаптируется к среде и постепенно улучшает свои стратегии, минимизируя ошибки в процессе обучения.

Таблица 2. / Table 2.

Epochs	GAIL Loss (PPO)	Policy Loss (PPO)	Pretraining Loss (PPO)	Value Loss (PPO)
0k	1.10	0.20	0.75	60
40k	0.90	0.18	0.70	45
80k	0.80	0.15	0.65	30
120k	0.75	0.12	0.60	25
160k	0.65	0.05	0.56	20
200k	0.60	0.04	0.55	19

Параметр Policy Loss, отражающий потери, связанные с политикой агента, также показывает значительное улучшение. На начальных этапах обучения значение данного параметра составляло 0.20, однако к концу обучения оно снизилось до 0.04. Это указывает на эффективную оптимизацию процесса принятия решений агентом, что подтверждает его способность корректировать свои действия в соответствии с изменяющимися условиями среды.

Показатель Pretraining Loss, характеризующий потери на этапе предобучения, также демонстрирует положительную динамику. Исходное значение данного параметра составляло 0.75, однако к завершению обучения оно снизилось до 0.55. Это свидетельствует о том, что процесс предобучения был успешно завершен, и агент смог эффективно использовать полученные на этом этапе знания для дальнейшего обучения.

Наиболее заметное улучшение наблюдается в параметре Value Loss, который отражает потери, связанные с оценкой ценности действий агента. На начальном этапе обучения значение этого параметра составляло 60, однако к 200 тысячам итераций оно снизилось до 19. Такое резкое снижение указывает на значительное улучшение способности агента оценивать ценность своих действий, что является ключевым фактором для повышения эффективности его стратегий.

Таким образом, анализ представленных данных позволяет сделать вывод, что использование метода GAIL способствует значительному улучшению показателей обучения агента. Снижение потерь по всем ключевым параметрам (GAIL Loss, Policy Loss, Pretraining Loss и Value Loss) подтверждает эффективность данного подхода для оптимизации стратегий агента в заданной среде.

В таблице 3 представлены данные о политике обучения агента, на основании которых можно судить о производительности подхода. Представленные метрики помогают оценить прогресс агента и его адаптацию к динамичной среде.

Исследование динамики изменения ключевых параметров в процессе обучения интеллектуального агента выявляет ряд значительных трендов, указывающих на повышающуюся эффективность его стратегий и общую продуктивность. Детальный анализ данных позволяет заключить, что позитивные изменения отражаются на стабильности и детерминированности принимаемых агентом решений.

Первое внимание заслуживает коэффициент бета, который демонстрирует последовательное снижение. Этот факт указывает на уменьшение стохастичности в выборе агентом действий. По мере завершения процесса обучения данный коэффициент приближается к

значению 1.50, что свидетельствует о переходе агента к более систематизированному подходу в принятии решений. Это можно интерпретировать как успешное уменьшение случайных колебаний в действиях агента, что способствует оптимизации его стратегий.

Таблица 3. / Table 3.

Epochs	Beta (PPO)	Entropy (PPO)	Epsilon (PPO)	Extrinsic Reward (PPO)	Extrinsic Value Estimate (PPO)	GAIL Expert Estimate (PPO)
0k	4.40	2.10	0.20	0	0	0.60
20k	3.80	1.90	0.18	0.50	0.40	0.65
40k	3.50	1.60	0.16	0.80	0.60	0.70
60k	3.00	1.30	0.14	1.00	0.75	0.75
80k	2.50	1.20	0.12	1.50	0.80	0.75
100k	2.00	1.15	0.10	1.70	0.85	0.80
120k	1.80	1.00	0.08	1.80	0.70	0.85
140k	1.70	0.90	0.06	1.90	0.60	0.85
160k	1.60	0.80	0.05	2.00	0.65	0.90
180k	1.50	0.70	0.05	2.10	0.72	0.92

Энтропия является еще одним ключевым показателем, и ее динамика показывает уменьшение. Это указывает на возрастание уверенности агента в выборе оптимальных стратегий и сокращение неопределенности в его действиях. Данные изменения подтверждают гипотезу о том, что процесс обучения позволяет формировать у агента устойчивые и повторяемые стратегии поведения.

