

УДК 004.896/621.865

DOI: 10.22213/2413-1172-2024-2-4-14

**Нейросетевой алгоритм обучения мобильного робота
в задаче следования за целью****И. С. Звонарев**, аспирант, ИжГТУ имени М. Т. Калашникова, Ижевск, Россия**Ю. Л. Караваев**, доктор технических наук, доцент, ИжГТУ имени М. Т. Калашникова, Ижевск, Россия

Статья посвящена исследованию, разработке и реализации искусственной нейронной сети с подкреплением для управления мобильным роботом с дифференциальным приводом для решения задачи следования за целью. Авторы исследования представляют подробное описание архитектуры нейронной сети, ее обучения с использованием алгоритма глубокого детерминированного градиента политики и интеграции с системой управления роботом. Выбрана распределенная структура нейронной сети для специализированной генерации управлений, в качестве которых авторами выбираются угловая и линейная скорость робота. Синтезирована система правил, учитывающая изменение расстояния между роботом и целевым объектом и отслеживание курсового угла в реальном времени. Рассмотрена математическая модель робота с дифференциальным приводом, на базе которой реализована программа имитационного моделирования для промежуточного обучения нейронной сети, в результате которой сформированы первоначальные весовые коэффициенты. С помощью данной программы можно обойтись меньшими энергетическими и временными затратами на начальном этапе исследования. Эксперименты по проверке эффективности разработанной нейронной сети проводились в среде симуляции Gazebo с использованием коммуникационного интерфейса ROS2. В статье описывается процесс интеграции нейронной сети с системой управления робота в среде симуляции, а также результаты испытаний и анализ полученных данных. Приведены результаты экспериментов для трех сценариев, в которых начальная позиция робота одинакова, остальные параметры генерируются по описанным авторами правилам. Подтверждена эффективность использования подобного решения для планирования траектории. Исследование вносит вклад в область развития автономных систем и демонстрирует потенциал применения искусственных нейронных сетей с подкреплением в робототехнике.

Ключевые слова: обучение мобильного робота, курсовой угол, система управления.**Введение**

В последнее время всё чаще системы управления роботизированных устройств, такие как колесные, шагающие, гусеничные и ползающие мобильные роботы, беспилотные воздушные суда самолетного, вертолетного и аэростатного типа, а также другие роботизированные системы оснащаются интеллектуальными составляющими. Эти составляющие строятся на базе искусственных нейронных сетей (ИНС) как с учителем, так и без него. Основной и наиболее интересной задачей интеллектуальных систем управления является обеспечение безаварийного движения робота в целевую точку пространства, или, другими словами, планирование пути.

Суть классической задачи планирования пути заключается в поиске оптимального маршру-

та от текущей до целевой точки при условии, что известны их координаты. Для ее решения может быть использовано множество разнообразных алгоритмов, в том числе метод искусственного потенциального поля [1], генетические алгоритмы [2], методы нечеткой логики [3] и методы обучения с подкреплением [4].

Однако при планировании пути в неизвестной динамической среде возникает проблема, заключающаяся в учете огромного количества факторов внешней среды, на основании которых система должна принимать решение о корректировке маршрута. Существует множество методов и алгоритмов планирования пути [5, 6]: метод глобального планировщика, который заключается в полном построении пути из начальной в целевую точку; локального планирования маршрута, который генерирует траекто-

рию на ближайший временной интервал (Δt); рефлекторный метод, основанный на наборе правил, определяющих действие по событию. Из приведенных выше методик наиболее подходящей для планирования пути в реальной динамической среде является метод локального планирования, так как глобальный планировщик будет вынужден перестраивать полный маршрут при изменении сцены, что является весьма ресурсозатраной операцией, а рефлекторный метод не эффективен в условиях динамической внешней среды.

Основной задачей системы управления является преследование динамической точки в пространстве и достижение ее окрестности. Данная задача схожа с классической задачей наведения ракеты на цель, которой посвящены работы А. Н. Толстикова [7], А. Ю. Сивова [8] и А. В. Богданова [9], откуда следует, что есть множество способов преследования объекта, например, методы погони, постоянного угла упреждения, параллельного сближения, базирующиеся на теории оптимального управления и основанные на дифференциальных играх преследования. Для осуществления однозначного преследования было решено использовать идею самого простого метода погони, заключающегося в поддержании нулевого значения курсового угла.

На основании статей [10–13], посвященных решению задач интеллектуальной навигации, было принято решение использовать в качестве основы для разработки самообучающейся системы управления глубокий детерминированный градиент политики DDPG (Deep Deterministic Policy Gradients), так как, несмотря на большую стабильность ИНС с учителем, выражаемую в принятии закономерных решений в типовых ситуациях [14–16], они проигрывают шаблонностью действий в условиях динамической среды так же, как и методы нечеткой логики. Что касается метода потенциальных полей и генетического алгоритма, то они не так успешно функционируют в неизвестных средах (без карты местности), как методы обучения с подкреплением [17–20]. Однако не каждая самообучающаяся нейронная сеть подходит для задач планирования пути в динамической среде. Наибольшее применение нашли глубокие нейронные сети с подкреплением [21], к которым относится DDPG; например, данный алгоритм был успешно применен для подобной задачи в работе [22].

Цель исследования – разработка интеллектуальной самообучающейся системы управления мобильным роботом с дифференциальным приводом.

Описание объекта управления и симуляции

Исследование проводится на базе ROS2 и среды виртуальной симуляции Gazebo. Объектом управления в виртуальной среде является мобильный робот с дифференциальным приводом (рис. 1). Данная модель оснащена камерой и лазерным дальномером, которые в дальнейшем могут использоваться для задач идентификации объектов окружающей среды и состояния мобильного робота.

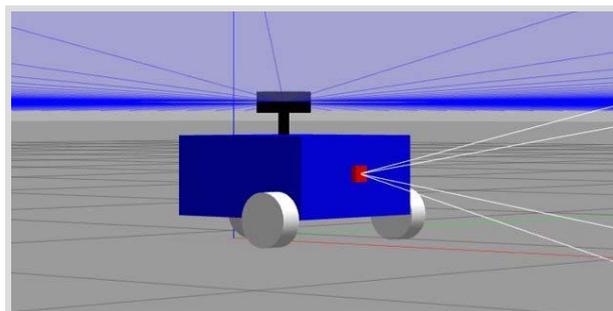


Рис. 1. Модель мобильного робота в среде Gazebo

Fig. 1. Model of a mobile robot in the Gazebo environment

Для реализации взаимосвязи ROS2 и Gazebo была создана программа, принцип работы которой основан на механизме взаимодействия API Gymnasium (в прошлом Gym) и марковском процессе принятия решения (MDP).

