Channel Selection Strategy in 5g Based on Contextual Mab

Xiuxian Peng

College of Computer Science, Beijing University of Technology, Beijing, 100124, China

Abstract. 5th Generation (5G) has become an essential part of humans' lives, much research has been done in network areas like 2G, Long-Term Evolution (LTE) channel selection, but less research has been done on 5G channel selection; hence, this study aims to provide an insight into the 5G channel selection issue. This study uses a contextual Multi-armed Bandit (MAB) algorithm to select the best arm based on contextual features, and the main contribution is constructing a methodology to select a 5G channel in a real situation to improve people's quality of life. The conclusion drawn from the experiment is that the best arm can be selected well through context features, but due to the number of channels in the dataset is not sufficient and the algorithm strategy and parameter settings is not optimal, the cumulative regret shows a linear trend rather than a sublinear trend and eventually reaches about 3000. However, the selection strategy used in the algorithm is very effective, causing the optimal arm selection rate to rise sharply in the early stage and quickly reach about 0.99.

1 Introduction

As 5G technology becomes more and more mature, 5G is widely used among people and has become an essential part of people's lives. Over the past decade or so, the amount of wireless network data used by people has increased dramatically, even more than 150 times [1]. According to the research done by Darijo Raca et al., people have greater demands for 5G throughput, and these demands require a high bandwidth and low delay 5G channel [2]. Hence, it is crucial to select channels with better performance in daily life.

At present, studies about selecting channels for wireless networks such as UMTS and LTE are based on classic MAB algorithms, which do not consider the features of channels at the time, contributing to the difference between the experimental result and the real situation. Features such as channel interference intensity, bandwidth utilization, historical delay, signal strength, etc., need to be comprehensively considered [3]. Based on the analysis of these features, the most suitable channel at present is selected and provided to users. Hence, research on 5G channels and the environment of these channels is critical to improve people's quality of life and work efficiency. This work aims to use a contextual MAB algorithm that considers features of the channel to select 5G channels.

AFpxx@emails.bjut.edu.cn

2 Related Work

In a decade, research on the network channels selection based on MAB has varied a lot, trying to find the best way to simulate reality and select channels.

MAB: In the Multi-armed Bandit algorithm, each arm is considered as a selection. After pulling one arm, a reward will be provided to the operator, and the better the choice the operator made, the better the reward will be provided. Hence, operators need to find a way to maximize the total reward for hundreds or thousands of selections, which aims to simulate people's choices, in the experiment. In 5G channel selection, channels are arms, throughput, signal strength, etc., are rewards, and use classic MAB algorithms such as Explore-Then-Commit (ETC), Upper Confidence Bound (UCB), and Thompson Sampling (TS) to maximize the reward [4].

Contextual MAB: The contextual Multi-armed Bandit algorithm takes the environment and features of arms into account. Hence, in 5G channel selection, the features of the present channel, such as channel interference intensity, bandwidth utilization, historical delay, signal strength, etc., are defined as context. Contextual MAB dynamically selects the optimal arm, which is more suitable for selecting a 5G channel than classic static MAB [5].

R. Desai Hinal et al. conducted research on channel selection by defining three different strategies: one is fixed channel selection, another is random channel selection for a fixed interval, and the last one is a combination of both. Finally, the conclusion is that the second strategy has the highest average throughput [6]. Feng Li et al. proposed the definition that one round with only one participant and one round with multiple participants, at the same time, divided MAB models into centralized and decentralized by whether a central infrastructure exists. Researchers also indicated the algorithms for centralized and decentralized situations [7].

Zhiwu Guo et al. introduced a MAB model defined as a Fair Probabilistic Multi-Armed Bandit (FP-MAB), asserting that the fairness among arms is a critical objective. Researchers aimed to maximize the reward of each arm or maximize the total reward. Researchers also proposed two algorithms defined as Fair Probabilistic Explore-Then-Commit (FP-ETC) and Fair Probabilistic Optimism in the Face of Uncertainty (FP-OFU). FP-ETC is based on the classic MAB ETC algorithm, but differs from ETC in the exploitation phase. FP-ETC selects arms based on possibility rather than selecting based on the result of the exploration phase. Researchers concluded that FP-ETC and FP-OFU perform better than maxmin-UCB and classic MAB algorithms through experiments on a real network environment dataset [8]. Though many related works have been done in the channel selection realm, research on how contextual MAB performs in 5G channel selection is still a mystery. Hence, this work aims to apply contextual MAB on the channel selection issue and evaluate the performance through channel features.