Параметр эпсилон, отражающий степень случайности в действиях, также снижается, достигая значения 0.05 к финалу обучения. Такое развитие событий указывает на уменьшение доли случайных решений и рост надежности стратегий, выбираемых агентом. Агент все менее зависим от случайного выбора и все более полагается на усвоенные стратегии, что свидетельствует об успехах в обучении.

Параметр внешних наград показывает устойчивую положительную динамику в течение всего процесса обучения, означая, что агент эффективно решает поставленные задачи и постепенно повышает свою результативность. Рост внешних наград демонстрирует, что агент не только приспосабливается к окружающей среде, но и активно улучшает свои показатели.

Параметры внешней ценности, отражающие успешность оценки агентом собственной деятельности, также демонстрируют положительную динамику, подтверждая, что обучение положительно влияет на его способности оценивать ценность собственных действий.

Наконец, оценка эксперта, основанная на методике GAIL, показывает, что агент все более точно и успешно воспроизводит экспертные стратегии. Это говорит о том, что процесс обучения приближает его поведенческие паттерны к экспертным, что является ключевым показателем успеха.

В заключение отметим, что приведенный анализ ключевых параметров подтверждает успешность обучения агента, которое способствует значительному улучшению его стратегической эффективности. Снижение стохастичности в деле, увеличение уверенности, рост внешних наград и развитие способностей по имитации экспертного поведения демонстрируют, что агент активно адаптируется к среде и достигает возрастания производительности.

На основе данного анализа можно утверждать, что агент, оснащенный лучевым сенсором, демонстрирует наивысшую результативность в выполнении поставленных задач. В экспериментальных условиях, при использовании PPO с лимитом в 200 000 итераций, данный агент успешно преодолел около 20 трасс, что значительно превосходит результаты, полученные при использовании SAC.

Модель на основе сеточных сенсоров

В данной симуляции в качестве ключевого сенсорного элемента использовался модуль для распознавания двух категорий объектов: преграды и целевые точки. Визуальная идентификация на сетке осуществлялась через цветовое кодирование: цели выделялись зеленым, а барьеры – красным цветом. Сенсорная матрица обладает размерностью 20x20 ячеек, каждая из которых составляет 0.5 условных единиц. Это сеточное покрытие обеспечивает агенту анализ и восприятие окружающих объектов [17].

При внешнем мониторинге обучающей динамики отмечены значительные трудности, возникшие у агента в процессе освоения задачи. Возможные причины могут включать неэффективные начальные параметры конфигурации или недостаточную продолжительность тренировочного периода. Для улучшения результатов необходимо провести дальнейшие эксперименты, включающие коррекцию гиперпараметров и увеличение времени обучения.

Таблица 4 содержит результаты применения алгоритмов PPO и SAC, что позволяет оценить их эффективность и определить направления для оптимизации.

Таблица 4. / Table 4.

Epochs	Cumulative Reward (PPO)	Cumulative Reward (SAC)	Episode Length (PPO)	Episode Length (SAC)
0k	0	0	1800	1800
20k	2	1.5	1600	1700
40k	3	2.0	1400	1500
60k	6	3.0	1200	1300
80k	7	4.5	1000	1100
100k	10	5.0	800	1000
120k	12	6.0	600	900
140k	14	8.0	500	700
160k	15	10.0	400	600
180k	16	12.0	300	500

Анализ динамики параметра кумулятивной награды для алгоритма PPO демонстрирует устойчивый рост от начального значения, близкого к нулю, до 16 единиц к завершению обучения. Такая положительная динамика свидетельствует о постепенном улучшении навыков агента и его способности эффективно взаимодействовать с окружающей средой. В то же время, для алгоритма SAC также наблюдается увеличение кумулятивной награды, однако его результаты являются менее выраженными по сравнению с PPO, что может указывать на различия в эффективности данных методов в контексте поставленной задачи.

