Q-Learning Based Adaptive Modulation and Coding for Throughput Maximization in Dynamic Wireless Environments

Wei-Shun Liao and Akihiro Nakao
School of Engineering, The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan
(Email: wsliao@g.ecc.u-tokyo.jp, nakao@nakao-lab.org)

***Abstract*—Adaptive Modulation and Coding (AMC) is an essential technology for optimizing performance in modern wireless communication systems. In conventional implementations of AMC, the modulation and coding scheme (MCS) is typically adjusted based on fixed thresholds. However, the setting of these thresholds has a significant impact on system performance, and optimal results may not be achieved. In recent years, AMC methods that automatically adjust MCS by using machine learning and AI methods, particularly by the k-Nearest Neighbor (kNN) are proposed. In this study, we propose a new AMC method that introduces Q-learning, a kind of reinforcement learning algorithm in which an agent interacts with the environment and learns optimal actions based on rewards, to automatically adjust MCS with the goal of maximizing throughput. Simulation results show that the proposed method can provide 44%, 49%, and 81% throughput improvement compared to the kNN method used in previous researches in EPA3 (Extended Pedestrian A Channel with speed 3 km/hr), EVA30 (Extended Vehicular A Channel with speed 30 km/hr), and ETU30 (Extended Typical Urban Channel with speed 30 km/hr) channels respectively, which validates the effectiveness of the proposal.**

***Index Terms*—Adaptive modulation and coding (AMC), throughput maximization, reinforcement learning, Q-learning.**

I. Introduction

In modern wireless communication systems, achieving high throughput and maintaining link reliability is crucial across various scenarios, such as mobile, vehicular, and IoT networks. However, time-varying wireless channels impose significant challenges, degrading performance and requiring systems to adapt in real-time to maintain quality of service (QoS). To effectively address these dynamics, adaptive transmission strategies have become an essential feature of modern communication systems.

Among these adaptive strategies, adaptive modulation and coding (AMC) [1] has been widely adopted to dynamically select the optimal modulation and coding scheme (MCS) according to real-time channel conditions. Traditional AMC implementations highly rely on pre-defined threshold values which are obtained from simulation or field experimental results, stored in lookup tables (LUTs). Although LUT-based methods are computationally efficient, they are sensitive to incorrect threshold choices, which may result in degraded system performance. Therefore, in the literature some methods are proposed to adjust MCS heuristically using some system signaling (e.g., ACK/NAK) as channel condition indications [2], [3], [4], but lack theoretical convergence guarantees.

In the literature, numerous studies have explored the use of machine learning and artificial intelligence (AI) to enhance AMC. For example, machine learning has been applied in AMC for satellite communications [5], underwater communication systems [6], and vehicular networks [7] to support MCS adaptation. Deep learning-based AMC has been explored in MIMO systems [8], while reinforcement learning techniques have been proposed for IoT networks [9] and 5G systems [10]. In addition, autoencoders have emerged as a promising approach for optimizing the channel coding or modulation components of AMC [11]. However, most existing studies primarily focus on QoS-driven adaptation and do not explicitly aim at maximizing throughput. Autoencoder-based methods, for instance, can learn nonlinear mappings between input messages and transmit symbols, to optimize physical-layer processing. Nevertheless, these methods typically require extensive offline training under specific channel conditions and may suffer from limited generalization when channel statistics change.

In this paper, we propose a novel AMC method based on Q-learning, a model-free reinforcement learning algorithm, which enables an agent to interact with the wireless environment and autonomously learn an optimal MCS selection policy. Our approach directly targets throughput maximization, rather than merely satisfying QoS constraints. Distinct from existing works, our method integrates a reinforcement learning objective designed to maximize long-term throughput while maintaining limited computational complexity.

The main benefit of employing a machine learning approach, such as Q-learning, is its ability to automatically adapt to dynamic channel conditions without relying on manually tuned thresholds. Traditional LUT-based or heuristic AMC methods often suffer from performance degradation when pre-defined thresholds are mismatched with the actual wireless environment. In contrast, the machine learning approach can learn optimal decision policies directly from interactions with the environment, achieving better generalization across different channel models. Moreover, although the proposed Q-learning method costs computational resources during the training phase, once the MCS selection policy is trained and deployed, it only requires a constant-time table lookup, making

it both adaptive and computationally efficient for real-time AMC operations.

