УДК 004.042

ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ ПРИ НАВИГАЦИИ МОБИЛЬНЫХ РОБОТОВ

д-р техн. наук, проф. А. В. СИДОРЕНКО
(Белорусский государственный университет, Минск)

Предложен новый алгоритм машинного обучения для навигации мобильных роботов, основанный на комбинации методов Deep-Q-Learning и Double Q-Learning. Предложенная модель рассматривается при перемещении робота в некоторой среде (среда задается программным пакетом Gazebo) и известном его местоположении с учетом огибания встречающихся на пути препятствий. В качестве программного обеспечения используются Mobile Robotics Simulation Toolbox и Gazebo. При тестировании показано, что новый алгоритм более чем в десять раз улучшает временные параметры выполнения задачи по сравнению с традиционными алгоритмами. Представленный алгоритм может быть интегрирован в аппаратуру.

Ключевые слова: *алгоритм, робот, управление, движение, машинное обучение.*

Введение. При внедрении мобильных роботов в различные сферы деятельности человека одной из актуальных проблем является управление движением группы роботов. При решении подобной задачи такая группа мобильных роботов может рассматриваться как мультиагентная система, состоящая из множества взаимодействующих агентов в сфере искусственного интеллекта [1]. Под агентом понимают устройство, обладающее искусственным интеллектом.

При внедрении мобильных роботов в космическую, производственную сферы проблема управления их движением в некоторой среде сводится к обеспечению безопасного движения робота без столкновения со встречающимися на его пути препятствиями.

При решении подобных задач используются алгоритмы обучения с подкреплением, нейросетевые алгоритмы, алгоритмы глубокого обучения [2; 3]. В основу применения таких алгоритмов положены принципы моделирования, а критерием оптимизации при этом является определение вознаграждения в зависимости от числа производимых при перемещении итераций.

В данной работе представлен алгоритм управления безопасным движением робота, основанный на сочетании Deep_Q-Learning и двойного Q-обучения.

1. Методы машинного обучения. Среди алгоритмов управления на основе машинного обучения рассмотрим алгоритм управления Q-Learning, алгоритмы глубокого обучения и двойного Q-обучения.

Алгоритм управления Q-Learning представляет собой метод, используемый при машинном обучении в сфере искусственного интеллекта при мультиагентном подходе. На основе полученного от среды вознаграждения агент формирует функцию полезности Q, что впоследствии дает ему возможность уже случайным образом выбирать стратегию поведения, а также учитывать опыт предыдущего взаимодействия со средой.

Алгоритм управления Q-Learning позволяет агенту получить вознаграждение, совершая в конкретном состоянии наиболее оптимальное действие. Опираясь на таблицу вознаграждений, он позволяет выбрать следующее действие в зависимости от того, насколько оно полезно и дает возможность агенту обновить величину, называемую Q-значением. Q-величины инициализируются случайными величинами, которые обновляются согласно выражению

$$Q(s_i, a_i) \leftarrow Q(s_i, a_i) + \alpha\left[r_{i+1} + \gamma \max_a Q\right], \quad (1)$$

где a_i и s_i – соответственно действие и состояние агента в момент времени t;

α и γ – соответственно скорость обучения и дисконтирующий множитель, параметры которых находятся в области [0, 1];

r – значение вознаграждения.

В результате создается новая таблица, называема Q-таблицей, в которой хранится информация о состоянии и действии агента.

Глубокое обучение используется при более основательном подходе. В процесс Q-обучения вводится аппроксимация функции. При этом одним из решений в качестве аппроксимации функций является применение нейронной сети [3]. Нейронная сеть может использоваться для аппроксимации функции значения или пары: действие–состояние в значении Q. Мы можем обучить нейронную сеть на выборках из состояния или пространства действия, чтобы научиться прогнозировать, насколько они ценны по отношению к цели обучения с подкреплением. В обучении с подкреплением можно использовать сверточные сети (convolution neural network, CNN). Как правило, используется глубокая сверточная нейронная сеть со слоями мозаичных сверточных фильтров для имитации рецептивных полей. Структура нейронной сети позволяет

эффективно распознавать изображение и состояние агента, когда используется визуальный ввод или робот находится на местности. Точность распознавания таких сетей превосходит обычные нейронные сети на 10–15%. Сверточные нейронные сети являются ключевой технологией глубокого обучения. Однако если для представления значения Q используется нейронная сеть, то обучение с подкреплением может быть нестабильным. Это обусловлено необходимостью постоянного проведения коррекций в последовательности наблюдений и снижением корреляции с целевым значением Q.