Для алгоритма PPO длина эпизодов сократилась с 1800 до 300 шагов, что свидетельствует о значительном повышении эффективности агента. Уменьшение продолжительности эпизодов, сопровождаемое ростом кумулятивной награды, подтверждает оптимизацию стратегии агента и его способность быстрее достигать целевых состояний. Это указывает на то, что агент не только улучшает свои навыки, но и становится более рациональным в использовании ресурсов и времени. Таким образом, оба алгоритма – PPO и SAC – демонстрируют успешную адаптацию к среде и стабильный прогресс в процессе обучения. Однако PPO показал себя лучше, с связи с чем далее будем приводить анализ по данному алгоритму. Результаты мониторинга поведения агента представлены в таблице 5.

Таблица 5. / Table 5.

Epochs	GAIL Loss (PPO)	Policy Loss (PPO)	Pretraining Loss (PPO)	Value Loss (PPO)
0k	0.50	0.052	0.85	70
20k	0.45	0.051	0.78	50
40k	0.40	0.050	0.69	30
60k	0.35	0.055	0.60	25
80k	0.30	0.049	0.55	20
100k	0.25	0.047	0.50	18
120k	0.20	0.045	0.48	15
140k	0.15	0.046	0.40	14
160k	0.10	0.040	0.30	12
180k	0.05	0.039	0.20	10

Анализ динамики изменения параметра GAIL Loss указывает на недостаточную эффективность текущей процедуры клонирования поведения. Наблюдаемое незначительное снижение уровня потерь свидетельствует о том, что модель не способна точно воспроизводить экспертные стратегии. Это указывает на необходимость пересмотра подхода к обучению, поскольку текущая конфигурация не достигает достаточно высокой точности в имитации целевого поведения.

Параметр Policy Loss, связанный с потерями агента при реализации стратегии, практически не изменяется на протяжении всего обучения. Такая стагнация предполагает, что агент не способен адаптировать свои стратегии, возможно, из-за недостаточной гибкости модели или неоптимальных гиперпараметров. Это подчеркивает необходимость тщательной настройки алгоритма для улучшения процессов обучения.

Value Loss, отвечающий за потери, связанные с оценкой ценности действий, также демонстрирует неидеальное поведение. Хотя на начальных этапах обучения можно наблюдать ожидаемый рост потерь, далее происходит их увеличение после краткосрочного снижения, что противоречит ожидаемым тенденциям. Такая динамика свидетельствует о неспособности модели точно прогнозировать будущие действия, что может быть вызвано ограничениями архитектуры или недостаточным объемом обучающих данных.

В общем выявленные аномалии в динамике параметров потерь подчеркивают необходимость пересмотра методов обучения. Для достижения более устойчивых и прогнозируе-

мых результатов требуется провести углубленные исследования, направленные на оптимизацию гиперпараметров, совершенствование архитектуры модели и увеличение объема данных. Лишь при устранении этих недостатков возможно значительное улучшение эффективности и точности модели.

Анализ результатов симуляции, в которой использовались сеточные сенсоры, показал неудовлетворительные показатели. Это может быть связано либо с неэффективностью сенсорного компонента, либо с недостаточным качеством данных демонстрационной версии, используемой для имитации. Низкая результативность объясняется ограниченной способностью сеточных сенсоров точно распознавать и интерпретировать важные элементы окружающей среды, что ведет к неоптимальным стратегиям. Также возможная причина заключается в недостаточной репрезентативности предоставленных данных, что отрицательно сказывается на процессе обучения.

Для повышения эффективности обучения целесообразно пересмотреть методы и параметры, включая модификацию архитектуры сенсоров, оптимизацию гиперпараметров обучения и улучшение качества демонстрационных данных. Дальнейшие исследования могут выявить более продуктивные подходы к обучению, что позволит улучшить производительность агента в решении поставленных задач.