The key contributions of this paper are as follows:

- We propose a new AMC method based on model-free Q-learning, which directly maximizes long-term throughput under time-varying wireless channels. unlike autoencoder-based AMC, the proposed Q-learning method does not require altering the transceiver structure and still achieves significant throughput gains.
- The proposed method achieves low complexity during policy deployment and only requires constant-time Q-table lookup. In contrast, benchmark kNN-based methods incur significant runtime complexity due to distance computations over training data.
- Extensive simulations under 3GPP channel models (EPA3, EVA30, and ETU30) confirm that our method outperforms benchmark kNN-based AMC by at least 40% improvement in throughput.

The remainder of this paper is organized as follows. Sec. II presents the system model and problem formulation. Sec. III describes the proposed Q-learning-based AMC algorithm. Sec. IV provides the simulation setup and performance evaluation. Finally, Sec. V concludes the paper and discusses future directions.

II. System Description and Study Problem

In this study, we investigate an AMC mechanism in an end-to-end (E2E) wireless communication system, as illustrated in Fig. 1. The system begins by estimating the instantaneous channel condition, which serves as input to the AMC module for selecting the most suitable MCS. Once the optimal MCS is determined, the transmitter encodes and modulates the information accordingly. Since the selected MCS is known to both transmitter and receiver via control channel, the receiver can demodulate and decode the received signal using the appropriate settings.

In existing wireless systems, AMC is typically implemented using an LUT containing fixed MCS switching thresholds, which are obtained by simulations or experiments. However, the conventional methods using fixed thresholds may suffer from performance degradation if the thresholds are not accurately determined under appropriate channel assumptions. In Fig. 2 we show an example to illustrate the problem of performance degradation due to improper thresholds. In Fig. 2, the red solid curve shows the throughput due to optimal MCS switching thresholds which are obtained by exhaustively measuring the throughputs of all available MCSs, while the blue dashed curve shows the throughput due to improper thresholds, which are obtained by offseting each optimal threshold by several dB, to show how threshold values affect the system throughput. From the results shown in Fig. 2, it can be easily seen that the throughput loss is due to the improper threshold settings.

In this work, the MCS set is defined as a discrete set of 16 MCSs, denoted by $\mathcal{S}_m = \{\text{MCS0}, \text{MCS1}, \cdots, \text{MCS15}\}$. Each MCS corresponds to an unique pair of modulation format (BPSK, QPSK, 16QAM, or 64QAM, as summarized

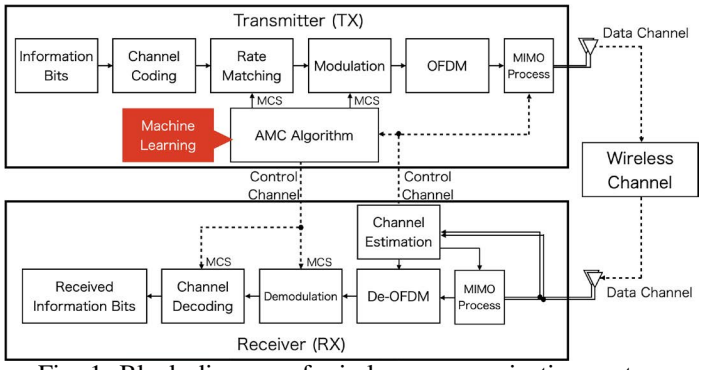

Fig. 1: Block diagram of wireless communication system.

