
МОДЕЛИРОВАНИЕ АЛГОРИТМА АДАПТАЦИИ СКОРОСТИ ПЕРЕДАЧИ ДАННЫХ НА ОСНОВЕ ГЛУБОКОГО Q-ОБУЧЕНИЯ В БЕСПРОВОДНЫХ СЕТЯХ IEEE 802.11

Малышев Владислав Владимирович,

магистрант

Воронежский государственный университет, кафедра информационных систем

E-mail: valdesius@yandex.ru

Аннотация

В работе рассматривается задача адаптации скорости передачи данных в беспроводных сетях стандарта IEEE 802.11. Эффективный выбор скорости передачи оказывает непосредственное влияние на пропускную способность сети, задержку передачи и вероятность потери пакетов в условиях изменяющегося состояния радиоканала. Для решения данной задачи предложен алгоритм, основанный на методе глубокого обучения с подкреплением Deep Q-Network. Реализация выполнена в среде моделирования сетей ns-3 с использованием библиотеки ns3-ai, обеспечивающей взаимодействие сетевого симулятора с моделью машинного обучения. Агент принимает решение о выборе скорости передачи на основании текущего состояния сети и накопленного опыта, формируя стратегию, направленную на максимизацию эффективности передачи данных. Проведено моделирование разработанного алгоритма и выполнена оценка его поведения по основным характеристикам беспроводной сети.

Ключевые слова: адаптация скорости передачи данных; IEEE 802.11; глубокое обучение с подкреплением; Deep Q-Network; DQN; ns-3; ns3-ai; беспроводные сети.

SIMULATION OF A DATA RATE ADAPTATION ALGORITHM BASED ON DEEP Q-LEARNING FOR IEEE 802.11 WIRELESS NETWORKS

Malyshev Vladislav Vladimirovich,

Master's Student

Voronezh State University, Department of Information Systems

E-mail: valdesius@yandex.ru

ABSTRACT

This paper addresses the problem of data rate adaptation in IEEE 802.11 wireless networks. The effective selection of the transmission rate has a direct impact on network throughput, transmission delay, and packet loss probability under dynamically changing wireless channel conditions. To solve this problem, a data rate adaptation algorithm based on the Deep Q-Network (DQN) reinforcement learning method is proposed. The algorithm is implemented in the ns-3 network simulator using the ns3-ai library, which enables interaction between the network simulation environment and the machine learning model. The agent selects the transmission rate

based on the current network state and previously acquired experience, gradually learning a policy aimed at maximizing data transmission efficiency. Simulation experiments were conducted to evaluate the performance of the proposed algorithm with respect to the main characteristics of a wireless network.

Keywords: data rate adaptation; IEEE 802.11; deep reinforcement learning; Deep Q-Network; DQN; ns-3; ns3-ai; wireless networks.

Введение

Беспроводные сети стандарта IEEE 802.11 широко применяются для организации локальных сетей благодаря высокой скорости передачи данных и простоте развертывания. Одной из важных задач при их функционировании является выбор скорости передачи данных, который оказывает непосредственное влияние на пропускную способность сети, задержку передачи и вероятность потери пакетов [1, с. 1139].

Для решения данной задачи используются алгоритмы адаптации скорости передачи данных (Rate Adaptation Algorithm, RAA), обеспечивающие выбор схемы модуляции и кодирования (Modulation and Coding Scheme, MCS) в зависимости от текущего состояния радиоканала. Однако большинство существующих алгоритмов основано на заранее определенных правилах и не всегда обеспечивает эффективную работу в условиях изменяющегося качества беспроводного канала.

Одним из перспективных подходов к решению данной задачи является применение методов обучения с подкреплением, позволяющих агенту самостоятельно формировать стратегию выбора скорости передачи на основе накопленного опыта взаимодействия со средой [2, с. 1].

Предлагаемый метод

В работе предлагается алгоритм адаптации скорости передачи данных на основе метода глубокого обучения с подкреплением Deep Q-Network (DQN), предполагающего аппроксимацию функции ценности действий с помощью глубокой нейронной сети [3, с. 529]. Алгоритм реализован в сетевом симуляторе ns-3 [4, с. 15] с использованием библиотеки ns3-ai, обеспечивающей обмен данными между сетевой моделью и алгоритмом машинного обучения [5, с. 57]. В качестве программной основы использован пример rate-control/thompson-sampling, в котором базовый алгоритм Thompson Sampling заменён на модель DQN.