Модель на базе визуальных сенсоров

На завершающем этапе эксперимента интеграция визуального сенсора сыграла ключевую роль, обеспечивая восприятие окружающей среды путем обработки изображений [18]. Входные изображения были установлены с разрешением 32×32 пикселя как минимально возможным, чтобы сохранить различимость ключевых образов при минимизации вычислительной нагрузки. Такое уменьшение размера было необходимо из-за значительной ресурсоемкости обработки визуальных данных, которая экспоненциально возрастает с увеличением разрешения.

Экспериментальные условия предполагали обучение агента только с использованием алгоритма PPO. SAC, напротив, не демонстрировал необходимой эффективности. Это может быть обусловлено тем, что для SAC требуется более тонкая настройка гиперпараметров или же значительные вычислительные мощности для достижения аналогичных с PPO результатов.

Применение визуального сенсора с низким разрешением дало возможность сократить вычислительные затраты, сохраняя при этом способность агента распознавать главные объекты окружающей среды. Однако недостигнутая эффективность агента SAC говорит о необходимости продолжения исследований в области оптимизации алгоритмов для работы с визуальными данными.

Результаты эксперимента хотя и уступают показателям лучевых сенсоров, но превышают эффективность системы с сеточными сенсорами. Процесс обучения успешно проходил даже при ограниченном разрешении сенсора, демонстрируя адаптационные способности агента к работе с недостатком данных, что подтверждает перспективность визуальных сенсоров в этой сфере.

Таблицы 6 и 7 представляют ключевые метрики процесса обучения, служащие базой для сравнительной оценки влияния разных типов сенсоров на эффективность модели.

Исходя из данных применение визуального сенсора показало свою применимость и перспективы для дальнейших исследований и оптимизации обучения, несмотря на ограничение разрешения.

Таблица 6. / Table 6.

Epochs	Cumulative Reward	Episode Length
0k	6	1800
20k	8	1750
40k	10	1600
60k	12	1500
80k	14	1400
100k	16	1300
120k	17	1400
140k	15	1600
160k	18	1550
180k	18	1350

Policy Loss устойчиво снижается, подтверждая успешную оптимизацию стратегии агента и его адаптацию к среде. Value Loss демонстрирует колебания: первоначальный рост сменяется временным снижением и последующим увеличением, что указывает на возможные ошибки в прогнозировании будущих состояний, вероятно, из-за ограничений архитектуры сети или недостатка обучения. Entropy постепенно падает, отражая повышение уверенности агента в принимаемых решениях. Extrinsic Reward стабильно растет, подтверждая улучшение общей производительности. Несмотря на проблемы с нестабильностью Value Loss, система показывает признаки эффективного обучения. Для ускорения прогресса и стабилизации результатов, возможно, потребуются дополнительная настройка гиперпараметров или увеличение длительности тренировки.

Таблица 7. / Table 7.

Epochs	Beta	Entropy	Epsilon	Extrinsic Reward	Extrinsic Value Estimate	GAIL Expert Estimate
0k	4.50	2.15	0.20	0	0.30	0.40
20k	4.00	1.90	0.18	3	0.25	0.45
40k	3.80	1.75	0.16	5	0.28	0.50
60k	3.50	1.50	0.14	7	0.40	0.55
80k	3.00	1.30	0.12	10	0.50	0.60
100k	2.50	1.10	0.10	12	0.55	0.65
120k	2.00	0.90	0.08	14	0.60	0.70
140k	1.80	0.75	0.05	15	0.65	0.75
160k	1.60	0.60	0.05	16	0.67	0.80
180k	1.50	0.50	0.04	18	0.70	0.85

Эксперимент подтвердил трудности работы с визуальными сенсорами, увеличив время обучения и вызвав технические проблемы. Однако, несмотря на низкое разрешение сенсоров и временные ограничения, агент успешно справился с задачами на пяти трассах, частично адаптировавшись к среде и освоив базовые стратегии. Результаты показывают, что даже при субоптимальных параметрах агент способен развиваться, открывая перспективы для дальнейших исследований по улучшению разрешения, увеличению времени обучения и тонкой настройке гиперпараметров.