3 Methodology

In this study, the methodology section is divided into three parts: data preprocessing, model construction, and result evaluation. In the data preprocessing part, how to preprocess the dataset and how features are selected as context, and how reward is defined will be explained. In the model construction part, how parameters are set and how the arm is defined, and how the model is updated will be illustrated. In the result evaluation part, how evaluation metrics are ensured and what results indicate will be demonstrated.

3.1 Data Preprocessing

The dataset collected by Darijo Raca et al. aims to provide a large, public, and contextual dataset for research on 5G channel selection and other aspects. The dataset includes date, 5G channels, series metrics of channel's quality, user position and mobile status, download and upload bitrate, etc. [2]. To better simulate a real situation, this study selects the data collected when the user is downloading in a dynamic status, such as driving. Due to the dataset is divided by date, the first step is to integrate the dataset and set invalid values. Reference Signal Receiving Power (RSRP) used to measure signal strength, Reference Signal Receiving Quality (RSRQ) used to measure the quality of signal, Signal-to-Noise Ratio (SNR) used to evaluate the quality of signal, Channel Quality Metric (CQI) used to indicate the quality of channel and Speed used to reflect moving status of user are defined as context because these features indicate quality of network and user mobile status. Position of the user is not set as context because position may be too specific, leading to overfitting, and the current features, such as RSRP, RSRQ, etc., already imply the location information. To improve the stability and robustness of the model, contextual features are standardized to a similar extent. The specific method used to standardize is RobustScaler rather than StandardScaler imported from the sklearn library, because RobustScaler is not sensitive to abnormal values, hence, RobustScaler is more suitable for wireless communication data, which may contain burst noise and abnormal values. In addition, this study defines reward as download bitrate and normalizes it to facilitate model learning and comparison. Similar to the contextual features, reward is also standardized through RobustScalerand and normalized through the formula as follows:

$$y = \frac{y - min(y)}{y - max(y)} \tag{1}$$

y represents the matrix of the reward download bitrate.

3.2 Model Construction

This study mainly uses the Linear Upper Confidence Bound (LinUCB) algorithm with partial improvement as a methodology to select an arm. This study defines three parts of the algorithm: metrics initialization, arm selection, and context updating. Each arm has a covariance matrix to record the covariance relationship between features and calculate the size of the confidence interval. Each arm also has two vectors, one for recording the correlation between features and rewards, and the other for representing the weight of each feature on the reward obtained by the least squares method and used to predict the expected reward under a given context. In addition to these metrics, a parameter defined as alpha is used to balance exploration-exploitation; the bigger the alpha, the more the algorithm tends to select new or uncertain arms. On the contrary, the smaller the alpha, the more the algorithm tends to use arms that perform well.

In the arm selection section, this study utilizes the UCB algorithm and calculates the UCB value of each arm and selects the arm has the maximum UCB value as the best arm. In the update section, the covariance matrix and related vectors are updated, rewards of each arm and times of each arm are selected are recorded as well [9, 10].

3.3 Result Evaluation

This study sets four metrics to evaluate the result of the experiment: cumulative reward, cumulative regret, optimal arm selection rate, and average reward. Cumulative reward

reflects the overall performance of the algorithm and shows the total benefits accumulated over time. The higher the value, the better the algorithm's effect. The slope can reflect the learning speed. Using it as an evaluation metric can intuitively show the long-term benefits of the algorithm. The formula is presented as follows:

$$Reward_{cum}(t) = \sum_{i=1}^{t} r_i \quad (2)$$

i represents the current round number, t represents the present round number, r_i represents the reward obtained in round i.

Cumulative regret measures the gap between the algorithm and the optimal strategy and reflects the learning effect of the algorithm. The lower the value, the closer the algorithm is to the optimal. The growth rate indicates the learning efficiency. Using it as an evaluation metric can evaluate the convergence of the algorithm. The formula is presented as follows:

$$Regret_{cum}(t) = \sum_{i=1}^{t} (r_i^* - r_i) \quad (3)$$

i represents the current round number, t represents the present round number, r_i^* represents the max reward in round i, r_i represents the reward obtained in round i.

