ФАКУЛЬТЕТ РАДИОФИЗИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ

ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ МУЛЬТИАГЕНТНЫХ СИСТЕМ

К. А. Акула

Белорусский государственный университет, Минск
Ksenea1@gmail.com
науч. рук. – А. В. Сидоренко, д-р техн. наук, проф.

Исследуются способы машинного обучения с подкреплением. Охарактеризованы алгоритмы на основе метода временных различий для обучения мультиагентных систем. Для проведения вычислительного эксперимента для трех алгоритмов – SARSA, Q-learning и Deep-Q-learning – разработаны компьютерные программы. Проведен сравнительный анализ применения указанных алгоритмов по критерию значения вознаграждения в зависимости от числа итераций, что позволяет определить оптимальный алгоритм для обучения с подкреплением мультиагентных систем.

Ключевые слова: мультиагентные системы; обучение; алгоритмы; обучение с подкреплением.

МЕТОДЫ ВРЕМЕННЫХ РАЗЛИЧИЙ

Алгоритмы, основанные на методах временных различий (Temporal difference learning), позволяют агенту осуществлять обучение после каждого временного шага (отдельного действия) и обновлять информацию о состоянии агента. Оценка вознаграждения на каждом временном шаге (действии) определяется так:

$$s_t \leftarrow s_{t-1} + \alpha\,[Target\text{-}\ s_{t-1}],$$

где *Target* – целевая ошибка, s_t – состояние в момент времени t, α – скорость обучения.

Приведенное выражение помогает достичь цели при обучении путем обновления на каждом временном шаге.

Target определяется следующим образом:

$$\text{Target} = E_\pi\left[\sum_{k=0}^{\infty}\gamma^k r_{t+k+1}\right],$$

где γ – дисконтирующий множитель, r – значение вознаграждения, α и γ – параметры алгоритма для Q-обучения. Значения обоих параметров лежат в диапазоне от 0 до 1. Значение γ определяет влияние, которое придается вознаграждению, оно может быть равным нулю. Параметр α, ко-

торый определяется при положительной скорости обучения, учитывая, что при обновлении значение потерь должно компенсироваться, не может равняться нулю.

В процессе обучения агент не знает о состоянии, вознаграждении и переходах. Он взаимодействует со средой (совершает случайные или информированные действия) и сохраняет информацию о новых параметрах (значениях пар состояния-действия), постоянно обновляя имеющиеся данные после выполнения каждого действия. Для обучения агентов мультиагентной системы при перемещениях в различных направлениях используются алгоритмы обучения с подкреплением.

Для этого используют алгоритмы управления SARSA и Q-Learning.

АЛГОРИТМ УПРАВЛЕНИЯ SARSA

SARSA (State-Action-Reward-State-Action) – это алгоритм управления, основанный на методе временных различий, с использованием политики. Политика – это набор пар «состояние-действие». Политика отображает действия, которые должны быть предприняты агентом в каждом состоянии. Метод управления на основе политики выбирает действие для каждого состояния агента во время обучения. При этом целью является оценить *Q(s, a)* для текущей политики π и всех пар «состояние-действие» (*s*-*a*) [1].

Q-значение (*Q(s, a)*) – это отображение между парой «состояние-действие» и действительным числом, обозначающим его полезность. Алгоритм SARSA представляет собой алгоритм, совершающий оценку оптимального Q-значения для всех пар «состояние-действие», обновляющееся согласно выражению:

$$Q(s_{t+1}, a_{t+1}) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)].$$

Если состояние S является терминальным (состояние цели или конечное состояние), то $Q(S, a) = 0$, $a \subset A$, где A – множество всех возможных действий.

Агент может проходить через одни и те же состояния снова и снова, экспериментируя с различными действиями, пока он не сможет определить, какие действия являются лучшими при некоторых состояниях.

АЛГОРИТМ УПРАВЛЕНИЯ Q-LEARNING

Алгоритм Q-Learning, в отличие от алгоритма SARSA, представляет собой алгоритм управления, основанный на методе временных различий, без использования политики. Агент получает вознаграждение, совершая в конкретном состоянии наиболее оптимальное действие. Опираясь на

таблицу вознаграждений, он выбирает следующее действие в зависимости от того, насколько оно полезно, а затем обновляет величину, именуемую Q-значением. В результате создается новая таблица, называемая Q-таблицей. Если Q-значения оказываются больше, то мы получаем более оптимизированные вознаграждения.

Q-величины инициализируются случайными значениями, и по мере того, как агент взаимодействует со средой и получает различные вознаграждения, совершая те или иные действия, Q-значения обновляются на каждом временном шаге согласно выражению:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)].$$

Как и в случае с алгоритмом SARSA, целью является оценить значения Q, но алгоритм Q-Learning определяет действие a', взяв за него действие a для которого Q-значения равно $\max_a Q$ [1].

АЛГОРИТМ DEEP Q-LEARNING

Алгоритм DEEP Q-LEARNING основан на использовании нейронных сетей при обучении с подкреплением мультиагентных систем. Нейронная сеть используется для аппроксимации функции значения или функции политики. То есть нейронные сети могут научиться отображать состояния в значения или пары «состояние-действие» в значения Q. Вместо того чтобы использовать таблицу поиска для хранения, индексирования и обновления всех возможных состояний и их значений, что невозможно при очень больших объемах данных, мы можем обучить нейронную сеть на выборках из состояния или пространства действий, чтобы научиться прогнозировать, насколько они ценны по отношению к обучению с подкреплением.