За основу реализации был принят алгоритм адаптации скорости передачи данных на основе обучения с подкреплением, предложенный в работе [1, с. 1139–1141]. В отличие от исходной реализации, использующей алгоритм SARSA, в настоящей работе применяется алгоритм Deep Q-Network (DQN).

Общая архитектура взаимодействия сетевого симулятора, библиотеки ns3-ai и агента DQN представлена на рис. 1.

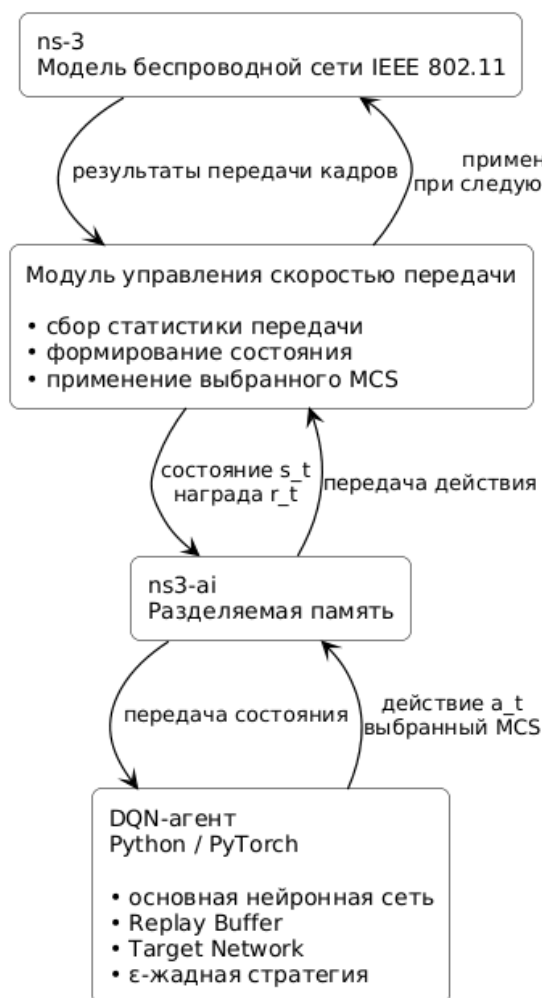


Рис. 1. Архитектура взаимодействия сетевого симулятора ns-3 и DQN-агента

В процессе моделирования агент взаимодействует со средой, получая информацию о текущем состоянии сети и выбирая скорость передачи данных. После выполнения передачи вычисляется награда, отражающая эффективность принятого решения, после чего параметры нейронной сети обновляются с использованием накопленного опыта.

Состояние среды представляется вектором:

$$s_t = [r_t, p_t, n_t^{succ}, n_t^{fail}]$$

где r_t – нормализованная скорость передачи, p_t – доля потерь пакетов, n_t^{succ} и n_t^{fail} – числа успешных и неуспешных передач соответственно.

В качестве действий агент выбирает одну из доступных схем модуляции и кодирования стандарта IEEE 802.11. Выбор действия осуществляется по ϵ -жадной стратегии, сочетающей исследование новых состояний и использование накопленного опыта.

Функция награды определяется выражением:

$$R = \frac{(Rate * (1 - PacketLoss))}{Rate_{max}},$$

что позволяет одновременно учитывать скорость передачи данных и надежность канала. Для обучения используется алгоритм DQN с буфером воспроизведения (Replay Buffer) и целевой нейронной сетью (Target Network), обеспечивающими устойчивость процесса обучения.

Экспериментальное исследование

Для оценки работоспособности разработанного алгоритма выполнено моделирование в сетевом симуляторе ns-3. В качестве среды использовалась беспроводная

сеть стандарта IEEE 802.11, в которой алгоритм DQN осуществлял выбор схемы модуляции и кодирования в зависимости от текущего состояния канала.