The optimal arm selection rate reflects the accuracy of the algorithm in selecting the optimal arm and shows the effect of balancing exploration-exploitation. The higher the value, the more accurate the decision. Using it as an evaluation metric can evaluate the effect of the strategy. The formula is presented as follows:

$$Reward_{avg}(t) = \frac{\sum_{i=1}^{t} r_i}{t} \quad (4)$$

i represents the current round number, t represents the present round number, r_i represents the reward obtained in round i.

The average reward reflects the immediate performance of the algorithm and shows the short-term learning effect. Using it as an evaluation metric can help observe the stability of the algorithm and reflect the algorithm's ability to adapt to environmental changes. After each time of selection, all these metrics will be updated. The formula is presented as follows:

$$P_{opt}(t) = \frac{\sum_{i=1}^{t} I(a_i = a_i^*)}{t} \quad (5)$$

i represents the current round number, t represents the present round number, $I()$ is the indicator function, 1 when the optimal arm is selected, 0 when not, a_i represents the arm selected in round i, a_i^* represents the best arm in round i.

To better observe the result, this study imports matplotlib library to plot figures of each metrics, each figure reflects different effects, cumulative reward demonstrates the growth tendency of reward, cumulative regret demonstrates the effect of learning, the optimal arm selection rate demonstrates the convergence of the strategy, and the average reward demonstrates real-time performance.

4 Experimental Evaluation and Results

4.1 Setup and Dataset

This part demonstrates the basic setup of the experiment and briefly introduces the dataset used in the experiment. The setup section includes hardware and software used in the experiment, and some crucial settings of parameters. These settings are for ensuring the experiment can obtain good performance and results. Significant hardware details are listed in Table 1, and significant software details are presented in Table 2. Three core parameters need to be set, respectively, the exploration parameter alpha, the number of arms, and the dimension of the feature vector. After exploration, alpha is set to 0.2, the number of arms is set to 3 according to the data set, and the dimension of the feature vector is set to 5.

The dataset section introduces the dataset briefly. This dataset is collected from three different ways: streaming on Amazon and Netflix, and downloading from the website. Each of them is divided into two parts: dynamic and static. This dataset has a total of 14,000 records, with a period from December 2019 to February 2020, a sampling frequency of one second, and contains information from three channels. The RSRP ranges from -116dBM to -50dBM, with an average of -82.78dBM, the RSRQ ranges from -22dB to -5dB, with an average of -11.94dB, the SNR ranges from -14dB to 35dB, with an average of 5.29dB, the CQI ranges from 1 to 15, with an average of 11.36, and the Speed ranges from 0 to 75km/h, with an average of 30.17km/h. To better simulate the real situation, this research chooses the downloading and dynamic data. This dataset includes context such as timestamp, location, speed, channel, RSRP, etc. This study chooses RSRP, RSRQ, SNR, CQI, and Speed as contextual features.

Table 1 Hardware setup

Hardware	Version
Operating System	macOS Sonoma 14.3
Graphics Processing Unit	Apple M3 Pro
Memory	18 GB

Table 2 Software setup

Software	Version
python	3.11.7
numpy	2.1.1
pandas	2.2.3
scikit-learn	1.6.1
matplotlib	3.10.1

4.2 Cumulative Reward

According to the figure produced by the experiment, figure is presented in Fig. 1, the curve shows a continuous linear growth tendency, without obvious changes or inflection points, and finally reaches a cumulative reward of about 8500. The linear growth of the curve shows that the average reward obtained by the algorithm at each time step is relatively stable, and the lack of obvious inflection points indicates that the algorithm quickly finds a good strategy, and the environment of users is relatively stable without dramatic changes, and performs well in balancing exploration-exploitation. The fundamental reason for the result is that the

methodology precisely predicts the expected reward and sets parameter alpha appropriately, and ucb strategy balances exploration-exploitation effectively.

