UDC 004.8

A REVIEW OF REINFORCEMENT LEARNING ALGORITHMS FOR REAL-TIME PROBLEM SOLVING

Ismailov P.
master's degree, Al-Farabi Kazakh National University (Almaty, Kazakhstan)

ОБЗОР АЛГОРИТМОВ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ ДЛЯ РЕШЕНИЯ ПРОБЛЕМ В РЕАЛЬНОМ ВРЕМЕНИ

Исмаилов П.Н.
магистр, Казахский национальный университет имени аль-Фараби (Алматы, Казахстан)

Abstract

This article provides an overview of modern reinforcement learning (RL) algorithms used for solving real-time tasks. Various methods, including Q-learning, gradient algorithms, recurrent neural networks, and distributed learning, are analyzed, highlighting their capacity to adapt to changing environments and make effective decisions. Special attention is paid to computational resource optimization and model stability, which are critical for tasks requiring rapid response. Knowledge adaptation and transfer methods, such as multi-task learning, are also discussed as approaches that accelerate model training in data-scarce conditions. The application of these methods makes RL an effective tool for dynamic, high-tech fields, including autonomous control and robotics. The findings demonstrate the potential of RL in real-time conditions but also underline the need for further research to enhance algorithm resilience and reduce computational costs.

Keywords: reinforcement learning, adaptation, real-time algorithms, autonomous control, stability.

Аннотация

В статье представлен обзор современных алгоритмов обучения с подкреплением (ОП), применяемых для решения задач в режиме реального времени. Рассмотрены различные методы, включая Q-обучение, градиентные алгоритмы, рекуррентные нейронные сети и распределенное обучение, которые позволяют моделям адаптироваться к изменениям в окружающей среде и принимать эффективные решения. Особое внимание уделено проблемам оптимизации вычислительных ресурсов и обеспечению устойчивости моделей, что критично для задач с быстрым откликом. Также обсуждаются методы адаптации и переноса знаний, такие как мультизадачное обучение, которые позволяют ускорить обучение моделей в условиях недостатка данных. Результаты исследования демонстрируют потенциал использования ОП в условиях реального времени, однако также выявляют необходимость дальнейших исследований для решения задач, связанных с устойчивостью алгоритмов и снижением вычислительных затрат.

Ключевые слова: обучение с подкреплением, адаптация, алгоритмы реального времени, автономное управление, устойчивость.

Introduction

The modern development of artificial intelligence (AI) technologies has led to the need for adaptive algorithms capable of making real-time decisions. Reinforcement learning (RL), as one of the key AI directions, enables the creation of models that learn to interact with dynamic environments and optimize actions based on accumulated experience. Unlike traditional learning methods, RL relies on reward and punishment principles, allowing models to independently develop strategies to achieve specified goals. This article aims to systematize RL methods for solving real-time problems and evaluate their effectiveness.

The main tasks that RL addresses include building models capable of considering environmental uncertainties and changing conditions, which is especially relevant for areas such as unmanned vehicle control, robotics, and automated decision-making systems. RL requires handling large volumes of data and adapting models to real-time environmental changes. This article explores algorithms focused on high performance and minimal response time, allowing them to be successfully applied in various practical fields.

Introducing RL methods in real-time tasks requires hybrid approaches combining classical learning algorithms and the latest deep learning technologies. This article examines various approaches to the development and application of RL in real-time, analyzing their strengths and limitations. The study aims to provide an overview of modern RL algorithms and identify the most effective solutions for applications that require rapid system response.

Main part

RL algorithms are divided into several categories, with the most interest in Q-learning and gradient-based methods. Q-learning, one of the basic RL algorithms, enables efficient agent training using state tables; however, as environment complexity grows, this method encounters the "curse of dimensionality" problem [1]. For real-time tasks, algorithms that can quickly adapt to environmental changes are crucial. An example is gradient-based methods, which optimize agent behavior by minimizing loss functions. Algorithms like Policy Gradient and Actor-Critic focus on continuous strategy updates, making them suitable for high-dynamic tasks. Figure 1 shows the architecture of the Actor-Critic model, widely used in real-time tasks.

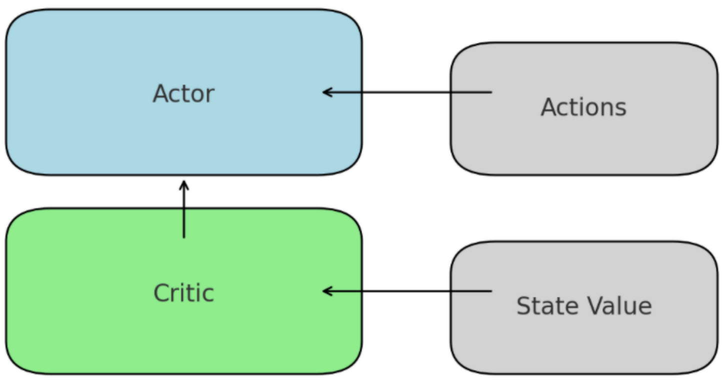

Figure 1. Actor-Critic model architecture

This figure shows the interaction between the critic, evaluating the current state, and the actor, adjusting the agent's actions.