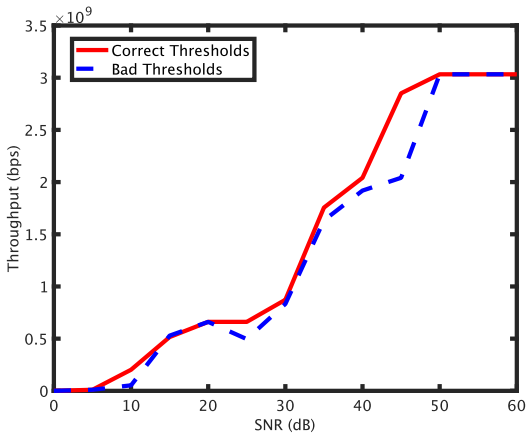

Fig. 2: Performance degradation due to improper MCS threshold.

in Table I) and coding rate. These MCSs are designed to with a range of spectral efficiencies (SEs) similar to 3GPP physical downlink shared channel (PDSCH) settings, which span from 0.2344 to 5.5547 bits/s/Hz (Table 5.1.3.1-1 in [15])), ensuring compatibility with practical communication systems. The full mapping between MCS index, modulation scheme, coding rate, and spectral efficiency is detailed in Table II.

We formulate the AMC optimization as a discrete-time decision-making problem, where the objective is to maximize system throughput by optimal selection of MCS under varying channel conditions. We adopt Q-learning [12], which is a model-free reinforcement learning algorithm, to learn a policy that maps observed channel states to optimal MCS selections. Q-learning is particularly suitable in this problem as it does not require prior knowledge of the channel model, and it can progressively improve its performance over time through interactions with the environment. Furthermore, by focusing on throughput maximization as the primary reward, the proposed approach aligns well with the performance goals of future wireless communication systems, where data rate is often the critical metric of interest.

III. Proposed Method

In this study, we focus on proposing an AMC method which can maximize system throughput by selecting optimal

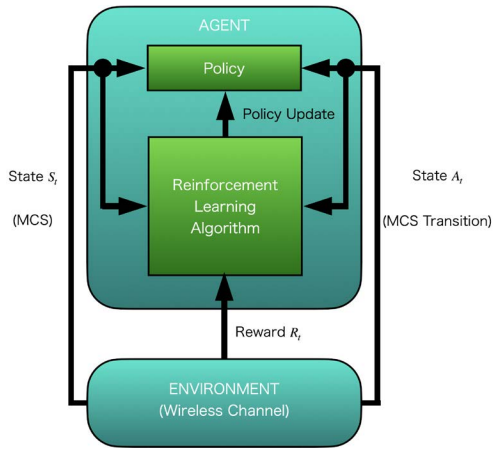

Fig. 3: Block diagram of Q-learning.

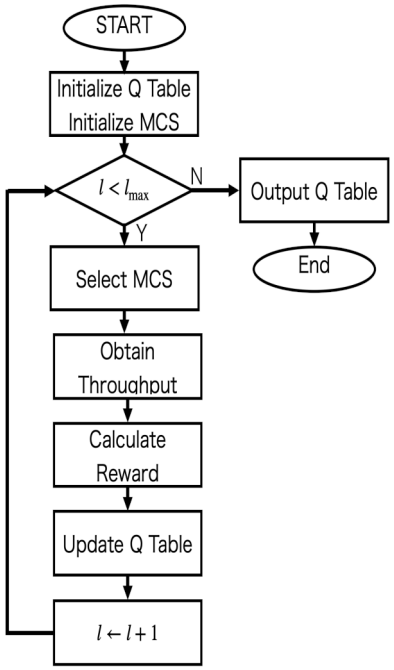

Fig. 4: Flowchart of Q-learning algorithm.

MCSs. For this purpose, we adopt offline Q-learning, which is one kind of reinforcement learning algorithm and suitable for discrete optimization problems, for our algorithm design. The block diagram of Q-learning for the problem in this study is shown in Fig. 3.

As shown in Fig. 3, the mechanism of Q-learning consists of two main elements: agent and environment. The agent selects MCS according to its policy, and the wireless system interacts with environment, i.e., wireless channel in this study, by transmitting signals using selected MCS. The resultant throughput is evaluated in reward function, which is used to update the policy of agent. The workflow of the Q-learning, which is the "Reinforcement Learning Algorithm" part in Fig. 3, is shown in Fig. 4. The learning process is terminated when current epoch count l reaches its maximum limit $l_{\max}$.

After offline learning is finished, the trained policy is equipped in the wireless system, and the AMC is operated to select MCS based on the trained policy. Compared to the existing AMC methods, the proposed method can maximize system throughput in a more effective way.