Двойное обучение характеризуется тем, что в алгоритмах Q-обучения приближенные значения действия и текущая политика выбора действий разделены и целесообразно использовать две отдельных функций значения ценности Q. В зашумленной среде это может замедлить процесс обучения. Для исключения этого предложен вариант под названием Double-Q-Learning (Двойное Q-обучение), в котором оценка Q используется для выбора последующего действия [4]. Практически с использованием разных опытов симметрично друг другу обучаются две отдельные функции значений ценности Q. Этап обновления двойного Q-обучения выглядит следующим образом:

$$Q_{i+1}^{A}\left(s_i,a_i\right)=Q_i^{A}(s_ia_i+\alpha_i\left(s_i,a_i\right)(r_i+\gamma Q_i^{A}\left(s_{i+1,}\arg_a\max Q_i^{A}\left(s_{i+1,}a\right)-Q_i^{A}\left(s_i,a_i\right)\right),$$
$$Q_{i+1}^{B}\left(s_i,a_i\right)=Q_i^{B}\left(s_ia_i\right)+\alpha\left(s_i,a_i\right)(r_i+\gamma Q_i^{B}\left(s_{i+1,}\arg_a\max Q_i^{B}\left(s_{i+1,}a\right)-Q_i^{B}\left(s_i,a_i\right)\right). \quad (3)$$

При таком подходе исключается проблема возможного завышения ценности Q. Модификация алгоритма с глубоким обучением позволила создать новый алгоритм Double DQN (двойного Q-обучения), который превосходит исходный алгоритм DQN.

2. Алгоритм, основанный на сочетании Deep Q-Learning и двойного обучения. В процессе выполнения данной работы нами разработан алгоритм управления, основанный на сочетании Deep Q-Learning и двойного Q-обучения. Для разделения работы по выбору оптимального действия и оценки оптимального действия используются рекуррентные нейронные сети. Архитектура модели алгоритма управления, основанного на сочетании Deep Q-Learning и двойного Q-обучения, приведена на рисунке 1.

В предложенном алгоритме для определения состояния s^1 робота в момент времени $t+1$ входные данные, соответствующие предыдущему состоянию s и действию a в момент времени t при получении вознаграждения r, извлекаются из модуля памяти D. Входные данные отправляются в блок, характеризующий выполнение действия и называемый основной сетью (рисунок 2), а также в блок, характеризующий достижение цели и называемый целевым (рисунок 3).

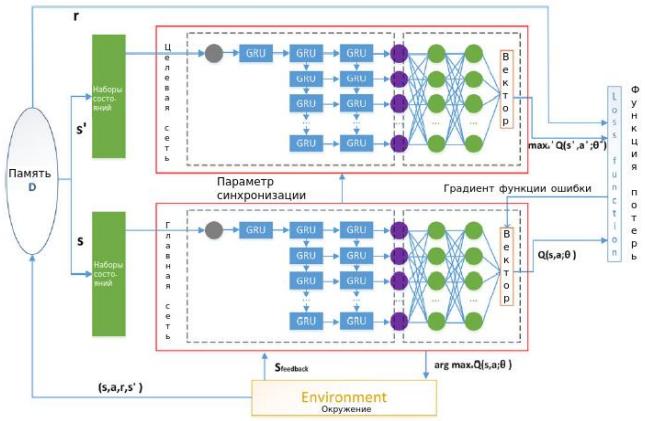


Рисунок 1. – Архитектура модели алгоритма управления, основанного на сочетании Deep Q-Learning и двойного обучения

Основная и целевая сети синхронизируются в реальном времени, а параметры сети остаются такими же.