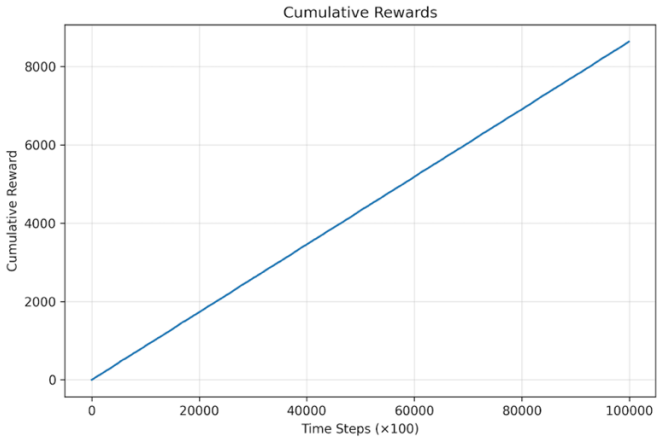

Fig. 1 Cumulative Rewards (Picture credit: Original)

4.3 Cumulative Regret

Through observing the result produced by the experiment, a figure is presented in Fig. 2, the conclusion is that the cumulative regret curve has a linear growth trend instead of a sublinear one, and its growth rate remains stable, with the final cumulative regret value reaching about 3000. An effective Contextual MAB algorithm should present a sublinear trend instead of a linear one, because this indicates the algorithm can learn the best strategy better with training. The linear growth trend of the cumulative regret indicates that the strategy used to select arm has not reached the theoretical optimum, this is because the algorithm cannot select the best arm for each time and users are in a moving status which contributed to the changing environment, hence continuous exploration is needed to adapt to environmental changes and strategy need improvement. The fundamental reason for the result is that the parameter alpha may be too large, the environment is too noisy, and the parameter setting is not optimal.

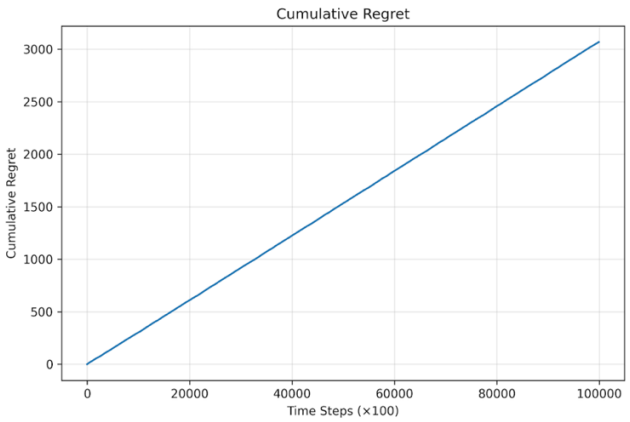

Fig. 2 Cumulative Regret (Picture credit: Original)

4.4 The Optimal Arm Selection Rate

After analyzing the result obtained from the experiment, the figure is presented in Fig. 3, the study comes to conclusion that the curve rises very quickly in the early stage, quickly reaches and maintains above 0.99, and does not fall in the later stage. The rapid rise in the early stage is due to the efficient learning ability of the LinUCB algorithm and the context information that provides useful features for the selection of the best arm and the appropriate setting of the exploration parameter alpha. The reason for the stability in the later stage is that the algorithm has learned the relationship between features and rewards, and the exploration mechanism effectively prevents over-exploration. The fundamental reason is that the parameter update mechanism is effective and quickly converges to the optimal strategy.

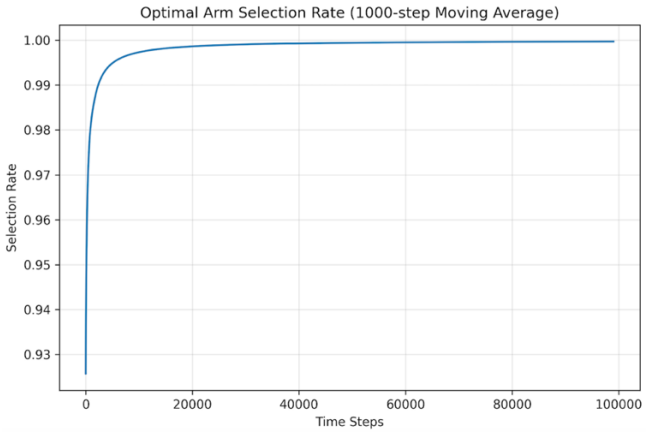

Fig. 3 Optimal Arm Selection Rate (Picture credit: Original)

4.5 The Average Reward

Through the result produced by the experiment, a figure is presented in Fig. 4, it is observed that the fluctuations are obvious in the early stage, converge quickly in the middle stage, and fluctuate slightly around 0.0865 in the late stage. The reason for the large fluctuations in the early stage is that the algorithm is actively exploring different arms while the parameters have not been precisely estimated, and the sample data is insufficient. The reason for the stability in the late stage is that the algorithm has accumulated enough data in the late stage, and the parameter estimation is much precise than at the beginning, and an appropriate strategy has been found to select the arm.

