МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»
Кафедра математической кибернетики и компьютерных наук

ПРИМЕНЕНИЕ АЛГОРИТМОВ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ ДЛЯ ЗАДАЧИ РАЗРАБОТКИ ИГРОВОГО ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные технологии
факультета КНиИТ
Горбачевой Марины Вадимовны

Научный руководитель
к. ф.-м. н., доцент _____ А. С. Иванов

Заведующий кафедрой
к. ф.-м. н., доцент _____ С. В. Миронов

Саратов 2022

ВВЕДЕНИЕ

Актуальность темы. Достижение компьютером уровня профессиональных игроков может быть осуществлено благодаря множеству разнообразных инструментов: от классических алгоритмов искусственного интеллекта до методов и подходов, присущих машинному обучению, в частности, его области, называемой обучением с подкреплением. Подходы, используемые в последнем, открывают новые возможности для разработки игрового искусственного интеллекта, в том числе для игр, обучение классическими методами в которых не представляется возможным — например, 3D-видеоиграх. В качестве примера можно привести модель OpenAI Five, в 2019ом году ставшую чемпионом по игре в многопользовательскую онлайн-игру Dota 2. Однако она является лишь одним из множества проявлений нарастающей тенденции переноса соревнования игрового искусственного интеллекта и человека из плоскости «классических» игр в пространство видеоигр. В этой связи кажется ясным, что тема применения обучения с подкреплением для разработки игрового искусственного интеллекта является в настоящее время актуальной как никогда.

Цель работы. Целью бакалаврской работы является исследование концепций и подходов, применяемых в обучении с подкреплением, связи с теорией игр и особенностей обучения в мультиагентных средах, являющихся играми, а также проведение экспериментов над алгоритмами обучения с подкреплением в разнообразных игровых средах.

Поставленная цель определила следующие задачи:

- исследование взаимосвязи между областями искусственного интеллекта, теории игр и обучения с подкреплением;
- выделение особенностей мультиагентного обучения с подкреплением относительно одноагентного;
- обзор алгоритмов обучения с подкреплением и их таксономии;
- программная реализация различных алгоритмов обучения с подкреплением;
- разработка программной среды, или фреймворка, для проведения экспериментов в области обучения с подкреплением;
- сбор статистики и анализ результатов обучения представленных алгоритмов в различных играх.

Методологические основы. Методологическими основами для исследований в областях искусственного интеллекта, теории игр и обучения с подкреплением послужили работы Стюарта Рассела и Питера Норвига, Джона фон Неймана и Оскара Моргенштерна, Ричарда Саттона и Эндрю Барто.

Теоретическая значимость работы. Теоретическая значимость бакалаврской работы заключается в наблюдениях и выводах, сформированных по итогам проведения экспериментов в реализованной программе, которые характеризуют поведение различных алгоритмов обучения с подкреплением в разных вариантах игровых сред.

Практическая значимость работы. Практическая значимость бакалаврской работы заключается в том, что реализованная программа представляет собой гибкий инструмент для проведения экспериментов с применением алгоритмов обучения с подкреплением в игровых средах, что выражается в легкости реализации и включения в эксперименты новых алгоритмов и сред и что, как следствие, предоставляет широкий простор для возможных исследований.

Структура и объём работы. Бакалаврская работа состоит из введения, 3х разделов, заключения, списка использованных источников и 4х приложений. Общий объем работы — 90 страниц, из них 59 страниц — основное содержание, включая 4 изображения и 11 таблиц, 20 страниц — приложения. В качестве приложения к работе прикладывается CD-диск. Список использованных источников информации содержит 40 наименований.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел. Первый раздел «Игровой искусственный интеллект» посвящен основным сведениям, необходимым для разработки игрового искусственного интеллекта.

В качестве основных концепций, составляющих задачу разработки искусственного интеллекта, берутся такие ключевые понятия из агентно-ориентированного подхода к разработке искусственного интеллекта, как «интеллектуальный агент» и «среда».

Под интеллектуальным агентом понимается сущность, чье поведение может быть представлено как цикл трех событий:

восприятие — получение сенсорных данных;
мышление — обработка полученных данных и принятие решения;
действие — воплощение принятого решения.

Среда задает рамки и условия проектирования агента. В каждый момент времени она может быть представлена своим состоянием, структура которого определяется специально для каждой конкретной задачи. Множество всевозможных состояний среды называется пространством (множеством) состояний. Беря в расчет информацию о состоянии среды, полученной благодаря акту восприятия, агент инициирует процесс мышления и совершает действие, которое в свою очередь изменяет состояние среды.