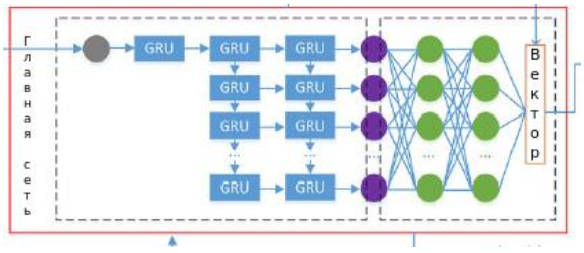


Рисунок 2. – Структура основной сети

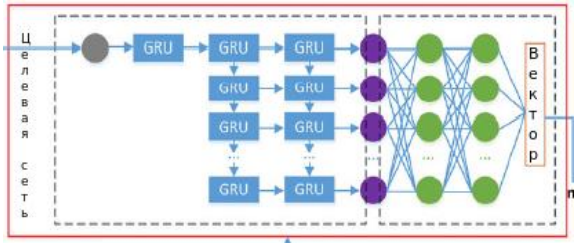
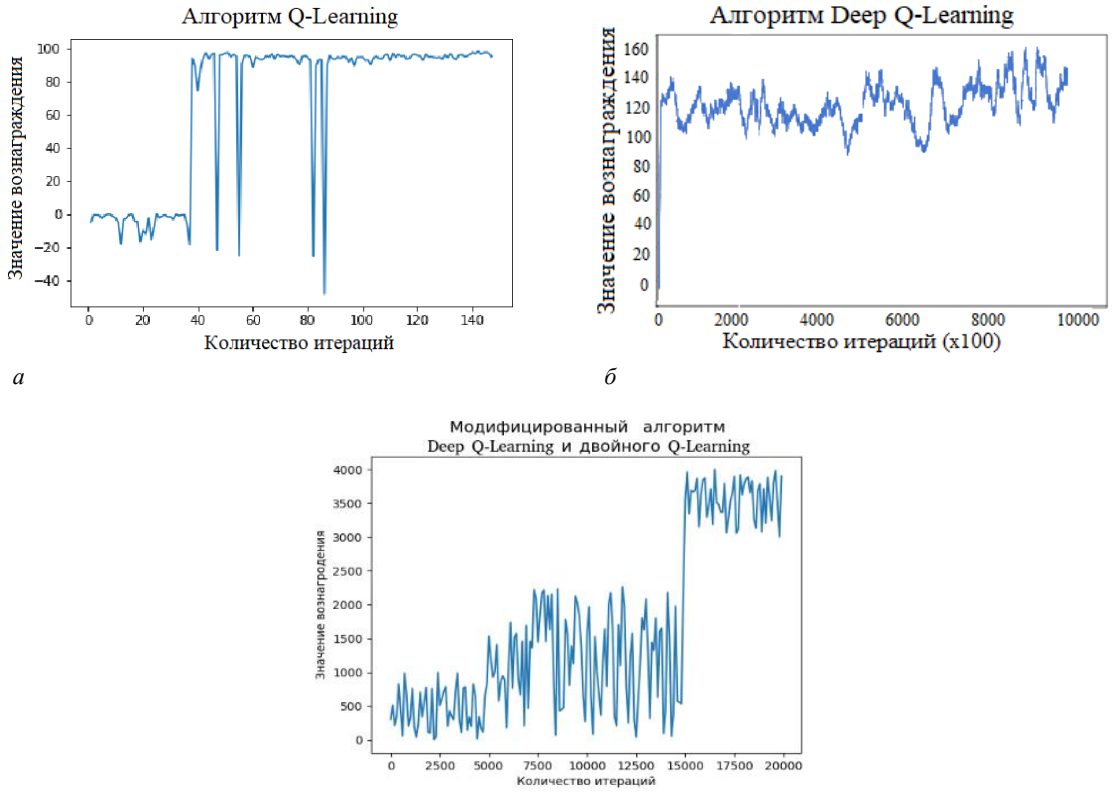


Рисунок 3. – Структура целевой сети

Данные о состоянии на входе должны быть направлены в блок GRU. Указанный блок состоит из восьми ячеек. Это означает, что в наборе поведений есть восемь состояний. Набор поведения A включает перемещения: вперед, назад, влево, вправо, влево вперед, вправо вперед, влево назад, вправо назад. Данные восьми состояний обрабатываются тремя уровнями GRU, а средние значения данных отправляются на уровень FC (fully connected hidden layer). Структура параметров уровня FC представляет собой матрицу, включающую 8, 64, 64, 8 членов соответственно. Функция активации в нейронной сети реализована при помощи линейного блока. Во избежание переобучения при оптимальных параметрах состояния на уровнях GRU и FC соответственно устанавливается структура исключения. Это означает, что если достигнуты оптимальные параметры состояния, то дальнейшее обучение прекращается. Для учета потерь используется двойное глубокое обучение DDQN. Сначала определяется оптимальное действие в основной сети, а затем определяется действие в целевой сети. Дальнейшее обучение для данных производится согласно описанию исходного алгоритма.