Целевой функцией ИНС является осуществление управления мобильным роботом при движении за объектом при его, в том числе, случайном движении.

Структура данных «вход-выход»

На вход ИНС передается вектор, содержащий 15 значений:

- 7 значений описывают положение робота в пространстве (x_r, y_r, z_r) , (Qx_r, Qy_r, Qz_r, Qw_r) – вектор поворота (кватернион);
- 6 значений содержат информацию о линейных и угловых скоростях $(v_x, v_y, v_z, w_x, w_y, w_z)$;
- 2 значения координат целевой точки (x_g, y_g) .

На выходе ИНС формирует 2 управления: скорость линейного перемещения и угловую скорость (v_s, w_s) .

Для увеличения скорости обучения и предварительной коррекции весовых коэффициентов перед симуляцией в ROS2 ИНС обучалась на базе упрощенной математической модели, после чего предобученная сеть тренировалась и тестировалась в ROS2.

Кинематическая схема мобильного робота, используемая в исследовании, представлена на рисунке 2.

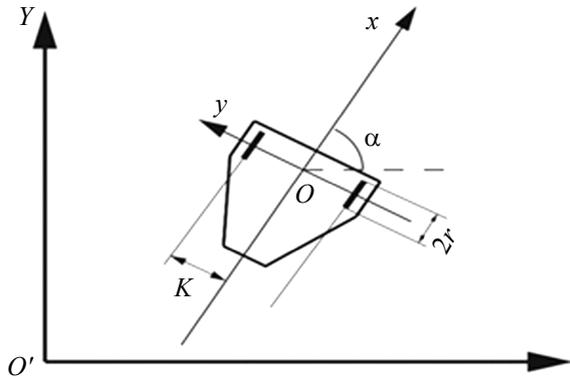


Рис. 2. Кинематическая схема мобильного робота

Fig. 2. Kinematic scheme of a mobile robot

Для вычисления управления используется система уравнений, применяемая на этапе предварительного обучения:

$$\begin{aligned} \frac{dx}{dt} &= V_1 \cos(\alpha); \\ \frac{dy}{dt} &= V_1 \sin(\alpha); \\ \frac{d\alpha}{dt} &= w, \end{aligned} \quad (1)$$

где $V_1 = \frac{r}{2}(w_r + w_l)$ – вектор линейной скорости робота в подвижной системе координат, направленной вдоль оси OX; w_l, w_r – угловая скорость левого и правого колес робота; α – угол поворота подвижной системы координат относительно неподвижной; w – угловая скорость робота; r – радиус колеса; x, y – координаты начала подвижной системы координат.

Базовый алгоритм DDPG

Базовый алгоритм DDPG (Deep Deterministic Policy Gradient – глубокий детерминированный градиент политики), представленный на рисунке 3, можно описать следующим образом [23]:

1. Инициализация θ_μ, θ_Q, B , а также целевые веса $\theta_\mu \rightarrow \theta'_\mu, \theta_Q \rightarrow \theta'_Q$.

2. Цикл для каждого эпизода:

- инициализировать s (вектор состояния);
- пока s не является терминальным состоянием:
 - вычислить новое действие с помощью сети актора и добавить шум ($\mu_\theta(s) + \text{noise} \rightarrow a$);

– выполнить действие a , а затем наблюдать за наградой r и следующим состоянием s' ;

– сохранить кортеж (s, a, r, s') в буфере воспроизведения B ;

– выбрать случайно мини-пакет из N кортежей (s_i, a_i, r_i, s'_i) из буфера B ;

– предсказать значение действия с использованием целевой сети и текущего вознаграждения $r_i + \gamma Q_\theta(s_i, \mu_\theta(s'_i)) \rightarrow \gamma_i$, где γ – коэффициент дисконтирования;

– обновить критика, минимизируя

$$\frac{1}{N} \sum_i (\gamma_i - Q_\theta(s_i, a_i))^2;$$

– обновить акторы, минимизируя

$$\frac{1}{N} \sum_i (\nabla_\theta Q_\mu(s_i, \mu_\theta(s_i)))^2;$$

– копировать целевые веса в локальные $\tau\theta_Q + (i - \tau)\theta'_Q \rightarrow \theta'_Q, \tau\theta_\mu + (i - \tau)\theta'_\mu \rightarrow \theta'_\mu$, где

τ – скорость обновления сети;

– перейти в следующее состояние $s'(s_{new}) \rightarrow s$.

Алгоритм, описанный выше, был взят за основу в текущем исследовании [24]. Для обучения были сформированы архитектуры для сетей актора и критика с двумя выходами в соответствии с задачей управления мобильным роботом с дифференциальным приводом.

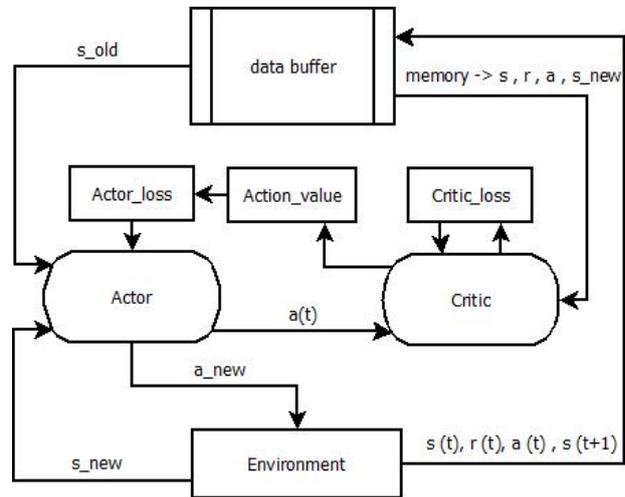


Рис. 3. Схема работы алгоритма DDPG

Fig. 3. DDPG algorithm work flow diagram

На рисунке 3 блоки Critic_loss и Actor_loss отвечают за вычисление ошибки; Action_value – за формирование ожидаемого (целевого) значения действия; databuffer – буфер хранения «исторических» данных; блоки Critic и Actor – это

непосредственно 2 составляющие сети DDPG; a_{new} – вектор действия, сформированный на выходе ИНС; $a(t)$ – действие сформированное сетью актора; Environment – среда симуляции, в которой робот совершает действия; s_{old} и s_{new} – соответственно, вектор состояния на предыдущем шаге и текущий.