Параметры эксперимента представлены в табл. 1.

Таблица 1

Параметры эксперимента и обучения

Параметр	Значение
Стандарт беспроводной сети	IEEE 802.11a/g
Фреймворк	PyTorch

Продолжение таблицы 1

Размер вектора состояния	4
Размер Replay Buffer	10000
Batch Size	32
Коэффициент дисконтирования γ	0,95
Начальное значение ϵ	1,0
Минимальное значение ϵ	0,05
Скорость обучения	0,001
Период обновления целевой сети	1000 шагов симуляции

В ходе моделирования анализировались основные показатели качества передачи данных: пропускная способность сети, средняя задержка передачи пакетов и успешность передачи кадров. Обучение агента выполнялось непосредственно в процессе моделирования, что позволило постепенно формировать стратегию выбора скорости передачи данных на основе накопленного опыта взаимодействия со средой.

В процессе экспериментального моделирования удалось достичь значимых показателей работы системы. Средняя пропускная способность сети составила 42,3195 Мбит/с, при этом фиксировались пиковые значения до 50,7114 Мбит/с. За время наблюдения был передан общий объём данных, равный 55,469 мегабайт. Время задержки передачи данных в среднем составило 21,0212 миллисекунд. Система продемонстрировала стабильную работу в различных условиях функционирования, эффективно используя доступные ресурсы канала связи.

В ходе исследования подтвердилось, что система способна обеспечивать надёжную передачу информации с минимальными потерями. Наблюдается рациональное использование пропускной способности канала при различных нагрузках. Полученные результаты свидетельствуют о работоспособности разработанного алгоритма и подтверждают перспективность применения методов глубокого обучения с подкреплением для решения задачи адаптации скорости передачи данных.

Заключение

Разработан алгоритм адаптации скорости передачи данных на основе Deep Q-Network и реализован в среде моделирования ns-3 с использованием библиотеки ns3-ai. Предложенный подход обеспечивает принятие решений на основе текущего состояния сети и накопленного опыта взаимодействия со средой и может служить основой для дальнейших исследований методов глубокого обучения с подкреплением.

Список литературы:

1. Peserico G., Fedullo T., Morato A., Vitturi S., Tramarin F. Rate Adaptation by Reinforcement Learning for Wi-Fi Industrial Networks // 2020 IEEE 25th International Conference on

Emerging Technologies and Factory Automation (ETFA). Vienna, Austria, 2020. P. 1139–1142.

2. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. 2nd ed. Cambridge, MA: MIT Press, 2018. 552 p.
3. Mnih V., Kavukcuoglu K., Silver D. et al. Human-level control through deep reinforcement learning // Nature. 2015. Vol. 518, No. 7540. P. 529–533.
4. Riley G. F., Henderson T. R. The ns-3 Network Simulator // Modeling and Tools for Network Simulation. Berlin: Springer, 2010. P. 15–34.
5. Yin H., Liu P., Liu K. et al. ns3-ai: Fostering Artificial Intelligence Algorithms for Networking Research // Proceedings of the 2020 Workshop on ns-3 (WNS3 '20). New York: ACM, 2020. P. 57–64.

References:

1. Peserico G., Fedullo T., Morato A., Vitturi S., Tramarin F. Rate Adaptation by Reinforcement Learning for Wi-Fi Industrial Networks // 2020 IEEE 25th International Conference on Emerging Technologies and Factory Automation (ETFA). Vienna, Austria, 2020, pp. 1139–1142.
2. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. 2nd ed. Cambridge, MA: MIT Press, 2018. 552 p.
3. Mnih V., Kavukcuoglu K., Silver D. et al. Human-level control through deep reinforcement learning // Nature. 2015. Vol. 518, No. 7540. P. 529–533.
4. Riley G. F., Henderson T. R. The ns-3 Network Simulator // Modeling and Tools for Network Simulation. Berlin: Springer, 2010. pp. 15–34.
5. Yin H., Liu P., Liu K. et al. ns3-ai: Fostering Artificial Intelligence Algorithms for Networking Research // Proceedings of the 2020 Workshop on ns-3 (WNS3 '20). New York: ACM, 2020. pp. 57–64.