If the action of agent A_t is decided, the next state S_{t+1} and state-action quality function Q are updated. Then the agent under state S_t selects action A_t, obtains reward R_{t+1}, and transfers to state S_{t+1}. The update of the Q function is expressed as follows,

$$Q(S_t, A_t) \leftarrow (1-\alpha)Q(S_t, A_t) + \alpha\left(R_{t+1} + \beta \max_a Q(S_{t+1}, a)\right), \quad (1)$$

where α, $0 < \alpha < 1$ is learning rate and β, $0 < \beta < 1$ is discount rate. In (1), the transfer from current to next state means the Q function gradually approaches its maximum. Therefore, when a high reward is obtained in a certain state, the reward is propagated with each update. This enables learning of the optimal state transitions. In this study, the state S_t is defined as the MCS selected in time t, and the action A_t is defined as the MCS transition from S_t to S_{t+1}. For example, if state S_t is MCS-1, S_{t+1} is MCS-3, then action A_t is 2.

The reward function used in the proposed method is shown as follows,

$$R_t = K \cdot \left(\frac{\eta(\gamma, S_t) - \eta(\gamma, S_{t-1})}{\left| \frac{C(\gamma) - \eta(\gamma, S_t)}{C_{\max}} \right| + \delta} \right), \quad (2)$$

where γ is current signal-to-noise ratio (SNR), and $\eta(\gamma, S_t)$ is the throughput under the condition that SNR equals γ and MCS is S_t. $C(\gamma)$ represents the channel capacity under SNR γ, and $C_{\max}$ is maximum available rate in this system. Besides, K is a constant used to adjust reward function, and δ is a small constant set to prevent the denominator of expression (2) becoming zero. In reward function (2), the numerator $\eta(\gamma, S_t) - \eta(\gamma, S_{t-1})$ directly measures the throughput improvement or degradation from the previous time slot to the current one, while the denominator $\left|\frac{C(\gamma)-\eta(\gamma,S_t)}{C_{\max}}\right| + \delta$ penalizes throughput values far from the Shannon capacity $C(\gamma)$, scaled by a maximum possible capacity $C_{\max}$. This pushes the agent to move closer to the theoretical limit.

To formalize the Q-learning formulation in our AMC system, we define the state, action, and reward used in the learning process as follows:

- State (S_t): The state at time t is defined as the currently selected MCS index, i.e., $S_t \in \mathcal{S}_m$, where $\mathcal{S}_m = \{MCS\}$. This definition captures the AMC configuration being used in the current transmission.
- Action (A_t): The action corresponds to the transition from the current MCS S_t to the next MCS S_{t+1}. Formally, $A_t = S_{t+1} - S_t$, and the action space $(S)_a$ includes all allowable MCS shifts, such as increasing, decreasing, or maintaining the same MCS.
- Reward (R_t): The reward reflects the benefit of the selected MCS in terms of throughput efficiency. It is defined in (2) and captures both the throughput gain from the previous step and the proximity of the current throughput

Data: Maximum epoch $l_{\max}$, MCS set $S_t \in \mathfrak{S}_m$, MCS transition set $A_t \in \mathfrak{S}_a$, training SNR $\gamma \in \mathfrak{S}_r$ training data $\eta(\gamma, S_t) \in \mathfrak{S}_t$.
Result: Policy $Q(S_t, A_t)$.
Initialization:
Construct training data $\eta(\gamma, S_t)$ by $\mathfrak{S}_r$, $\mathfrak{S}_m$, and $\mathfrak{S}_t$;
Construct training SNR vector;
Choose $S_0 \in \mathfrak{S}_m$;
(PHASE 1: Policy Training by Q-Learning)
while $l \leq l_{\max}$ **do**
  Read training SNR γ;
  Update the Q function by (1) with reward function (2);
  $l \leftarrow l + 1$;
end
(PHASE II: Policy Deployment)
while Received SNR γ_t, current MCS S_t **do**
  Obtain operating SNR by $\hat{\gamma} = \min_{\theta;\gamma_\theta \in \mathfrak{S}_r} ||\gamma_\theta - \gamma_t||$;
  Decide next MCS S_{t+1} by $S_{t+1} = S_t + Q(\hat{\gamma}, S_t)$;
end

Algorithm 1: Proposed algorithm.

to the channel capacity. Specifically, the reward encourages MCS decisions that both improve performance and closely approach the theoretical upper bound.