Немаловажным также является понятие рационального агента: рациональным считается агент, который в любой момент времени, исходя из всех накопленных знаний, выбирает действие, которое, по его мнению, максимизирует его показатели производительности.

Игра по теории игр — конфликтная ситуация, в котором участвуют две и более сторон, целью каждой из которых является реализация своих интересов, причем от реальной конфликтной ситуации игра отличается тем, что ведется по определенным заранее правилам.

В рамках бакалаврской работы для игры как среды были установлены обязательными к выполнению свойства дискретности и мультиагентности.

Теория игр определяет ряд важных понятий, таких как:

Партия. Конкретный пример разыгрывания игры некоторым конкретным образом от начала и до конца.

Ход. Право выбора между различными возможными действиями в конкретном состоянии.

Стратегия. Совокупность правил, определяющих однозначно выбор игрока в зависимости от состояния игры в момент его личного хода.

Выигрыш. Числовое значение, характеризующее результат партии для игрока.

Исход партии. Вектор значений, соответствующих выигрышу, по одному на каждого игрока. Определяется как значение функции выигрыша.

Оптимальной называется стратегия, которая при многократном повторении игры обеспечивает данному игроку максимально возможный средний выигрыш. Агент, следующий оптимальной стратегии, считается рациональным.

Теория игр допускает разную классификацию игр, например:

По наличию коалиций — кооперативная/некооперативная.

По очередности ходов — очередные ходы/последовательные ходы.

По полноте информации — игра с полной информацией/игра с неполной информацией.

По совершенности информации — игра с совершенной информацией/игра с несовершенной информацией.

Второй раздел. Второй раздел «Обучение с подкреплением» посвящен обучению с подкреплением и его алгоритмам.

В обучении с подкреплением для описания взаимодействия обучающегося агента со средой в терминах состояний, действий и вознаграждений используется формализм марковских процессов принятия решений.

В общем случае марковские процессы принятия решений можно представить как кортеж $(\mathcal{S}, \mathcal{A}, R, p)$:

$\mathcal{S}$ Пространство состояний.

$\mathcal{A}$ Множество действий.

$\mathcal{R}$ Множество вознаграждений, $\mathcal{R} \subseteq \mathbb{R}$.

Состояние, действие и вознаграждение в момент t обозначаются как S_t, A_t и R_t.

Функция $p : \mathcal{S} \times \mathcal{R} \times \mathcal{S} \times \mathcal{A} \to [0, 1]$ — функция динамики МППР. Она задает распределение вероятностей возникновения значений $s' \in \mathcal{S}$ и $r \in \mathcal{R}$ в

момент времени t с учетом предыдущего состояния и совершенного действия:

$$p(s', r|a, a) \stackrel{\text{def}}{=} P(S_t = s', R_r = r|S_{t-1} = s, A_{t-1} = a)$$
$$\forall s', s \in \mathcal{S}, r \in \mathcal{R}, a \in \mathcal{A}(s),$$

где $\mathcal{A}(s)$ — множество всех действий, допустимых в состоянии s.

Среди концепций, составляющих системы обучения с подкреплением, выделяются следующие:

— стратегия;
— сигнал вознаграждения;
— функция ценности.

Как и в теории игр, стратегия отвечает за поведение агента в конкретном состоянии в конкретный момент времени. Формально представляет собой отображение пар действий и состояний на вероятность выбора данного действия из этого состояния: $\pi : \mathcal{A} \times \mathcal{S} \to [0, 1]$:

$$\pi(a|s) \stackrel{\text{def}}{=} P(A_t = a|S_t = s).$$

Сигнал вознаграждения характеризует реакцию среды на действия агента. Целью агента является максимизация полного вознаграждения G_t, полученного начиная с момента времени t:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{T} \gamma^k R_{t+k+1},$$

где $\gamma \in [0, 1]$ — фактор дисконтирования, T — конечный момент времени.

В отличие от функции выигрыша, фигурирующей в теории игр, функция вознаграждения возвращает значение после каждого хода агента и характеризует полезность совершенного им действия, в то время как функция выигрыша возвращает значение только по окончанию партии и характеризует ее исход, оценивая таким образом стратегию игрока.

