This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LNET.2020.3000334, IEEE Networking Letters

Link-level Throughput Maximization Using Deep Reinforcement Learning

Saeed Jamshidiha*, Vahid Pourahmadi*, Abbas Mohammadi*, Mehdi Bennis†
* Department of Electrical Engineering, Amirkabir University of Technology, Tehran, Iran
† Centre for Wireless Communications, University of Oulu, Finland

***Abstract*—A multi-agent deep reinforcement learning framework is proposed to address link level throughput maximization by power allocation and modulation and coding scheme (MCS) selection. Given the complex problem space, reward shaping is utilized instead of classical training procedures. The time-frame utilities are decomposed into subframe rewards, and a stepwise training procedure is proposed, starting from a simplified power allocation setup without MCS selection, incorporating MCS selection gradually, as the agents learn optimal power allocation. The proposed method outperforms both weighted minimum mean squared error (WMMSE) and Fractional Programming (FP) with idealized MCS selections.**

***Index Terms*—Resource Allocation, Link-level Throughput, Reinforcement Learning, DDPG.**

I. Introduction

THE problem of resource allocation in cellular networks has been extensively investigated and different algorithms have been proposed for different objectives; e.g. for power allocation [1] and spectrum access [2]. In the context of densely deployed intelligent wireless networks, most of the research has focused on power allocation [3], and different model-based optimization algorithms, such as weighted minimum mean squared error (WMMSE) [1] and fractional programming (FP) [4], have been proposed. However, these methods generally rely on perfect channel state information (CSI) of the whole network, which is a strong and impractical assumption; futhermore, delayed or partial CSI is shown to deteriorate their performance. Another approach is to use machine learning algorithms, such as deep reinforcement learning, which has seen a surging interest in the context of wireless networks. Their applications include resource allocation and management, e.g. power allocation [3], [5], spectrum management [6], and caching and beamforming [7].

Different reinforcement learning (RL) algorithms have recently been compared to WMMSE and FP for power allocation [3, 5, 8] and were shown to outperform them, especially with imperfect or delayed CSI [5]. RL algorithms are also less computationally expensive in comparison to WMMSE and FP [3]. In spite of the desirable performance of the RL methods, the studied scenarios are generally not realistic; for example, their objective is to maximize sum capacity of the network, which cannot be achieved in practice due to limited code block lengths and the inefficiency of realistic modulation schemes.

In this paper, we consider a more realistic setting by maximizing the link level throughput in a cellular network, which is defined as the sum of users' data rates which depends on their modulations, coding rates, and transmission powers. The main challenge, however, is that *due to the complexity of the state-action space* of such a realistic scenario, *reinforcement learning algorithms*, with *common exploration strategies*, *do not converge to optimal policies*, and fail to outperform random resource allocation.

To solve the shortcomings in exploration while keeping the model practical, we have proposed: **a)** A multi-agent RL framework where the agents (each of the eNB-UE links) make decisions based on partial network CSI; i.e. their own CSI and the information from their immediate neighbors. **b)** To accommodate for more complex requirements such as fairness among users, a long-term utility function (instead of a single shot one) defined over multiple subframes (time slots). The idea of *reward shaping* has been applied to solve the slow convergence problem in these settings. **c)** Inspired by [9], we propose a step by step training procedure which starts from a simpler power allocation task without MCS selection and gradually becomes more difficult, culminating in joint power allocation and MCS selection.

The remainder of this paper is organized as follows: the system model is defined in Section II, the proposed method, along with the training strategy is discussed in Section III, Simulation setup is provided in Section IV, and the results are discussed in Section V. Section VI concludes the paper.

II. System Model

The downlink transmission of a cellular network with N cells, K users per cell ($M = N \times K$ users in total), cell size (eNB to eNB distance) d_{max}, and minimum allowable eNB to UE distance d_{min} is considered. The positions of eNBs are deterministic (fixed). The users are located randomly in the habitable zones of the cells (K users in each cell).

While most previous studies consider single shot scenarios, a frame-based scheduling scheme is analyzed here where each frame consists of T subframes (time slots). The goal of frame-based scheduling is to maximize the total transmission rate of the users during each frame (T subframes). This model enables us to study the fairness of a scheduling algorithm along with throughput analysis. Fairness study is not possible in single shot cases where transmission are only analyzed during one single subframe.