Архитектуры сетей актора и критика

Архитектуры сетей актора и критика представлены на рисунках 4 и 5 соответственно. В качестве базовых слоев используются стандартные полносвязные слои, однако в качестве начального слоя было решено использовать

слой LSTM (слой с долгой краткосрочной памятью), а также применить Batch Normalization (нормализация выходных значений слоя). Кроме того, из-за необходимости генерации двух управляющих воздействий глубокая нейронная сеть была разделена на 2 ветви, отвечающие за разные управления.

На рисунках 4, 5 представлены структурные блоки ИНС. Разберем их значение на примере первого блока рисунка 5: 1 – номерная типовая операция или слой; 2 – тип операции или слоя; 3 – количество выходных значений; 4 – количество нейронов в слое.

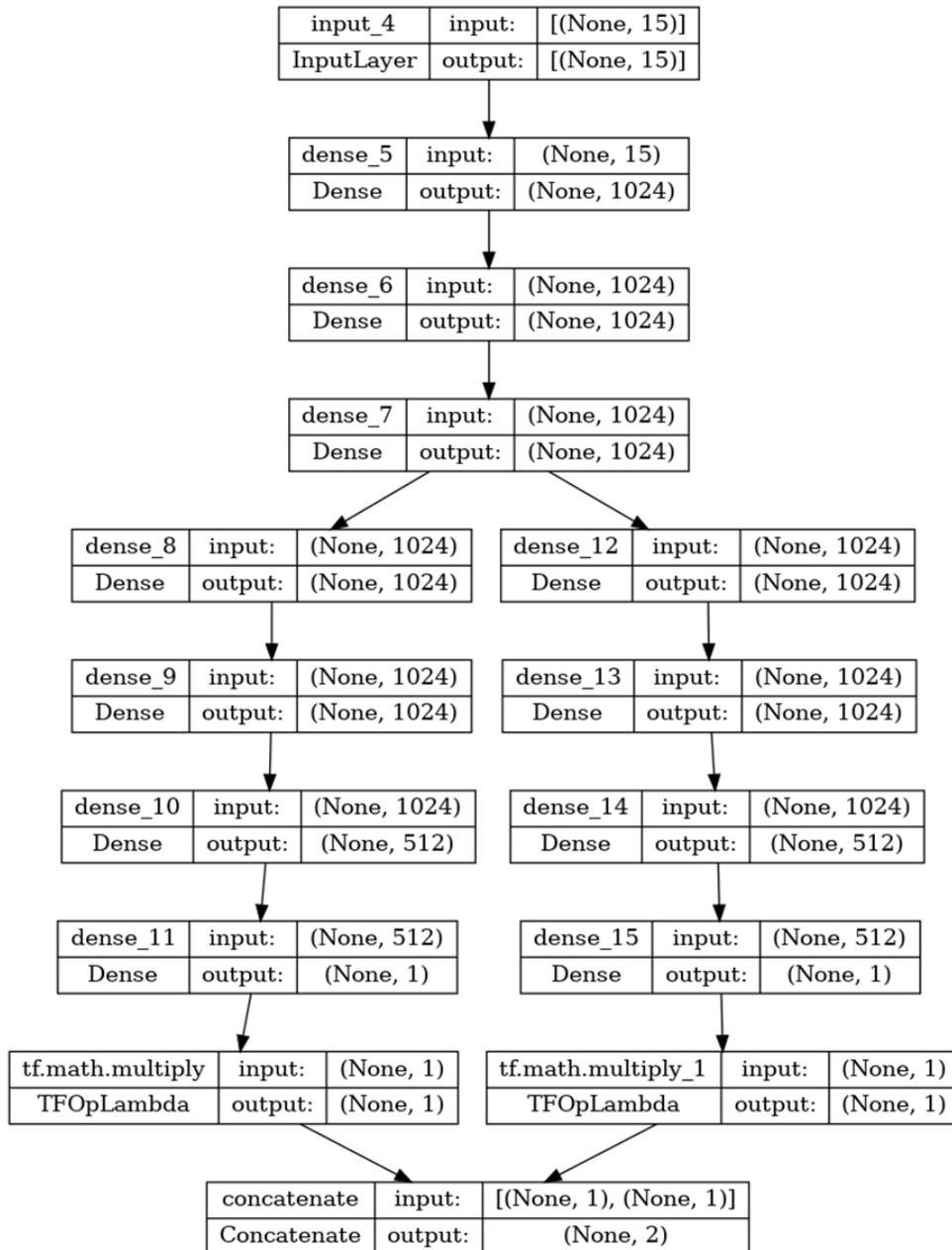


Рис. 4. Архитектура сети актора

Fig. 4. Actor network architecture

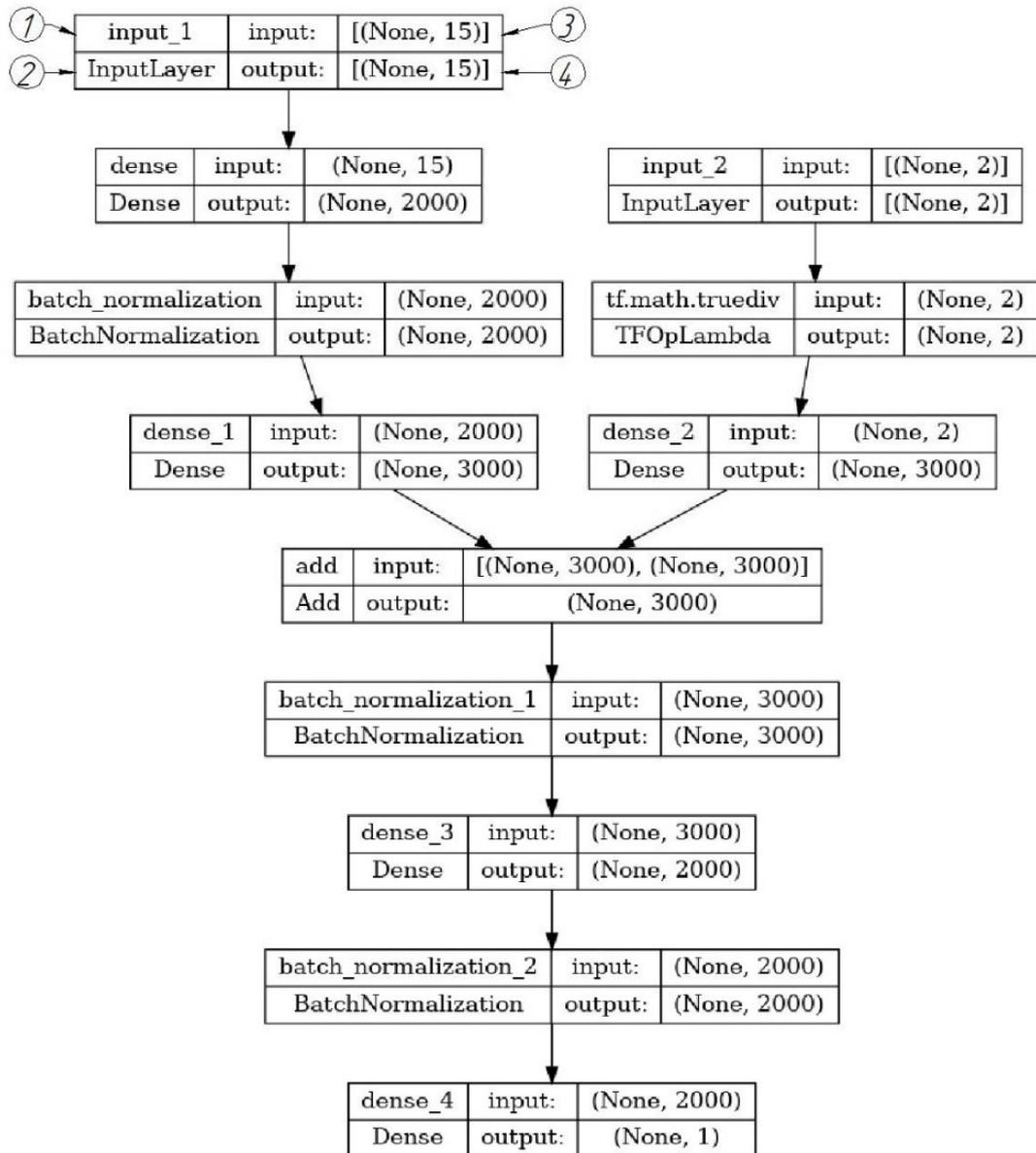


Рис. 5. Архитектура сети критика

Fig. 5. Critic network architecture

Политика начисления наград

Система расчета награды за эпизод основана на вычислении радиус-вектора от центра робота до координат целевой точки (табл. 1).

Таблица 1. Правила начисления наград

Table 1. Reward assignment rules

Условие	Награда, r
$dist_{new} < dist_{old}$	$r_{dist} = 17$
$ dist_{old} - dist_{new} \leq 1$	$r_{dist} = (-25) dist_{old} - dist_{new} + 15$
$ dist_{old} - dist_{new} \geq 1$	$r_{dist} = (-17)$

В таблице 1 $dist$ – расстояние от положения робота до координат целевой точки в текущий момент времени, диапазон значений $[-17...17]$ был подобран экспериментально и обеспечивает достаточную плавность изменения награды, а значения (-25) и 15 вычислены из уравнения $r_{dist} = x|dist_{old} - dist_{new}| + y$ при условии, что $r \in (-17; 17)$.

В системе правил (табл. 1) предусматривается постепенное изменение начисляемой награды, что помогает достичь более плавной траектории движения робота.

Кроме того, была введена дополнительная составляющая расчета награды за удержание курсового угла равным 0 , что позволило решить