Как и все нейронные сети, они используют коэффициенты для аппроксимации функции, связывающей входы с выходами, и их обучение заключается в поиске правильных коэффициентов или весов путем итеративной корректировки этих весов вдоль градиентов, которые обещают меньшую ошибку [2].

ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ

Разработанные нами компьютерные программы позволили нам провести вычислительный эксперимент по обучению с подкреплением мультиагентных систем. На рисунке приведены результаты вычислительного эксперимента, проведенного нами при обучении агента с применением алгоритмов SARSA, Q-Learning и Deep Q-Learning с использованием в качестве критерия обучения значения вознаграждения при разном числе итераций.

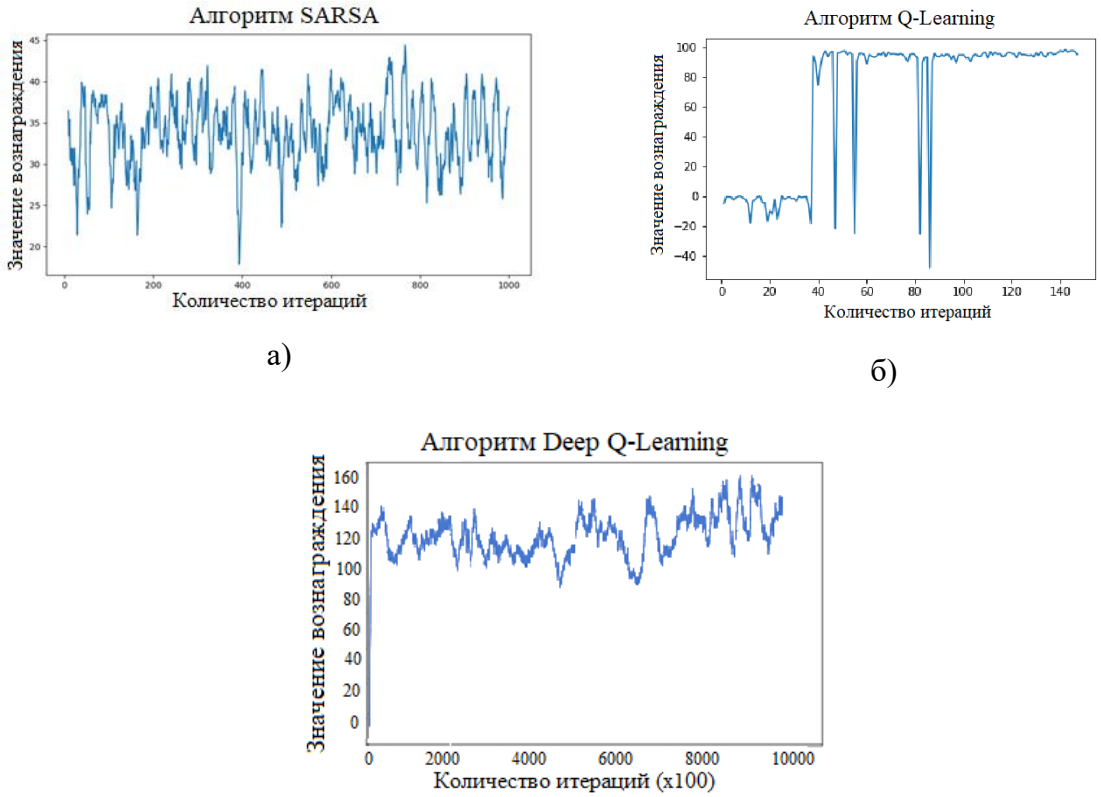

Рис. 1. График зависимости значение вознаграждения от количества итераций при реализации различных алгоритмов: а) SARSA, б) Q-Learning, в) Deep Q-Learning

ЗАКЛЮЧЕНИЕ

Изучены основные алгоритмы, основанные на методе временных различий, для обучения мультиагентной системы и программно реализованы алгоритмы обучения с подкреплением, а также проведен вычислительный эксперимент по обучению с подкреплением мультиагентной системы.

Анализ результатов вычислительного эксперимента по обучению с подкреплением мультиагентной системы показал, что по критерию вознаграждения наиболее эффективным является алгоритм Deep Q-Learning, обладающий наибольшей производительностью.

Библиографические ссылки

1. *Jens Kober, J. Andrew Bagnell, Jan Peters* Reinforcement Learning in Robotics a Survey// The International Journal of Robotics Research. 2013. V 32. № 11. P. 1238-1274. DOI: 10.1177/0278364913495721.
2. *Thanh Thi Nguyen, Ngoc Duy Nguyen, Saeid Nahavandi* Deep Reinforcement Learning for Multi-Agent Systems: A Review of Challenges, Solutions and Applications//IEEE Transactions on Cybernetics. 2020. VI. P. 1 – 14. DOI:10.1109/TCYB.2020.2977374.