Практическая работа № 4

«Обучение с подкреплением без модели»

по дисциплине «Проектирование интеллектуальных систем»

Цели: приобрести навыки обучения с подкреплением (ОСП) интеллектуальных агентов систем искусственного интеллекта с помощью безмодельных методов обучения.

Задачи:

1) Выполнить обучение интеллектуального(-ых) агента(-ов) автоматизированной интеллектуальной системы с помощью безмодельных (иногда – безмодальных, model-free, **см. Примечание 1**) алгоритмов ОСП (Reinforcement Learning, RL), выполнив следующие подзадачи:

– как и в практической работе №3 определить предметную область решаемой задачи (может совпадать с задачами из предыдущих практических работ и лучше всего из работы №3, либо из курса РСППР), таковыми могут выступать популярное обучение агентов компьютерных игр (стабилизация стержня / палки / перевернутого маятника, прохождение агентом лабиринта, игры-платформера, гонок с обгоном и без, настольный теннис и т.д.), задачи оптимизации (поиск оптимумов, обучение нейронной сети, поиск решений уравнений или задач), обучение многоагентной системы (симуляция жизни или других процессов, многоагентная оптимизация, прогнозирование), машинное обучение с подкреплением;

– определить обучаемого(-ых) агента(-ов) (ими могут выступать нейронные сети, агенты многоагентной системы, алгоритмы машинного обучения, персонажи игры и т.д.), определить среду системы (опять же марковским процессом принятия решений, его модификациями и MCTS), определить целевую функцию (или целевое состояние) системы и метрики качества модели (ошибка, точность);

– реализовать безмодельный метод обучения с подкреплением, основанный на ценностно-ориентированном (value-based) и/или основанном

на политике (policy-based) подходах вроде Q-Learning, Deep Q-Learning (DQN), SARSA, Vanilla Policy Gradient (VPG), Trust Region Policy Optimization (TRPO), Deep Deterministic Policy Gradients (DDPG), Soft Actor-Critic (SAC-2018 и SAC-2019) и т.д. (**опять см. Примечание 1**), для реализации алгоритмов можно использовать репозитории вроде dennybritz/reinforcement-learning и TianhongDai/reinforcement-learning-algorithms.

2) В качестве дополнительного задания реализовать хотя бы один из следующих пунктов:

– выполнить многоагентное обучение с подкреплением (multi-agent reinforcement learning, MARL, **см. Примечание 2**), если хотя бы у нескольких агентов будут разные методы обучения (лучше всего – все разного вида: с моделью, без модели и бандиты), то автоматически засчитывается и пункт о гибридной системе;

– выполнить нечеткое обучение с подкреплением (**см. Примечание 3**);

– создание гибридной системы (обучения с подкреплением несколькими методами или/и обучение нескольких моделей);

– как и в практической работе № 3, провести сравнение работы систем (на основе 3-ёх главных критериев – время, память и качество работы), реализованных на основе обучения с учителем и без учителя, с системами, реализованных с ОСП;

– сравнить по тем же критериям результаты работы безмодельных методов с модельными и/или с бандитами (из практической работы №3).

ПРИМЕЧАНИЕ:

1) Безмодельные методы ОСП (model-free reinforcement learning) опираются не на модель среды (что следует из их названия), а на последствия своих действий (так как модель может быть предвзятой или неправильной и тем самым не приводить агентов даже к хотя бы немного выгодному решению), то есть агенты в таких алгоритмах не разрабатывают и не используют внутреннюю модель среды и ее динамики (как правила, но существуют гибридные решения вроде того же Dyna-Q), вместо этого в среде используется метод проб и ошибок, чтобы оценить и отметить возможные пары «действие – состояние», а также последовательности пар «действие – состояние» для разработки политики, которая по сути представляют собой определённую карту возможных действий, такой алгоритм будет выполнять действие несколько раз и корректировать политику для оптимального вознаграждения, основанного на результатах своей деятельности. Безмодельные методы ОСП принято разделять на следующие группы:

а) Основанные на политике – алгоритмы, цель которых состоит в том, чтобы оптимизировать политику $\pi(a|s)$ для максимизации ожидаемого совокупного вознаграждения, где a (action) – действие, s (state) – состояние, при этом политика может быть как детерминированной (определённой, Deterministic Policy), так и стохастической (Stochastic Policy), в первом случае политика с уверенностью отображает каждое состояние на одно действие (другими словами, агент всегда будет предпринимать одно и то же действие в данном конкретном состоянии), во втором случае политика отображает каждое состояние на вероятностное распределение действий (другими словами в данном конкретном состоянии агент будет выбирать действие случайным образом на основе определённого закона распределения вероятностей). Наиболее известными алгоритмами, основанными на политике, являются градиент-политика (policy gradient), который работает путем обновления параметров политики посредством

стохастического градиента повышения эффективности политики (что обеспечивает повышение вероятности действий, которые приводят к более высокому вознаграждению, и понижении вероятностей действий, которые приводят к более низкому вознаграждению), а также его модификации (Monte-Carlo Policy Gradient и т.п.), не менее известный метод Актор-Критик (Actor-Critic), в котором структура политики называется актором, поскольку она используется для выбора действий, а оценочная функция вознаграждения называется критиком, поскольку она критикует действия, совершаемые актором, обучение всегда направлено на политику: критик должен узнать и раскритиковать ту политику, которой в данный момент придерживается актор, также известно множество модификаций этого метода (Advantage Actor Critic – A2C и Asynchronous Advantage Actor-Critic – A3C, Normalized Actor-Critic – NAC, Actor Critic with Experience Replay – ACER, Off-Policy Actor-Critic – Off-PAC) и метод Proximal Policy Optimization (PPO).