Функция ценности определяет ожидаемое полное вознаграждение при условии следования агентом стратегии π.

Функция ценности состояния s в момент времени t при стратегии π может

быть определена как средний доход в будущем:

$$v_\pi(s) \stackrel{\text{def}}{=} M_\pi[G_t|S_t = s], \qquad s \in \mathcal{S}.$$

Функция ценности действия a в состоянии s при стратегии π может быть определена следующим образом:

$$q_\pi(s, a) \stackrel{\text{def}}{=} M_\pi[G_t|S_t = s, A_t = a], \qquad s \in \mathcal{S}, a \in \mathcal{A}(s).$$

Алгоритмы обучения с подкреплением можно разделить на две группы:

1. Model-based — основанные на модели.
2. Model-free — независимые от модели.

Model-based обучение с подкреплением определяется наличием у агента доступа к модели среды или возможности ее изучения. В то же время model-free алгоритмы не требуют подобных знаний и основаны на процессе оптимизации стратегии и иных параметров.

Среди model-free алгоритмов принято выделять три подхода:

— value-based;
— policy-based;
— actor-critic.

Обучение value-based алгоритмов заключается в поиске наилучшей оценки функций ценности, значения которых характеризуют ценность того или иного состояния или действия. Стратегия агента в этом случае представляет собой фиксированное правило выбора действия. Policy-based алгоритмы не требуют оценки функций ценности и используют параметризированную стратегию, а их обучение заключается в выборе целевой функции и ее оптимизации с помощью метода градиентного подъема, применяемого к параметрам стратегии. Для алгоритмов категории actor-critic, или исполнитель-критик, определены два обучаемых компонента, взаимодействующих друг с другом: исполнитель и критик. Первый действует в соответствии со стратегией, параметры которой он обновляет согласно сигналам от критика. Критик помимо этого также отвечает за оптимизацию функции ценности.

Мультиагентные среды в зависимости от функции вознаграждения можно разделить на следующие группы:

1. Полностью кооперативные.

2. Полностью некооперативные.
3. Гибридные.

В полностью кооперативных средах агенты делят одну общую награду. В полностью некооперативных, как в играх с нулевой суммой, общая сумма итоговых вознаграждений равняется нулю, а целью агентов является максимизация своего собственного дохода. Гибридные среды сочетают оба типа наград.

Помимо этой особенности, свойство мультиагентности также накладывает некоторые ограничения на среду, алгоритмы и процесс обучения.

Третий раздел. Третий раздел «Программная реализация» посвящен реализации программы для проведения экспериментов над алгоритмами обучения с подкреплением в игровых средах.

В таблице 1 приведены характеристики алгоритмов, программно реализованных в рамках бакалаврской работы.

Алгоритм	Опыт	Подход		Нейронная сеть
		Value-based	Policy-based	
Q-Learning	Актуальный	+		
REINFORCE	Актуальный		+	+
DQN	Отложенный	+		+
A2C	Отложенный	+	+	+

Таблица 1 – Характеристики алгоритмов

В таблице 2 приведен список игр, использующихся в качестве сред для экспериментов в реализованной программе, и их классификация по теории игр.

Реализованная программа предоставляет следующий функционал:

Добавление новых агентов Все агенты, следующему одному из алгоритмов обучения с подкреплением, наследуют и реализуют методы класса Agent.

Добавление новых игр Для добавления игры, выступающей в качестве среды для проведения экспериментов над алгоритмами, необходимо создать класс, наследующий от класса Game.

Проведение экспериментов За проведение экспериментов отвечает класс Experiment.

Сбор статистики и вывод результатов Для работы со статистикой существует отдельный класс StatisticsLogger.

	Уно	Шашки	Пинг-понг	Крестики-нолики	Камень-ножницы-бумага
Полнота информации	-	+	-	+	+
Совершенность информации	+	+	-	+	-
Одновременные ходы	-	-	+	-	+
Кооперативная	-	-	+	-	-

Таблица 2 – Классификации используемых в качестве сред игр

По окончанию каждого эксперимента, заключающегося в проведении n игр, в течение которых агенты обучаются, генерируются разные графики и текстовый отчет, который содержит информацию об итоговых суммах очков обоих агентов, средних и максимальных вознаграждениях за эпизод, а также о затраченном времени.