This formulation allows the agent to iteratively learn a policy that maps each state-action pair to its expected long-term reward, enabling optimal MCS adaptation in dynamic channel conditions.

After offline training Q function iteratively using expressions (1) and (2), the resultant policy is implemented in the wireless system, and the throughput can be maximized by the AMC with trained policy. The proposed method is summarized in Algorithm 1.

During the offline training phase, the agent iteratively updates the Q-function by exploring the state-action space and refining its policy based on the received rewards. The training is performed offline using a predefined SNR vector and a throughput table constructed via simulations. As the training epoch index l approaches the maximum epoch $l_{\max}$ the Q-values converge, meaning that the policy becomes increasingly stable and consistent in selecting MCSs that maximize the long-term expected reward. In our experiments, we observed that with a learning rate $\alpha = 0.001$ and discount rate $\beta = 0.75$, the Q-function typically stabilizes before reaching 1000 epochs. The convergence is evaluated by monitoring the change in Q-values across epochs and verifying that the selected MCS no longer fluctuates significantly under identical SNR conditions. This behavior indicates that the agent has effectively learned a near-optimal MCS switching strategy for the given environment, making it suitable for deployment in real-time AMC applications.

IV. Simulation Results

To validate the effectiveness of the proposed method, we conduct computer simulations for proposed method and compare the performance with the results obtained by k-nearest neighbor (kNN) method [13], which is one of the most used machine learning technologies used for AMC, as a benchmark.

Besides, to generalize the usefulness of the proposed method, in this study three of 3GPP standard channels, EPA3 (extended pedestrian A channel with speed 3 km/hr), EVA30 (extended vehicular A channel with speed 30 km/hr), and ETU30 (extended typical urban channel with speed 30 km/hr) [14], are adopted in the simulations.

The wireless transceiver used in this simulation is an orthogonal frequency division multiplexing (OFDM) system with multi-antenna techniques. The simulation parameters are summarized in Table I. There are 16 MCSs implemented in this system, which are listed in Table II. In this simulation, it is assumed that there is only one single user in the system and effectively one resource block spanning 2048 subcarriers.

The simulation results under EPA3 are shown in Fig. 5. In Fig. 5, the red solid curve represents the throughput resulting from the AMC with optimal MCS switching thresholds which are found by manually checking the training data, and can be viewed as the ideal throughput result under EPA3 environment. Besides, the green dash-dot curve means the averaged throughput result of kNN-based method by averaging the results of $k = 3, 5, 7, 9, 11, 13, 15, 17, 19, 21$, while the black dotted curve shows the maximum throughput of kNN-based method with $k = 5$. In addition, the blue dashed curve is the throughput obtained by the proposed AMC method.

From the simulation results, although the throughput resulting from the proposed method (blue dashed curve) still shows little performance loss comparing with the ideal result obtained by LUT method (red solid curve), it can be observed that the proposed method can achieve the optimal performance under EPA3 environment. However, the best result obtained by kNN-based method (black dotted curve, $k = 5$) shows that, although the kNN-based method can obtain nearly optimal result under high SNR condition, its throughput is dramatically degraded under low SNR condition. Especially, when comparing the sum throughput of proposed method and the best result by kNN-based method in the simulated SNR range, it can be seen that the proposed method can provide about 44% throughput improvement compared to the kNN-based result.

The simulation results under EVA30 and ETU30 environments are shown in Fig. 6 and 7, respectively. In Fig. 6 and 7, the meanings of red solid, green dash-dot, black dotted, and blue dashed curves, are the same as those in Fig. 5. Although the results in Fig. 6 and 7 are obtained under different wireless environments, it can be obviously observed that, the proposed method can achieve nearly optimal performance and, can provide significant throughput improvement compared to kNN-based method. Especially, when comparing the sum throughput of proposed method and the best result by kNN-based method in the simulated SNR range, it can be seen that the proposed method can provide about 49% and 81% throughput improvement compared to the kNN-based results under EVA30 and ETU30 environments, respectively.