проблему ошибочного начисления награды взаимного сближения (табл. 2).

Таблица 2. Правила начисления награды за удержание курса

Table 2. Reward assignment rules for maintaining course

Условие	Награда, r
$\gamma = 0$	$r_\gamma = 17$
$-\frac{\pi}{2} < \gamma < \frac{\pi}{2}$	$r_\gamma = -20,4\Delta\gamma + 16$
$-\frac{\pi}{2} > \gamma, \frac{\pi}{2} < \gamma$	$r_\gamma = (-17)$

В таблице 2 γ – курсовой угол, значения – 20,4 и 16 найдены из уравнения $r_\gamma = x\Delta\gamma + y$ при условии, что $r \in (-17;17)$; $\Delta\gamma$ – разница между текущим углом и углом на предыдущем шаге.

Таким образом, награда за эпизод вычисляется следующим образом:

$$r = r_\gamma + r_{dist}.$$

Если робот достигает целевой области, то итерация обучения заканчивается и начисляется награда $r = 400$.

Экспериментальные исследования

Начальные данные DDPG

Значения ключевых параметров ИНС: буфер воспроизведения $B = 1 \cdot 10^6$ (это значение обусловлено особенностями алгоритма обучения); актор и критик нейронной сети $\mu_\theta(s)$ с весами θ_μ были представлены выше в разделе «Архитектуры сетей актора и критика»; размер тренировочного мини-пакета $N = 200$ (выявлено в результате экспериментов); скорость обновления сети $\tau = 0,001$; коэффициент дисконтирования $\gamma = 0,99$ (по правилам может принимать значения от 0,95 до 0,99); функция шума для исключения переобучения $noise_i = noise_{i-1} + \theta(1 - noise_{i-1})\Delta t + std_{dev}\Delta trd_{norm}$, где $noise_i$ – значение шума на шаге i , Δt – временной шаг, std_{dev} – коэффициент отклонения 0,2, rd_{norm} – случайное значение нормального (гауссового) распределения; количество шагов за один эпизод 5000 (экспериментально было выявлено, что для обучения на ранних этапах необходимо до 5000 шагов); количество эпизодов 1000.

Описание условий эксперимента

В основе экспериментов лежит задача преследования подвижного объекта на плоскости.

Управляемый мобильный робот начинает движение из точки с координатами (0, 0), одновременно с ним начинает движение целевая точка, движущаяся по случайно выбранной системе правил.