Для реализации использовался язык программирования Python версии 3.10 и библиотека PettingZoo. PettingZoo предоставляет средства взаимодействия со средой, а также сами среды, реализованные в виде отдельных модулей, которые использовались в данной работе. PettingZoo в своей основе содержит OpenAI Gym — библиотеку, предоставляющую API для разработки алгоритмов обучения с подкреплением. При реализации алгоритмов, использующих нейронные сети, использовалась библиотека PyTorch.

В реализованной программе были проведены эксперименты длительностью в 1000 эпизодов. В таблице 3 представлен рейтинги агентов в соревновательных играх (победа засчитывается, если агент по итогу эксперимента набирает большее число очков, чем оппонент).

	Уно	Шашки	Крестики-нолики	Камень-ножницы-бумага	Итоговое число побед
Q-Learning	1	–	**4**	1	6
REINFORCE	3	2	3	**4**	12
A2C	**5**	1	3	**4**	**13**
DQN	3	**3**	2	3	11

Таблица 3 – Рейтинг агентов

По итогам экспериментов был сформирован ряд наблюдений и выводов о применении алгоритмов обучения с подкреплением в игровых средах, например:

— В играх с большими пространствами состояний или действий, как шашки или уно, алгоритм Q-Learning показывает себя хуже остальных в плане производительности, или вовсе неприменим, потому что при обучении агент опирается на таблицу, в которой хранит значения функции ценности для всех пар состояний и действий. Этого лишены остальные три алгоритма, которые выбирают действия по предсказаниям нейронной сети, лежащей в их основе. Однако они за счет этого показывают более плохие результаты в играх, в которых возможно наличие недопустимых действий, поскольку архитектура их нейросетей не подразумевает существование дополнительного входного слоя.
— Policy-based алгоритмы сильно зависят от начального выбора стратегии, что делает их менее стабильными в играх с большой продолжительностью. Это связано с тем, что они проводят обновление либо с определенной частотой, либо в конце эпизода.
— Порядок совершения ходов сильно влияет на результаты обучения, особенно в небольших играх вроде крестиков-ноликов.

ЗАКЛЮЧЕНИЕ

Для достижения поставленной цели было выполнено следующее:

— проведено исследование на тему взаимосвязи между областями искусственного интеллекта, теории игр и обучения с подкреплением;
— выявлены основные концепции и подходы, применяемые в обучении с подкреплением;
— реализованы агенты, действующие и обучающиеся согласно алгоритмам Q-Learning, REINFORCE, Advantage Actor-Critic и Deep Q-Network;
— адаптированы под среды для экспериментов игры, отличающиеся друг от друга уникальными наборами характеристик: «Уно», «Шашки», «Пинг-понг», «Крестики-нолики», «Камень-ножницы-бумага»;
— реализован модуль для проведения экспериментов по обучению алгоритмов обучения с подкреплением в игровых средах;
— реализована система сбора, обработки и визуализации статистики по проведенным экспериментам;
— реализована программная среда, обеспечивающая взаимодействие между перечисленными компонентами, а также поддерживающая добавление новых алгоритмов и игр;
— проведены эксперименты над реализованными алгоритмами в перечисленных играх и по их итогам сформированы наблюдения и выводы о применении алгоритмов обучения с подкреплением для задачи разработки игрового интеллекта.

Основные источники информации

1. *Рассел, С.* Искусственный интеллект: современный подход. 2-ое издание / С. Рассел, П. Норвиг. — Москва: Издательский дом «Вильямс», 2007.
2. *фон Нейман, Д.* Теория игр и экономическое поведение / Д. фон Нейман, О. Моргенштерн. — Москва: Наука, 1970.
3. *Вентцель, Е. С.* Элементы теории игр. 2-ое издание / Е. С. Вентцель. — Москва: Государственное издательство физико-математической литературы, 1961.
4. *Саттон, Р. С.* Обучение с подкреплением: Введение. 2-е изд. / Р. С. Саттон, Э. Д. Барто. — Москва: ДМК Пресс, 2020.
5. *Canese, L.* Multi-agent reinforcement learning: A review of challenges and ap-

plications / L. Canese, G. C. Cardarilli, L. Di Nunzio, R. Fazzolari, D. Giardino, M. Re, S. Span // *Applied Sciences*. — 2021. — Vol. 11, no. 11.

6. *Алфимцев, А. Н.* Мультиагентное обучение с подкреплением: учебное пособие / А. Н. Алфимцев. — Москва: Издательство МГТУ им. Н. Э. Баумана, 2021.