Заключение

Исследование посвящено оценке эффективности методов обучения с подкреплением для выбора оптимальных траекторий автономных транспортных средств. Основное внимание уделено сравнению трех типов сенсоров – лучевых, сеточных и визуальных – и их влиянию на обучение агентов, учитывая преимущества и ограничения каждого из них.

Лучевые сенсоры показали наилучшие результаты, пройдя около 20 трасс благодаря высокой точности и дальности обнаружения объектов, что улучшило навигацию и предотвратило столкновения, несмотря на повышенные вычислительные затраты. Сеточные сенсоры оказались наименее эффективными из-за низкой точности и слабой адаптивности, несмотря на простоту и низкие ресурсные требования. Визуальные сенсоры заняли среднее положение, пройдя около пяти трасс, что свидетельствует о потенциальной перспективности при улучшении алгоритмов и увеличении разрешения, несмотря на высокую вычислительную сложность.

Анализ метрик показал улучшение стратегий агентов с лучевыми и визуальными сенсорами в отличие от сеточных, где наблюдались значительные колебания, требующие дальнейшей оптимизации. Рекомендуется продлить обучение для сеточных и визуальных сенсоров, чтобы снизить риск переобучения и улучшить адаптацию. Планируются дополнительные исследования по настройке гиперпараметров и тестированию более сложных архитектур нейронных сетей, таких как трансформеры или графовые сети, для обработки данных от разнородных сенсоров. Комбинирование различных сенсоров может повысить точность систем автоматического управления за счет улучшения восприятия окружения.

Обучение с подкреплением открывает перспективы для управления автономным транспортом, но требует оптимизации с учетом вычислительных ресурсов и качества данных. Результаты закладывают фундамент для дальнейшего развития технологий автономного вождения и систем помощи водителю.

СПИСОК ЛИТЕРАТУРЫ / REFERENCES

1. Zhang S., Xia Q., Chen M., Cheng S. Multi-Objective Optimal Trajectory Planning for Robotic Arms Using Deep Reinforcement Learning. *Sensors.* 2023. Vol. 23. P. 5974. DOI: 10.3390/s23135974

2. Tamizi M.G., Yaghoubi M., Najjaran H. A review of recent trend in motion planning of industrial robots. *International Journal of Intelligent Robotics and Applications*. 2023. Vol. 7. Pp. 253–274. DOI:10.1007/s41315-023-00274-2

3. Kollar T., Roy N. Trajectory Optimization using Reinforcement Learning for Map Exploration. *International Journal of Robotics Research.* 2008. Vol. 27. No. 2. Pp. 175–196. DOI: 10.1177/0278364907087426

4. Acar E.U., Choset H., Zhang Y., Schervish M. Path planning for robotic demining: robust sensor-based coverage of unstructured environments and probabilistic methods. *International Journal of Robotics Research.* 2003. Vol. 22. No. 7–8. Pp. 441–466.

5. Cohn D.A., Ghahramani Z., Jordan M.I. Active learning with statistical models. *Journal of Artificial Intelligence Research.* 1996. No. 4. Pp. 705–712.

6. Axhausen K. et al. Introducing MATSim. In: Horni, A et al (eds.). *Multi-Agent Transport Simulation MATSim.* London: Ubiquity Press. 2016. Pp. 3–8. DOI: 10.5334/baw.1

7. Wu G., Zhang D., Miao Z., Bao W., Cao J. How to Design Reinforcement Learning Methods for the Edge: An Integrated Approach toward Intelligent Decision Making. *Electronics.* 2024. Vol. 13. P. 1281. DOI: 10.3390/electronics13071281