1. Движение по окружности с центром в случайной точке, выбранной в диапазоне координат на плоскости [2...9], и случайными отклонениями по осям. Система уравнений движения выглядит следующим образом:

$$x = 3 \cos(\alpha) + \gamma_x \pm 0,1;$$

$$y = 3 \sin(\alpha) + \gamma_y \pm 0,1;$$

$$\alpha = \alpha + 1,$$

где α – угол в полярных координатах; x и y – соответствующие координаты точки; γ_x и γ_y – координаты центра окружности, выбираемые случайным образом.

На рисунке 6 представлен соответствующий эксперимент.

2. Движение по окружности с центром в случайной точке, выбранной в диапазоне координат на плоскости [2...9], основанное на системе уравнений

$$x = 3 \cos(\alpha) + \gamma_x;$$

$$y = 3 \sin(\alpha) + \gamma_y;$$

$$\alpha = \alpha + 1,$$

где α – угол в полярных координатах; x и y – соответствующие координаты точки; γ_x и γ_y – координаты центра окружности, выбираемые случайным образом.

На рисунке 7 представлен соответствующий эксперимент.

3. Циклическое движение по отрезку прямой, ограниченное диапазоном значений от 2 до 10 по одной из осей (начальная точка выбирается случайным образом).

Соответствующий эксперимент представлен на рисунке 8.

На рисунках 6–8 изменение угловых скоростей колес, рассчитанные на основании генерируемых алгоритмом значений линейных и угловых скоростей (угловые скорости в рад/с).

В результате проведенных экспериментов было выявлено, что система управления роботом на базе DDPG достигает целевой подвижной точки примерно в 76 % случаев; из 1000 эпизодов успешными были 758. Принятие решения в среднем занимает 0,04 с без использования графического процессора.

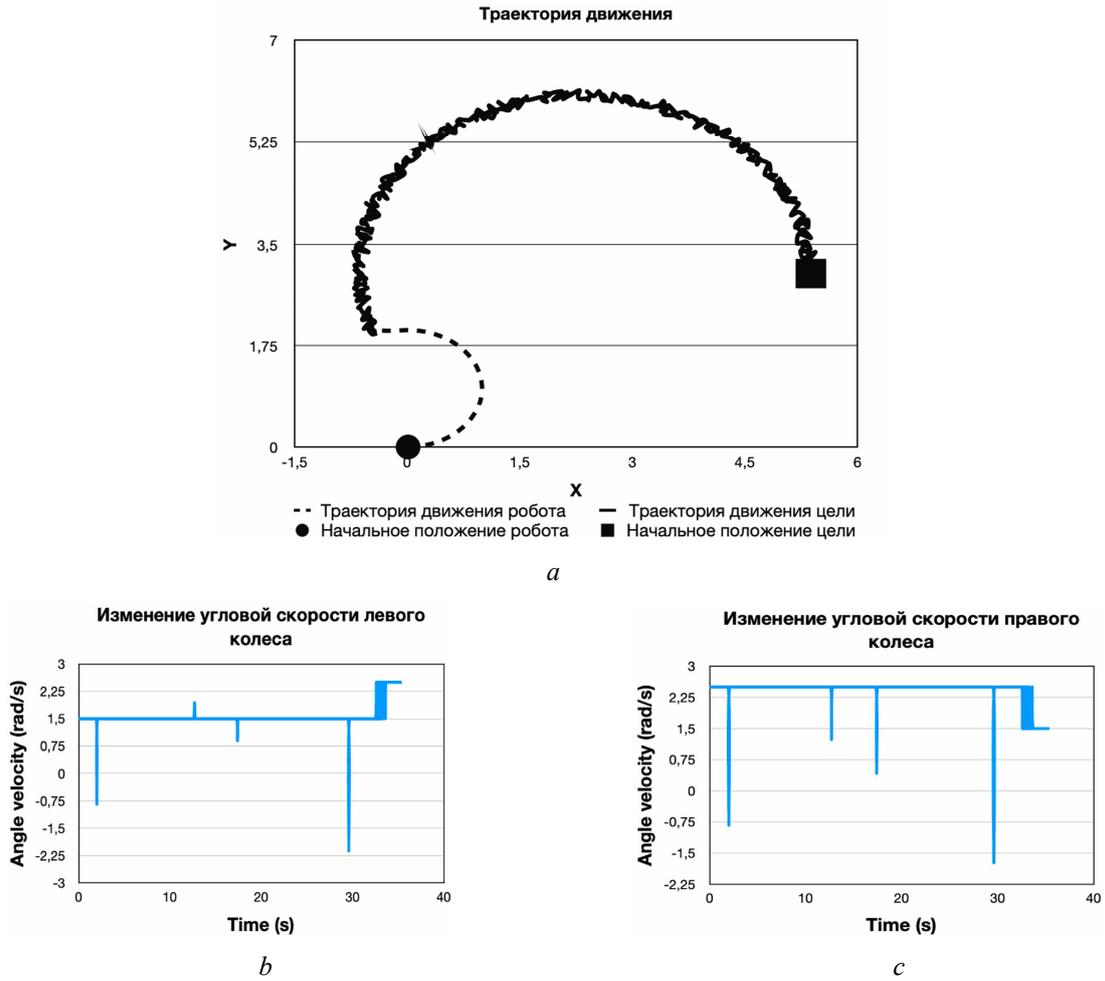


Рис. 6. Эксперимент № 1: *a* – траектории перемещения целевой точки и робота; *b* и *c* – изменение угловых скоростей
 Fig. 6. Experiment no. 1: *a* - trajectories of the target point and the robot; *b* and *c* - changes in angular velocities