From the simulation results shown above, there are several points should be further discussed. Firstly, considering the mechanism of the kNN-based method, when determining the

MCS of test data, the MCS of the k nearest training data points in the vector space is firstly obtained, and the desired MCS is determined based on a majority of the nearest k training data. Based on this principle, it is necessary to adjust the value of k, because an inappropriate choice of k may lead to underfitting or, in particular, performance degradation due to overfitting. The experimental results show that in the kNN-based method, the optimal value of k must be determined through learning for each SNR. Therefore, we can know that kNN-based method cannot automatically help AMC to maximize throughput.

In addition, by observing the best results of kNN-based method, i.e., the black dotted curves in Fig. 5, 6, and 7, it can be seen that the throughput is almost zero when SNR is low. However, by observing the averaged results of the kNN-based method, i.e., the green dash-dot curves in Fig. 5, 6, and 7, it can be seen that, when using different k values, a certain level of throughput can be achieved. This phenomenon shows the difference between the throughput behaviors of individual results and the averaged results. It should be noted that, in this study we have tested multiple k values of kNN-based method and even averaged their results to find the best performance to ensure the fairness of comparison. This fact means that kNN-based method requires adjusting k for each SNR, which implies a lack of adaptability.

Besides, it can be observed that, the kNN-based method outperforms LUT and proposed method in certain SNR regions. This behavior can be attributed to the underlying granularity difference in decision mapping. Specifically, both the LUT and the proposed Q-learning methods rely on discretized SNR bins (e.g., in 10 dB intervals) during training or table construction, which may result in suboptimal MCS selection at bin boundaries. In contrast, the kNN method decides the most likely MCS from the k nearest neighbors, which allows finer-grained interpolation across SNR values. As a result, in some scenarios, kNN may incidentally outperform the discretized schemes.

Furthermore, for the proposed Q-Learning-based method, although in policy training phase (PHASE I in Algorithm 1) it incurs a complexity of $O(l_{\max} \cdot |S| \cdot |A|)$, where $|S|$ denotes the size of state space (number of MCSs) and $|A|$ is the size of action space (number of MCS transitions), the policy deployment phase (PHASE I in Algorithm 1) requires only a simple Q-table lookup with constant-time complexity $O(1)$. In contrast, the kNN-based method requires computing distances to all stored training instances with complexity $O(n \cdot d)$, where n is the size of training dataset and d is the feature dimension ($d = 1$ in this study as only SNR is used). Therefore, the proposed method achieves lower complexity in the online wireless environment, making it more suitable for real-time AMC adaptation.

In addition, it should be noted that, in our current work, we focus on a simplified scenario involving a single user and a single physical resource block (PRB). This setting allows us to conduct a preliminary evaluation of the proposed Q-learning-based AMC scheme under controlled conditions, facilitating direct comparison with baseline methods such as kNN. While this assumption limits the spatial and frequency granularity of MCS adaptation, it serves as a necessary first step toward validating the learning framework and its practical feasibility.

TABLE I: Simulation Parameters.

Parameter	Value
Central Frequency f_c	4.7 GHz
Subcarrier Number	2048
Cyclic Prefix (CP) Length	512
Data Subcarrier Number	2030
Subcarrier Interval	15 kHz
Modulation Schemes	BPSK, QPSK, 16QAM, 64QAM
Channel Coding Scheme	LDPC
Number of Transmit Antennas	4
Number of Receive Antennas	4
Learning Parameters (α, β)	$(0.001, 0.75)$
Reward Parameters (K, δ)	$(10, 10^{-7})$
Maximum Epoch $l_{\max}$	1000

TABLE II: MCS Table.