8. Zhou T., Lin M. Deadline-aware deep-recurrent-q-network governor for smart energy saving. *IEEE Transactions on Network Science and Engineering.* 2021. Vol. 9. Pp. 3886–3895. DOI: 10.1109/TNSE.2021.3123280

9. Yang Y., Wang J. An overview of multi-agent reinforcement learning from game theoretical perspective. arXiv 2020, arXiv:2011.00583. DOI: 10.48550/arXiv.2011.00583

10. Mazyavkina N., Sviridov S., Ivanov S., Burnaev E. Reinforcement learning for combinatorial optimization: A survey. Comput. Oper. Res. 2021. Vol. 134. P. 105400. DOI: 10.1016/j.cor.2021.105400

11. Junwei Zhang, Zhenghao Zhang, Shuai Han, Shuai Lü, Proximal policy optimization via enhanced exploration efficiency. *Information Sciences.* 2022. Vol. 609. Pp. 750–765. ISSN 0020-0255. DOI: 10.1016/j.ins.2022.07.111

12. Hessel M., Modayil J., H. van Hasselt, Schaul T. et al. Rainbow: Combining improvements in deep reinforcement learning. *In AAAI Conference on Artificial Intelligence.* 2018. Pp. 3215–3222. DOI: 10.1609/aaai.v32i1.11796

13. Haarnoja T., Zhou A., Abbeel P., Levine S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *In International Conference on Machine Learning*. 2018. Pp. 1856–1865. DOI: 10.48550/arXiv.1801.01290

14. Lillicrap T.P., Hunt J.J., Pritzel A. et al. Continuous control with deep reinforcement learning. arXiv:1509.02971v1. 2015. file:///C:/Users/%D0%90%D1%80%D1%81%D0%B5%D0%BD/Downloads/1509.02971v1.pdf

15. Chen Y., Lam C.T., Pau G., Ke W. From Virtual to Reality: A Deep Reinforcement Learning Solution to Implement Autonomous Driving with 3D-LiDAR. *Applied Sciences.* 2025. Vol. 15. No. 3. P. 1423. DOI: 10.3390/app15031423

16. Guoyu Zuo, Kexin Chen, Jiahao Lu, Xiangsheng Huang. Deterministic generative adversarial imitation learning. *Neurocomputing.* 2020. Vol. 388. Pp. 60–69. ISSN 0925-2312. DOI: 10.1016/j.neucom.2020.01.016

17. Sawada R. Automatic Collision Avoidance Using Deep Reinforcement Learning with Grid Sensor. In: Sato, H., Iwanaga, S., Ishii, A. (eds). *Proceedings of the 23rd Asia Pacific Symposium on Intelligent and Evolutionary Systems.* IES 2019. Proceedings in Adaptation, Learning and Optimization. Springer, Cham. 2020. Vol. 12. Pp. 17–32. DOI: 10.1007/978-3-030-37442-6_3

18. Hachaj T., Piekarczyk M. On Explainability of Reinforcement Learning-Based Machine Learning Agents Trained with Proximal Policy Optimization That Utilizes Visual Sensor Data. *Applied Sciences.* 2025. Vol. 15. No. 2. P. 538. DOI: 10.3390/app15020538

Финансирование. Исследование проведено без спонсорской поддержки.

Funding. The study was performed without external funding.

Информация об авторе

Городничев Михаил Геннадьевич, кан. техн. наук, доцент, декан факультета «Информационные технологии», Московский технический университет связи и информатики;
111024, Россия, Москва, ул. Авиамоторная, 8А;
m.g.gorodnichev@mtuci.ru, ORCID: https://orcid.org/0000-0003-1739-9831, SPIN-код: 4576-9642

Information about the author

Mikhail G. Gorodnichev, Candidate of Engineering Sciences, Associate Professor, Dean of the Faculty of Information Technology, Moscow Technical University of Communications and Informatics;
111024, Russia, Moscow, 8A Aviamotornaya street;
m.g.gorodnichev@mtuci.ru, ORCID: https://orcid.org/0000-0003-1739-9831, SPIN-code: 4576-9642