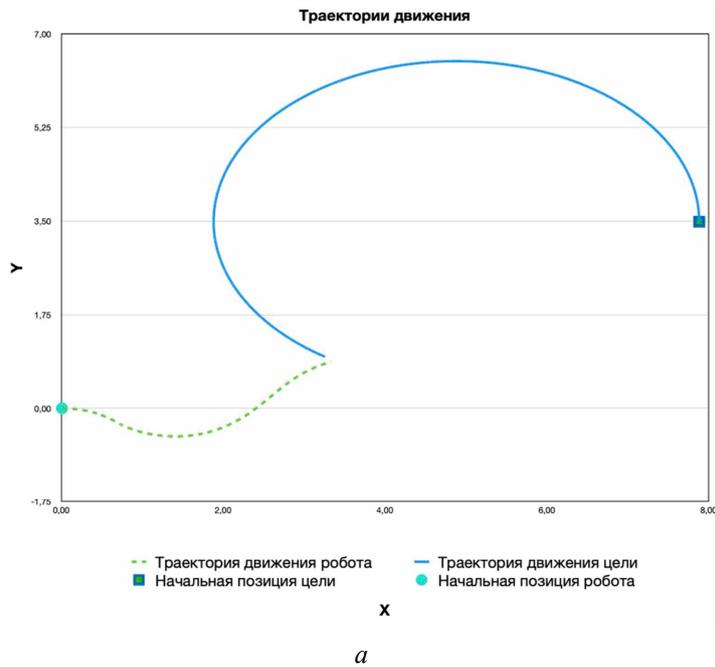


Рис. 7. Эксперимент № 2: *a* – траектории перемещения целевой точки и робота
 Fig. 7. Experiment no. 2: *a* - trajectories of the target point and the robots

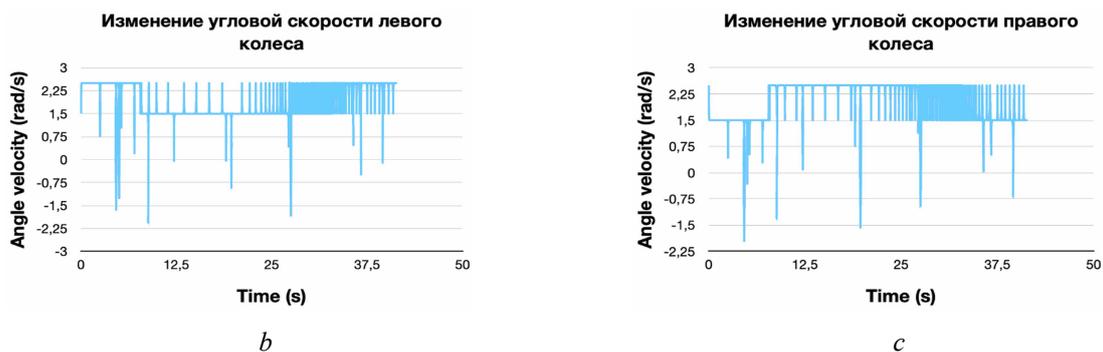


Рис. 7 (окончание, начало на с. 10): *b* и *c* – изменение угловых скоростей

Fig. 7 (continued from p. 10): *b* and *c* - changes in angular velocities

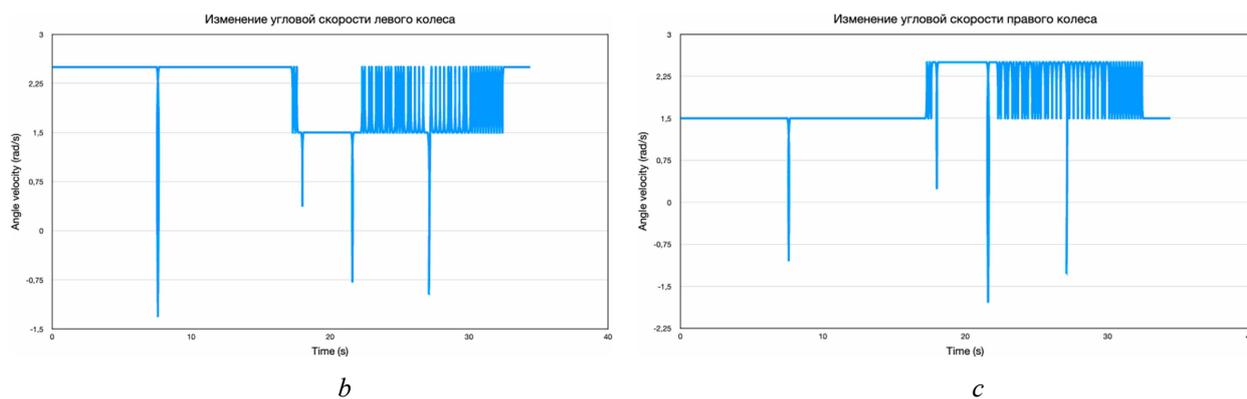
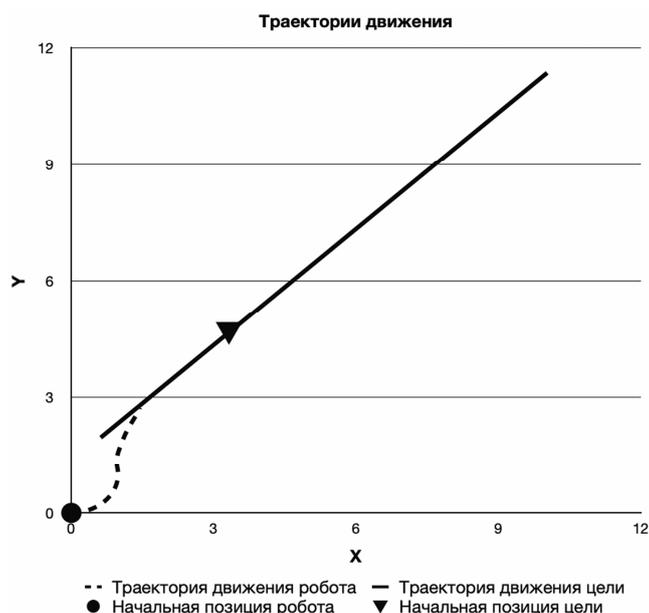


Рис. 8. Эксперимент № 3: *a* – траектории перемещения целевой точки и робота; *b* и *c* - изменение угловых скоростей

Fig. 8. Experiment no. 8: *a* - trajectories of the target point and the robot; *b* and *c* - changes in angular velocities

Выводы

Выявлено, что для задачи достижения подвижной целевой точки применимы нейронные сети без учителя DDPG, и, как следствие, модификация этого алгоритма также могут быть успешно применены для подобных задач.

Данные наработки будут использованы в дальнейшем изучении эффективности применения алгоритмов глубокого обучения с подкреплением для задач управления и планирования маршрута в динамических условиях для объектов с более сложной кинематикой, напри-

мер, многозвенных колесных мобильных роботов, сферороботов, омниколесных роботов.