3. Проведение вычислительного эксперимента. Программно реализованные алгоритмы обучения, примененные к разработанной нами модели управления системы мобильных роботов, позволили провести вычислительный эксперимент. При моделировании используемая нами модель входит в состав блоков пакета Mobile Robotics Toolbox [5]. В модели, описывающей движение группы роботов, применяется пакет Mobile Robotics Simulation Toolbox на операционной системе Linux при использовании пакета визуализации Gazebo. Взаимодействие агентов обеспечивается через пакет для Matlab ROS Toolbox. Пакет Mobile Robotics Simulation Toolbox поддерживает генерацию кода C++, что позволяет создавать узлы ROS непосредственно из Simulink-моделей в режиме реального времени. Для моделирования среды использовалась библиотека на языке Pyton, в которой имеется несколько встроенных сред. Каждая из этих сред представляет собой RGB-изображение экрана, в котором реализуется массив формы (210, 160, 3). Каждое действие многократно выполняется в течение продолжительности *k* кадров, где *k* равномерно выбирается из (2, 3, 4).

При выполнении работы мультиагентная система проходила оптимальный путь следования при использовании для обучения разработанного нами алгоритма, основанного на сочетании Deep Q-Learning и двойного обучения с применением в качестве критерия обучения значения вознаграждения при разном числе итераций. Для каждого из рассмотренных алгоритмов проведено тестирование, заключающееся в определении положительно проведенных испытаний для системы мобильных роботов, благополучно перешедших из начального состояния в конечное (целевое) без столкновений и с преодолением препятствий. Результаты вычислительного эксперимента приведены на рисунке 4.



а *б* *в*

Рисунок 4. – Зависимости значений вознаграждений от количества итераций при реализации алгоритмов: Q-Learning, Deep Q-Learning и алгоритм, основанный на сочетании Deep Q-Learning и двойного обучения

Заключение. Анализ полученных результатов показал, что среднее количество действий, необходимых для достижения целевого состояния, минимально для предложенного нами алгоритма, основанного на сочетании Deep Q-Learning и двойного обучения. Производительность предложенного алгоритма более чем в десять раз превышает результаты других алгоритмов. Алгоритм может быть интегрирован в аппаратуру.

ЛИТЕРАТУРА

1. Назарова, А. В. Методы и алгоритмы мультиагентного управления робототехнической системой / А. В. Назарова, Т. П. Ригова // Вестн. МГТУ им. Н. Э. Баумана. Сер. Приборостроение. – 2012. – С. 93–105.
2. Ростовцев, П. С. Обучение роботизированных систем с помощью нейронных сетей / П. С. Ростовцев, Д. Н. Васильев, М. И. Озерова // Россия молодая: передовые технологии в промышленности. – 2017. – № 2. – С. 123–125.
3. Neural Network-Based Learning from Demonstration of an Autonomous Ground Robot / Y. Fu, [et al.] // Machines. – 2019. – V. 7, № 2. – DOI: https://doi.org/10.3390/machines7020024.
4. Thanh, T. Deep Reinforcement Learning for Multiagent Systems: A Review of challenges, Solution and Applications / T. Thanh, N. Nguyen, S. // IEEE Transactions on Cybernetics. – Vol. 50, no. 9. – P. 3826–3839, 2020. – DOI: 10.1109/TCYB.2020.2977374.
5. Описание пакета ROS Toolbox {Электронный ресурс]. – Режим доступа: https.wwwmathworks.com/mathlab-central/filechange//66586-mobile-robotics-simulation-toolbox. – Дата доступа: 23.11.2020.

Поступила 20.09.2021

MASHINE REINFORCEMENT LEARNING FOR NAVIGATION OF MOBILE ROBOTS

A. SIDORENKO

New algorithm of machine learning for navigation of mobile robot navigation is introduced. It based on combination of Deep-Q-Learning and Double Q-Learning. Model is considered the movement of a mobile robot in some environment (environment is set Gazebo program package), known robot location and prevented obstackle collisions by navigation. Mobile Robotics Stimulation Toolbox and Gazebo visualization packages are used as Software. It is shown that the testing of new algorithm more than 10 times improved time characteristics in comparative with traditional algorithms of machine learning. The present algorithm may be to integrate in the apparatus means.

Keywords: *algorithm, robot, controlling, movement, machine learning.*