Another promising area involves algorithms that use recurrent neural networks (RNNs) capable of capturing temporal dependencies in data [2]. Using RNNs in RL allows models to remember previous states, which is particularly useful for sequence-based tasks. An example implementation is the Recurrent Q-Learning algorithm, which effectively trains agents in conditions where state information changes over time. A critical aspect of applying RL to real-time tasks is optimizing computational resources. Since tasks requiring rapid response have high performance requirements, methods that implement parallel computing, such as Distributed Reinforcement Learning, are popular. In such systems, computations are distributed across multiple processors or machines, significantly increasing training speed [3].

An example of real-time RL usage is in unmanned vehicle control, where the algorithm must quickly adapt to road changes and make decisions in unpredictable environments. Algorithms combining RL and deep learning, such as Deep Deterministic Policy Gradient (DDPG), are used for these tasks. DDPG optimizes agent behavior for continuous action tasks, making it effective in dynamic environments. To increase stability and improve training outcomes, algorithms with prioritization mechanisms, such as Prioritized Experience Replay, are applied. In this method, significant events carry more weight in the learning process, enabling the algorithm to adapt more quickly to environmental changes. This approach is used in systems with high stability and reliability requirements [4, 5].

Adaptation and knowledge transfer in real-time tasks

In real-time RL applications, a critical task remains adapting algorithms to new conditions and their ability to transfer knowledge between different tasks. During model development and training, the need often arises to accelerate the learning process by using knowledge already obtained in similar conditions [6-8]. This approach significantly reduces time and computational costs, allowing the model to quickly reach the required level of adaptability.

Knowledge transfer is implemented through methods that include retraining on new data using previous models or combining experience from multiple sources. A solution to this problem is multitask learning, where the model simultaneously trains on several tasks, with the ability to identify and use common patterns. This approach is especially useful when tasks have similar elements or goals, allowing the model to "switch" between tasks while retaining the achieved level of competence [9].

The implementation of knowledge transfer methods requires not only structural adaptation of the model but also controlling the possibility of excessive accumulation of "old" knowledge, which may lose relevance. A crucial aspect is a mechanism for tracking the significance of previously obtained data and updating it according to new task requirements. This dynamic data control system enables models to handle changing conditions effectively and avoid situations where accumulated knowledge begins to reduce performance in new tasks [10].

In real-time tasks, knowledge transfer plays an important role in cases of data scarcity and situations where training "from scratch" is difficult due to time or resource constraints. Knowledge transfer enables models to adapt more quickly and function effectively in conditions close to real, making it an integral part of RL applications in dynamic environment tasks.

Conclusion

Reinforcement learning algorithms offer extensive possibilities for creating adaptive models capable of solving tasks in real-time. The methods discussed, including Q-learning, gradient approaches, the use of recurrent neural networks, and distributed learning, demonstrate high efficiency when interacting with dynamic environments. These algorithms can quickly adapt to environmental changes, making them especially useful for areas like autonomous control and robotics.

When using RL methods, it is essential to consider complexities related to optimizing computational resources and ensuring model stability. Adapting algorithms to real-world conditions requires prioritization methods and hybrid approaches involving deep learning. Knowledge transfer also promotes efficient algorithm implementation in situations where comprehensive training "from scratch" is impossible. Transfer mechanisms and multitask learning significantly reduce training costs and accelerate model adaptation.

In general, the application of RL methods in real-time tasks opens prospects for their use in high-tech and dynamic fields. However, further research tasks remain, such as developing more robust algorithms, reducing computational costs, and improving quality control methods. Continuing progress in this field will enable integrating RL into more complex and variable conditions, expanding the practical applications of these methods.

References

1. Kulida E.L., Lebedev V.G. Methods for solving planning and air traffic flow control tasks. Part 2. Application of deep reinforcement learning methods // Control Problems. 2023. Vol. 2. P. 3-18.

2. Orlova E.V. Reinforcement learning as an artificial intelligence technology for solving socio-economic problems: evaluation of algorithm performance // π-Economy. 2023. Vol. 16. No.5. P. 38-50.
3. Eremeev A.P., Kozhukhov A.A. Implementation of reinforcement learning methods based on temporal differences and a multi-agent approach for real-time intelligent systems // Software Products and Systems. 2017. Vol. 30. No.1. P. 28-33.
4. Andronov S.A., Prokofeva M.S. Comparative evaluation of the effectiveness of reinforcement learning and optimization algorithms in transport process management in the Anylogic environment // Systems Analysis and Logistics: Journal. 2022. No.2. P. 32.
5. Agarkov Y.Yu. Machine learning methods for optimizing the design of neuromorphic systems // Innovations and Investments. 2023. No.6. P. 313-319.
6. Zaitseva Yu.S. Artificial intelligence methods for solving control tasks in robotic and mechatronic systems: a review // Proceedings of Higher Educational Institutions. Mechanical Engineering. 2024. No.1(766). P. 41-56.
7. Mohammed H.A.R., Zargaryan E.V., Zargaryan Yu.A. Machine learning and deep learning models for electronic information security in mobile networks // Proceedings of the Southern Federal University. Technical Sciences. 2022. No.3(227). P. 211-222.
8. Powell K.M., Machalek D., Quah T. Real-time optimization using reinforcement learning // Computers & Chemical Engineering. 2020. Vol. 143. P. 107077.
9. Shyalika C., Silva T., Karunananda A. Reinforcement learning in dynamic task scheduling: A review // SN Computer Science. 2020. Vol. 1. No.6. P. 306.
10. Padakandla S. A survey of reinforcement learning algorithms for dynamically varying environments // ACM Computing Surveys (CSUR). 2021. Vol. 54. No.6. P. 1-25.