Библиографические ссылки

1. Zhang C. (2018) Path planning for robot based on chaotic artificial potential field method. *Technology & Engineering*, 2018, vol. 317, no. 1, pp. 12-56.

2. De Santis R., Montanari R., Vignali G., Bottani E. (2018) An adapted ant colony optimization algorithm for the minimization of the travel distance of pickers in manual warehouses. *European Journal of Operational Research*, 2018, vol. 267, no. 1, pp. 120-137.

3. Sun P. and Yu Z. (2017) Tracking control for a cushion robot based on fuzzy path planning with safe angular velocity. *IEEE/CAA Journal of Automatica Sinica*, 2017, vol. 4, no. 4, pp. 610-619.

4. Xin J., Zhao H., Liu D., Li M. (2017) Application of deep reinforcement learning in mobile robot path planning: Proc. of the 2017 Chinese Automation Congress (CAC), pp. 7112-7116, Jinan, China, October 2017.

5. Lin Z. (2022) Path planning of mobile robot with PSO-based APF and fuzzy-based DWA subject to moving obstacles. *Transactions of the Institute of Measurement and Control*, 2022, vol. 44, no. 1, pp. 121-132.

6. Mishra D. K. (2022) Design of mobile robot navigation controller using neuro-fuzzy logic system. *Computers and Electrical Engineering*, 2022, vol. 101, p. 108044.

7. Толстиков А. Н., Толстиков Н. Г. Сравнение алгоритмов преследования объектов // Восточно-Европейский журнал передовых технологий. 2010. Т. 2, № 9 (44). С. 29–31.

8. Сивов А. Ю., Алешин М. Г. Алгоритм наведения луча фазированной антенной решетки на беспилотном летательном аппарате вертолетного типа // Журнал радиоэлектроники. 2020. № 4. С. 12. DOI: 10.30898/1684-1719.2020.4.5

9. Алгоритм наведения управляемой ракеты класса «воздух-воздух» с активной радиолокационной головкой самонаведения на вертолет при различном характере его полета / А. В. Богданов, С. А. Горбунов, А. А. Кучин, А. А. Хадур // Журнал Сибирского федерального университета. Серия: Техника и технологии. 2020. Т. 13, № 7. С. 829–842. DOI: 10.17516/1999-494X-0269

10. Li P. (2021) Research on dynamic path planning of mobile robot based on improved DDPG algorithm. *Mobile Information Systems*, 2021, vol. 2021, pp. 1-10.

11. Dong Y., Zou X. (2020) Mobile robot path planning based on improved DDPG reinforcement learning algorithm: IEEE 11th International Conference on software engineering and service science (ICSESS). IEEE, 2020, pp. 52-56.

12. Zhao Y. (2021) Path planning for mobile robots based on TPR-DDPG: International Joint Conference on Neural Networks (IJCNN). IEEE, 2021, pp. 1-8.

13. De Jesus J. C. (2021) Soft actor-critic for navigation of mobile robots. *Journal of Intelligent & Robotic Systems*, 2021, vol. 102, no. 2, p. 31.

14. Sun H. (2021) Motion planning for mobile robots - Focusing on deep reinforcement learning: A systematic review. *IEEE Access*, 2021, vol. 9, pp. 69061-69081.

15. Yu J., Su Y., Liao Y. (2020) The path planning of mobile robot by neural networks and hierarchical reinforcement learning. *Frontiers in Neurorobotics*, 2020, no. 14, p. 63.

16. Zhu K., Zhang T. (2021) Deep reinforcement learning based mobile robot navigation: A review. *Tsinghua Science and Technology*, 2021, vol. 26, no. 5, pp. 674-691.

17. Wen S. (2020) Path planning for active SLAM based on deep reinforcement learning under unknown environments. *Intelligent Service Robotics*, 2020, vol. 13, pp. 263-272.

18. Yan C., Xiang X., Wang C. (2020) Towards real-time path planning through deep reinforcement learning for a UAV in dynamic environments. *Journal of Intelligent & Robotic Systems*, 2020, vol. 98, pp. 297-309.

19. Wang D. (2021) Path planning of mobile robot in dynamic environment: fuzzy artificial potential field and extensible neural network. *Artificial Life and Robotics*, 2021, vol. 26, pp. 129-139.

20. Takahashi T. (2019) Learning heuristic functions for mobile robot path planning using deep neural networks: Proc. of the International Conference on Automated Planning and Scheduling, 2019, vol. 29, pp. 764-772.

21. Chen P. (2022) A deep reinforcement learning based method for real-time path planning and dynamic obstacle avoidance. *Neurocomputing*, 2022, vol. 497, pp. 64-75.

22. Yang B. (2021) Two-stage pursuit strategy for incomplete-information impulsive space pursuit-evasion mission using reinforcement learning. *Aerospace*, 2021, vol. 8, no. 10, p. 299.

23. Fu X. (2022) A UAV pursuit-evasion strategy based on DDPG and imitation learning. *International Journal of Aerospace Engineering*, 2022, vol. 2022, pp. 1-14.

24. Уиндер Ф. Обучение с подкреплением для реальных задач : пер. с англ. СПб. : БХВ-Петербург, 2023. 400 с.

References

1. Zhang C. (2018) Path planning for robot based on chaotic artificial potential field method. *Technology & Engineering*, 2018, vol. 317, no. 1, pp. 12-56.

2. De Santis R., Montanari R., Vignali G., Bottani E. (2018) An adapted ant colony optimization algorithm for the minimization of the travel distance of pickers in manual warehouses. *European Journal of Operational Research*, 2018, vol. 267, no. 1, pp. 120-137.

3. Sun P. and Yu Z. (2017) Tracking control for a cushion robot based on fuzzy path planning with safe angular velocity. *IEEE/CAA Journal of Automatica Sinica*, 2017, vol. 4, no. 4, pp. 610-619.

4. Xin J., Zhao H., Liu D., Li M. (2017) Application of deep reinforcement learning in mobile robot path plan-

ning: Proc. of the 2017 Chinese Automation Congress (CAC), pp. 7112-7116, Jinan, China, October 2017.

5. Lin Z. (2022) Path planning of mobile robot with PSO-based APF and fuzzy-based DWA subject to moving obstacles. Transactions of the Institute of Measurement and Control, 2022, vol. 44, no. 1, pp. 121-132.