b) Ценностно-ориентированные – алгоритмы, которые пытаются изучить оптимальную функцию вознаграждения, которая оценивает ожидаемое совокупное вознаграждение для каждого состояния или пары состояние-действие, что позволяет косвенно определить оптимальную политику, которой должен следовать агент, что отличает от предыдущей группы, где напрямую оптимизируется сама политика. В общем виде функция вознаграждения $V_\pi(s)$ определяется как ожидаемый доход, начиная с состояния $s_0 = s$ и последовательно следуя политике π, грубо говоря, функция ценности определяет, «насколько хорошо» находиться в данном состоянии. Наиболее известными представителями данной группы являются Q-метод (Q-learning, также Q-table), который использует уравнение Беллмана (оно определяет значение $Q(s|a)$ для определенной пары состояние-действие) для построения и поиска значений Q-таблицы, в которой столбцы — действия, а строки — состояния, Q-сеть (Q-Network), глубокая Q-сеть (DQN) и её

модификации, State-Action-Reward-State-Action (SARSA), Hindsight Experience Replay (HER), Prioritized experience replay (PER) и C51.

c) Гибридные безмодельные алгоритмы – алгоритмы, сочитающие в себе элементы как методов, основанных на политике, так основанных на функции вознаграждения, к ним относятся, например, глубокий детерминированный политический градиент (Deep Deterministic Policy Gradient, DDPG и его модификация DDPG from Demonstration), который одновременно изучает Q-функцию и политику, он использует данные, не относящиеся к политике, и уравнение Беллмана для изучения Q-функции, а саму Q-функцию использует уже для изучения политики, что реализовано для обеспечения применения метода в средах с непрерывным пространством действий, особого внимания стоит модификация с двойной Q-функцией DDPG – Twin Delayed DDPG (TD3), которая позволяет уменьшить вероятность переоценки Q-функции, выбирая наихудшую из пары, также популярен лёгкий метод актор-критик (Soft Actor Critic, SAC), который оптимизирует стохастическую политику путём без политики, образуя мост между оптимизацией стохастической политики и подходами в стиле DDPG, в этом методе политика обучается максимизировать компромисс между ожидаемым вознаграждением и энтропией (мерой случайности в политике, агент получает бонусное вознаграждение на каждом временном шаге, пропорциональное энтропии политики на этом временном шаге).

Стоит отметить, что существуют и другие виды обучения с подкреплением вроде инверсное обучение с подкреплением (inverse reinforcement learning, IRL), в котором функция вознаграждения не задается, а вместо этого она выводится на основе наблюдаемого поведения эксперта, безопасное обучение с подкреплением (safe reinforcement learning, SRL), в котором выполняется обучение политикам, которые максимизируют ожидаемую отдачу в проблемах, в

которых важно обеспечить разумную производительность системы и/или соблюсти ограничения безопасности, состязательное обучение с глубоким подкреплением (adversarial deep reinforcement learning, ADRL) и другие возможные модели и подходы, однако они на сегодняшний являются теоретическими или в лучшем случае **экспериментальными**, поэтому **не являются предметом настоящей практики**.

2) Многоагентное обучение с подкреплением (multi-agents reinforcement learning, MARL) подразумевает систему обучения, в которой несколько агентов совместно взаимодействуют в общей среде, это взаимодействие может иметь разный характер, в основном выделяют кооперативные среды, в которых агенты пытаются максимизировать общее вознаграждение (например, роботы на заводе или складе, строительная бригада и т.д.), конкурентные среды, в которых каждый агент стремится максимизировать лишь собственную прибыль (например, «царь горы», любая игра один на один, шахматы, гонки и т.д.), и смешанные среды, в которых агенты могут как конкурировать, так и сотрудничать для увеличения общего вознаграждения в конкретный момент времени, если в одиночку они могут проиграть (примерами могут выступать любые командные спортивные и интеллектуальные игры от футбола, до настольных ролевых игр). В таком виде обучения используются все теоретические и практические наработки теории многоагентных систем (BDI-модель, основные свойства и типы агентов, **см. практическую работу №1 первого семестра РСППР**). Применение разных алгоритмов и даже видов ОСП на каждом отдельном агенте или их группах в таких системах может позволить преодолеть проблемы предвзятости моделей и неоптимальности политик, а также расширить область многоагентных систем и породить новые устойчивые алгоритмы обучения с подкреплением.

3) Нечёткое обучение с подкреплением (Fuzzy reinforcement learning, FRL) путем введения нечеткого вывода (например, с помощью правил вывода Мамдани, Сугено, Заде, нечёткий Modus Ponens и т.д.) позволяет аппроксимировать функцию вознаграждения «состояния-действия»

нечеткими правилами в непрерывном пространстве, а форма нечетких продукционных правил (правил «ЕСЛИ-ТО») делает этот подход пригодным для выражения результатов в форме, близкой к естественному языку. Примером такого алгоритма может быть нечёткое Q-обучение (Fuzzy Q-Learning) и нечёткий алгоритм JAYA (JAYA-Optimized Fuzzy Reinforcement Learning algorithm). Главным же преимуществом такого вида ОСП то, что предварительные знания предметной области, которые могут быть очень приблизительными и неточными (то есть сама постановка задачи может быть очень расплывчатой и звучать как «через некоторое время робот должен находится недалеко от двери»), могут быть выражены в терминах нечетких правил и уточнены позже в процессе обучения.