MCS	Modulation Scheme	Coding Rate	SE
0	BPSK	0.25	0.25
1	BPSK	0.50	0.50
2	BPSK	0.60	0.60
3	BPSK	0.90	0.90
4	QPSK	0.25	0.50
5	QPSK	0.50	1.00
6	QPSK	0.60	1.20
7	QPSK	0.90	1.80
8	16QAM	0.25	1.00
9	16QAM	0.50	2.00
10	16QAM	0.60	2.40
11	16QAM	0.90	3.60
12	64QAM	0.25	1.50
13	64QAM	0.50	3.00
14	64QAM	0.60	3.60
15	64QAM	0.90	5.40

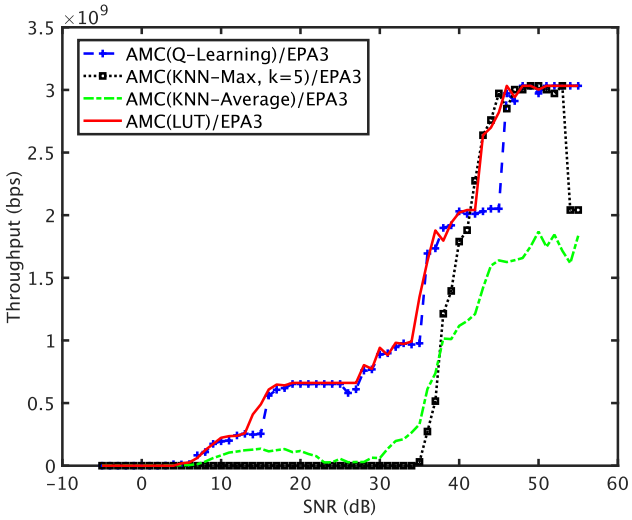

Fig. 5: The simulation results for EPA3 channel.

V. Conclusion

In this work, we proposed a Q-learning-based AMC scheme for maximizing system throughput in time-varying wireless channels. Unlike conventional LUT-based methods or recent

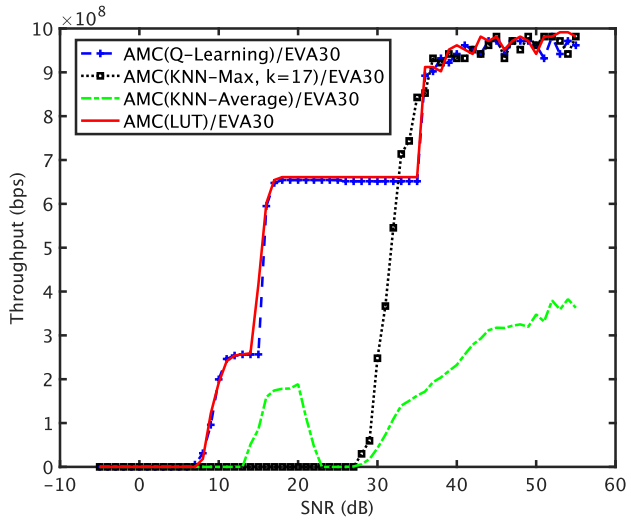

Fig. 6: The simulation results for EVA30 channel.

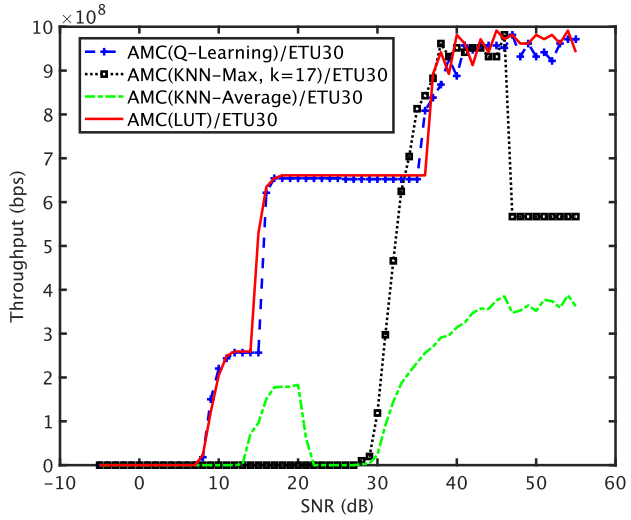

Fig. 7: The simulation results for ETU30 channel.

machine learning approaches such as kNN-based AMC, our method formulates MCS selection as a reinforcement learning problem and learns a policy offline to make MCS decisions with low online complexity.