6. Mishra D.K. (2022) Design of mobile robot navigation controller using neuro-fuzzy logic system. Computers and Electrical Engineering, 2022, vol. 101, p. 108044.

7. Tolstikov A.N., Tolstikov N.G. (2010) [Comparison of pursuit algorithms]. *Eastern-European Journal of Advanced Technologies*, 2010, no. 2(9), pp. 29-31 (in Russ.).

8. Sivov A.Yu., Aleshin M.G. (2020) [Beam steering algorithm for a phased antenna array on a rotary-wing unmanned aerial vehicle]. *Radioelectronics*, 2020, vol. 4, p. 12 (in Russ.). DOI: 10.30898/1684-1719.2020.4.5

9. Bogdanov A.V., Gorbunov S.A., Kuchin A.A., Khadur A.A. (2020). [Algorithm for guiding an air-to-air guided missile with an active radar homing head on a helicopter under different flight conditions]. *Journal of the Siberian Federal University. Engineering & Technologies*, 2020, vol. 13, no. 7, pp. 829-842 (in Russ.). DOI: 10.17516/1999-494X-0269

10. Li P. (2021) Research on dynamic path planning of mobile robot based on improved DDPG algorithm. Mobile Information Systems, 2021, vol. 2021, pp. 1-10.

11. Dong Y., Zou X. (2020) Mobile robot path planning based on improved DDPG reinforcement learning algorithm: IEEE 11th International Conference on software engineering and service science (ICSESS). IEEE, 2020, pp. 52-56.

12. Zhao Y. (2021) Path planning for mobile robots based on TPR-DDPG: International Joint Conference on Neural Networks (IJCNN). IEEE, 2021, pp. 1-8.

13. De Jesus J. C. (2021) Soft actor-critic for navigation of mobile robots. Journal of Intelligent & Robotic Systems, 2021, vol. 102, no. 2, p. 31.

14. Sun H. (2021) Motion planning for mobile robots - Focusing on deep reinforcement learning: A systematic review. IEEE Access, 2021, vol. 9, pp. 69061-69081.

15. Yu J., Su Y., Liao Y. (2020) The path planning of mobile robot by neural networks and hierarchical reinforcement learning. Frontiers in Neurorobotics, 2020, no. 14, p. 63.

16. Zhu K., Zhang T. (2021) Deep reinforcement learning based mobile robot navigation: A review. Tsinghua Science and Technology, 2021, vol. 26, no. 5, pp. 674-691.

17. Wen S. (2020) Path planning for active SLAM based on deep reinforcement learning under unknown environments. Intelligent Service Robotics, 2020, vol. 13, pp. 263-272.

18. Yan C., Xiang X., Wang C. (2020) Towards real-time path planning through deep reinforcement learning for a UAV in dynamic environments. Journal of Intelligent & Robotic Systems, 2020, vol. 98, pp. 297-309.

19. Wang D. (2021) Path planning of mobile robot in dynamic environment: fuzzy artificial potential field and extensible neural network. Artificial Life and Robotics, 2021, vol. 26, pp. 129-139.

20. Takahashi T. (2019) Learning heuristic functions for mobile robot path planning using deep neural networks: Proc. of the International Conference on Automated Planning and Scheduling, 2019, vol. 29, pp. 764-772.

21. Chen P. (2022) A deep reinforcement learning based method for real-time path planning and dynamic obstacle avoidance. Neurocomputing, 2022, vol. 497, pp. 64-75.

22. Yang B. (2021) Two-stage pursuit strategy for incomplete-information impulsive space pursuit-evasion mission using reinforcement learning. Aerospace, 2021, vol. 8, no. 10, p. 299.

23. Fu X. (2022) A UAV pursuit-evasion strategy based on DDPG and imitation learning. International Journal of Aerospace Engineering, 2022, vol. 2022, pp. 1-14.

24. Winder M. (2023) *Obuchenie s podkrepleniem dlja real'nyh zadach* [Reinforcement Learning for Real-World Problems]. St. Petersburg: BHV-Petersburg Publ., 2023, 400 p. (in Russ.).

Neural Network Algorithm for Training a Mobile Robot in the Task of Following a Target

I.S. Zvonarev, Post-graduate, Kalashnikov ISTU, Izhevsk, Russia

Yu.L. Karavaev, DSc in Engineering, Associate Professor, Kalashnikov ISTU, Izhevsk, Russia

The article is devoted to the research, development and implementation of an artificial neural network with reinforcement for controlling a mobile robot with a differential drive to solve the task of following a target. The authors of the study present a detailed description of the architecture of the neural network, its training using a deep deterministic gradient policy algorithm, and integration with the robot control system. A distributed neural network structure was selected for specialized generation of controls, for which the authors chose the angular and linear speed of the robot. A system of rules has been synthesized that takes into account changes in the distance between the robot and the target object and tracking of the heading angle in real time. A mathematical model of a robot with a differential drive is considered, on the basis of which a simulation program is implemented for intermediate training of a neural network, as a result of which the initial weighting coefficients are formed. Using this program, you can get by with less energy and time costs at the initial stage of the study. Experiments to test the effectiveness of the developed neural network were carried out in the Gazebo simulation environment using the ROS2 communication interface. The article describes the process of integrating a neural network with a robot control system in a simulation environment, as well as test results and analysis of the obtained data. The results of experiments are presented for three scenarios in which the initial position of the robot is the same, the remaining parameters are generated

according to the rules described by the authors. The effectiveness of using such a solution for trajectory planning has been confirmed. The research contributes to the field of autonomous systems and demonstrates the potential of artificial reinforcement neural networks in robotics.

Keywords: mobile robot learning, heading angle, control system.

Получено 09.04.2024

Образец цитирования

Звонарев И. С., Караваев Ю. Л. Нейросетевой алгоритм обучения мобильного робота в задаче следования за целью // Вестник ИжГТУ имени М. Т. Калашникова. 2024. Т. 27, № 2. С. 4–14. DOI: 10.22213/2413-1172-2024-2-4-14

For Citation

Zvonarev I.S., Karavaev Yu.L. (2024) [Neural Network Algorithm for Training a Mobile Robot in the Task of Following a Target]. *Vestnik IzhGTU imeni M.T. Kalashnikova*, 2024, vol. 27, no. 2, pp. 4-14 (in Russ.). DOI: 10.22213/2413-1172-2024-2-4-14