Fig. 4 Average Reward (Picture credit: Original)

5 Conclusion

This study concludes that the cumulative reward shows a rapid growth trend, which indicates that the algorithm can learn and gradually improve the selection. The cumulative regret shows a linear growth rather than a sublinear growth, which indicates that the performance of the algorithm is not ideal. The optimal arm selection rate quickly and nearly reaches 1, which indicates that the algorithm has a good learning strategy. The average reward shows a trend of rising in the early stage and converging in the later stage, which indicates that the algorithm has a good learning effect. The current methodology still has certain limitations. Due to the insufficient setting of parameters and preprocessing, the cumulative regret does not show an ideal tendency like sublinear, and it is contradict that the optimal arm selection rate reaches 1 but cumulative regret is still increase linear instead of sublinear, the reason may be that the expected reward estimated by the current model is used as the criteria for judging the optimal arm but this estimated value will change as the model is updated and the environment changes, at the same time, the regret calculation uses the optimal reward estimated by the model, while the optimal arm selection rate is based on the optimal arm predicted by the model, which may lead to contradictions between the indicators. Hence, this study may aim to improve the algorithm through adjusting the preprocessing strategy and parameter settings in future work. At the same time, other exploration strategies such as TS can be considered to optimize the algorithm, and adaptive mechanisms can also be introduced to adjust the parameters through training.

References

1. Andrews, J.G., Buzzi, S., Choi, W., Hanly, S.V., Lozano, A., Soong, A.C., Zhang, J.C.: 'What will 5G be?', IEEE Journal on Selected Areas in Communications, 2014, 32, (6), pp. 1065-1082
2. Raca, D., Leahy, D., Sreenan, C.J., Quinlan, J.J.: 'Beyond throughput, the next generation: A 5G dataset with channel and context metrics'. Proc. 11th ACM Multimedia Systems Conference, May 2020, pp. 303-308
3. Boldrini, S., De Nardis, L., Caso, G., Le, M.T., Fiorina, J., Di Benedetto, M.G.: 'muMAB: A multi-armed bandit model for wireless network selection', Algorithms, 2018, 11, (2), pp. 13

4. Kuleshov, V., Precup, D.: 'Algorithms for multi-armed bandit problems', arXiv preprint arXiv:1402.6028, 2014
5. Lu, T., Pál, D., Pál, M.: 'Contextual multi-armed bandits'. Proc. Thirteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, March 2010, pp. 485-492
6. Hinal, R.D., Patel, N.B., Suratwala, C.P.: 'Channel and rate selection strategy in cognitive radio network using multi-arm bandits (MAB)'. Proc. 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), May 2018, pp. 232-237
7. Li, F., Yu, D., Yang, H., Yu, J., Karl, H., Cheng, X.: 'Multi-armed-bandit-based spectrum scheduling algorithms in wireless networks: A survey', IEEE Wireless Communications, 2020, 27, (1), pp. 24-30
8. Guo, Z., Zhang, C., Li, M., Krunz, M.: 'Fair Probabilistic Multi-Armed Bandit with Applications to Network Optimization', IEEE Transactions on Machine Learning in Communications and Networking, 2024
9. Yang, W., Cai, L., Shu, S., Sepahi, A., Huang, Z., Pan, J.: 'QoS-driven Contextual MAB for MPQUIC Supporting Video Streaming in Mobile Networks', IEEE Transactions on Mobile Computing, 2024
10. Li, X., Zhou, C., Liang, Z., Yu, Q., Chen, X., He, Z.: 'UCB-Based Route and Power Selection Optimization for SDN-Enabled Industrial IoT in Smart Grid', Wireless Communications and Mobile Computing, 2022, 2022, (1), pp. 7424854