Through extensive simulations under 3GPP EPA3, EVA30, and ETU30 channel models, it can be seen that the proposed method achieves significant throughput improvements of 44%, 49%, and 81% compared to a benchmark kNN-based AMC scheme. These results highlight the practicality and adaptability of Q-learning for AMC under realistic channel variations.

It should be noted that in this work, we focused on a simplified single-user, single-PRB setting to validate the proposed learning framework under controlled conditions. An important next step is to extend the Q-learning AMC framework to multi-user and multi-PRB scenarios. Due to the low online complexity and modular design, such extension is both feasible and promising, and we identify it as our future research direction.

In summary, this study presents a reinforcement learning-based AMC solution that stands out from existing approaches in terms of both performance and complexity. The combination of offline policy training and low-complexity deployment results in a method that is not only effective, but also scalable and practical for next-generation wireless systems.

ACKNOWLEDGEMENT

This work was partly supported by NICT, Grant Number 09101, Japan, and JST ASPIRE, Grant Number JPMJAP2323, Japan.

REFERENCES

[1] A. Goldsmith, *Wireless Communications*, Cambridge University Press, 2005.

[2] A. Kamerman and L. Monteban, "WaveLan-II: a high-performance wireless LAN for the unlicense band," *Bell Labs Technical Journal*, pp. 118–133, Summer 1997.

[3] S. Falahati and A. Svensson, "Hybrid type-II ARQ scheme with adaptive modulation systems for wireless channels," *IEEE VTS Vehicular Technology Conference (VTC-Fall)*, Amsterdam, The Netherlands, Sept. 19–22 1999.

[4] H.-J. Su and L.-W. Fang, "A simple adaptive throughput maximization algorithm for adaptive modulation and coding systems with hybrid ARQ," *International Symposium on Communications, Control, and Signal Processing (ISCCSP)*, Marrakech, Morocco, Mar. 13–15 2006.

[5] D. Lee, Y. G. Sun, I. Sim, J.-H. Kim, Y. Shin, and D. I. Kim, "Neural episodic control-based adaptive modulation and coding scheme for inter-satellite communication link," *IEEE Access*, vol. 9, pp. 159175–159186, Nov. 2021.

[6] J. Byun, Y.-H. Cho, T. Im, H.-L. Ko, K. Shin, and J. Kim, "Iterative learning for reliable link adaptation in the internet of underwater things," *IEEE Access*, vol. 9, pp. 30408–30416, Feb. 2021.

[7] Y. Ji, G. Zhang, J. Huang, J. Yang, G. Gui, and H. Sari, "Deep learning for adaptive modulation and coding with payload length in vehicle-to-vehicle communications systems," *IEEE Vehicular Technology Conference (VTC-Fall)*, Norman, OK, USA, Sept. 27–30 2021.

[8] Q. An, M. Zafari, C. Dick, S. Segarra, A. Sabharwal, and R. Doost-Mohammady, "ML-based feedback-free adaptive MCS selection for massive multi-user MIMO," *Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, USA, Oct. 29–Nov. 1 2023.

[9] S. Mashhadi, N. Ghiasi, S. Farahmand, and S. M. Razavizadeh, "Deep reinforcement learning based adaptive modulation with outdated CSI," *IEEE Communications Letters*, vol. 25, no. 10, pp. 3291–3295, Oct. 2021.

[10] L.-S. Chen, C.-H. Ho, C.-C. Chen, and S.-Y. Kuo, "Learning scheme for adaptive modulation and coding in 5G new radio," *International Conference on System Reliability and Safety (ICSRS)*, Venice, Italy, Nov. 23–25 2022.

[11] Y. Jiang, H. Kim, H. Asnani, S. Kannan, S. Oh, and P. Viswanath, "Turbo autoencoder: deep learning based channel codes for point-to-point communication channels," *Conference on Neural Information Processing Systems (NeurIPS)*, Vancouver, Canada, Dec. 8–14 2019.

[12] R. S. Sutton and A. G. Barto, *Reinforcement Learning, second edition: An Introduction*, Bradford Books, 2018.

[13] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, Aug. 1992.

[14] 3GPP TS 36.104, *Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) Radio Transmission and Reception*, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network.

[15] 3GPP TS 38.214, *Physical Layer Procedures for Data (Release 18)*, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; NR.