Rayleigh fading and log-normal shadowing are taken into account; resulting in channel gain $h_i[t]$ between user i and its corresponding eNB, in subframe (time slot) t. A random

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LNET.2020.3000334, IEEE Networking Letters

velocity vector is assigned to the users, which along with their initial positions determines their positions in the following subframes. During each subframe, the channel gains between all UEs and eNBs are calculated based on their distances, log-normal shadowing and Rayleigh fading. Channel gains are assumed to be constant during one subframe but varying from one subframe to another. $h_{ij}[t]$ denotes the channel gain between user i and the eNB to which user j is assigned.

The data rate of *link* i (between user i and its associated eNB) in subframe t, using modulation order $\mathfrak{M}_i[t]$ and coding rate $\mathfrak{C}_i[t]$ can be calculated by [10]:

$$r_i[t] = \mathfrak{M}_i[t].\mathfrak{C}_i[t].(1 - BLER(\gamma_i[t])), \tag{1}$$

where $BLER(\gamma_i[t])$ is the block error rate of the selected modulation/coding scheme at SINR $\gamma_i[t]$. SINR $\gamma_i[t]$, itself, depends on the assigned power to link i ($P_i[t]$), $h_i[t]$, the power assigned for each of the neighboring users j, and $h_{ij}[t]$.

In order to simulate a realistic combination of modulation schemes and code rates ($\mathfrak{M}_i[t]$ and $\mathfrak{C}_i[t]$), the pairs defined in 3GPP LTE release 11 standards have been used. Under release 11 standards of the LTE, 15 standardized Modulation and Coding Scheme (MCS) are defined [11]. In this paper, $m_i[t]$ denotes the MCS that link i uses at time t. e.g. $m_2[3] = 4$ means that in subframe 3, link 2 will use the $4th$ modulation/coding pair defined in the LTE standard, which is QPSK modulation and with coding rate $308/1024$ [11].

In order to optimize both throughput and fairness, we aim to maximize α-fair [12] link level throughput of this network in each frame. The α-fair utility function is defined as follows:

$$U = \sum_{i=1}^{M} \frac{D_i[T]^{1-\alpha}}{1-\alpha}, \tag{2}$$

where $0 \leq \alpha \leq 1$, and $D_i[t]$, $i \in \{1, ..., M\}$for $t \in \{1, ..., T\}$ is the amount of data transmitted on link i upto time t (sum of the data rate of link i upto subframe t):

$$D_i[t] = D_i[t-1] + r_i[t] \tag{3}$$

$$D_i[0] = 0. \tag{4}$$

When $\alpha = 0$, (2) reduces to sum of link level throughput. When $\alpha = 1$, U is not well-defined (its denominator is zero), but maximizing it is equivalent to maximizing $U - \frac{1}{1-\alpha}$, since $\frac{1}{1-\alpha}$ is constant with respect to powers and MCS's, and by applying l'Hôpital's rule, we can obtain [12]:

$$\begin{aligned} &\underset{P_i[t], m_i[t]}{\text{argmax}} \lim_{\alpha \to 1} \left(\sum_{i=1}^{M} \frac{(D_i[t])^{1-\alpha} - 1}{1-\alpha}\right) \\ &= \underset{P_i[t], m_i[t]}{\text{argmax}} \left(\sum_{i=1}^{M} \ln(D_i[t]) - 1\right) \\ &= \underset{P_i[t], m_i[t]}{\text{argmax}} \left(\sum_{i=1}^{M} \ln(D_i[t])\right). \end{aligned} \tag{5}$$

The goal of this paper is to solve the following optimization problem:

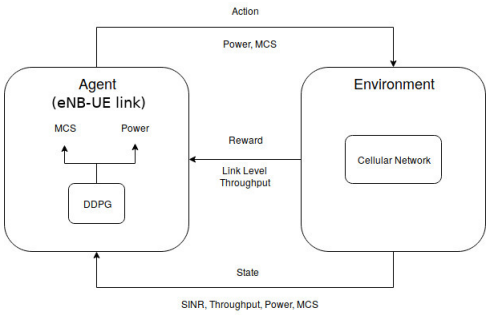


Fig. 1: Proposed System

$$\begin{aligned} \max_{P_i[t], m_i[t]} \quad & \sum_{i=1}^{M} \frac{(D_i[t])^{1-\alpha}}{1-\alpha} \\ \text{subject to} \quad & 0 \leq P_i[t] \leq P_{max} \\ & m_i[t] \in \{1, 2, 3, ..., 15\} \end{aligned} \tag{6}$$

where P_{max} is the maximum possible transmission power and the second constraint ensures that the all links select their modulation orders and coding rates in accordance with the standard LTE pairs.

III. The Proposed RL Framework

A multi-agent reinforcement learning system is proposed to solve the optimization problem (6), where each agent is a Deep Deterministic Policy Gradients (DDPG) agent which determines the power and MCS of each eNB-UE link (Fig. 1). As stated earlier, contrary to *network capacity maximization*, link level throughput maximization cannot be solved using classical RL training schemes. The environment and the action space of the framework are briefly defined below. Then in order to explain the techniques utilized to improve the convergence of RL training, the state (observation) space, reward function, and the stepwise training procedure are described in detail in subsections A through C, respectively.

Environment: The cellular network defined in Section II serves as the environment of the proposed reinforcement learning framework. The actions (power and MCS) of the agents are applied to the environment, and the rewards and the next states (as described below) are returned to the agents.

Action space: The action space is two dimensional and corresponds to 1) power level and 2) MCS scheme assigned to each link. The MCS output of the network is mapped to the modulation indices and code rates of the LTE rel. 11 standard [11]. A non-integer MCS output is regarded as a probabilistic selection of one of the two integer values that it falls in between.

A. State (Observation) Space

If the states contain the full CSI of all users, the problem can easily be modeled as a Markov decision process (MDP). However, such state definition, requires large signaling overhead to exchange CSI between eNBs. In this study, the state space is defined in a way that would provide the agents with enough information to solve the problem, while keeping

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LNET.2020.3000334, IEEE Networking Letters

the state space as local and low dimensional as possible. To simplify notations, $I_{ij}[t]$ is defined as:

$$I_{ij}[t] = \ln(1 + \frac{h_i^2[t]}{h_j^2[t]}). \tag{7}$$

$I_{ij}[t]$ provides information about the coupling between link i and j in subframe t, which contributes to the interference received by user i from transmission aimed at user j.

The state of agent i in subframe t is then defined as:

$$\begin{aligned}\mathbf{S}_i[t] = \big[&h_i^2[t], I_{i1}[t], \cdots, I_{iM}[t], \\ &P_i[t-1], m_i[t-1], r_i[t-1], \rho_i[t-1], t\big].\end{aligned} \tag{8}$$

Note that with the exception of $\rho_i[t-1]$, all terms of (8), depend only on the status of the agent in the last subframe. $\rho_i[t-1]$ (defined in section III-B) is proportional to the accumulated rewards that an agent receives from subframe zero to subframe t, and provides the agents with a memory of the previous subframes. An agent with a high $\rho_i[t-1]$ might decide to refrain from transmissions in future subframes in favor of other agents to accommodate for fairness.

B. Reward Function - Reward shaping idea

In the following section, two extremes of α-fair utility function ($\alpha = 0$ and $\alpha = 1$) are analyzed, and reward shaping is used to distribute the frame reward among subframes.

1) $\alpha = 1$: The α-fair utility function with $\alpha = 1$ results in equation (5). This utility forces all agents in the network to transmit data at some subframe during the frame, thereby ensuring fairness. Due to the long-term nature of this reward function, such a system cannot be implemented using a single step approach, since it is not practical to serve all users in each single subframe in an interference channel. As a results, rewards are defined for a complete frame (RL episode) as opposed to a single subframe:

$$\begin{aligned}\bar{\rho}_i[t] &= 0 \quad \forall t \in \{1, 2, 3, ..., T-1\} \\ \bar{\rho}_i[T] &= U.\end{aligned} \tag{9}$$

This definition of rewards is accurate but will adversely affect the convergence behavior of the system [6]. To improve convergence, the idea of reward shaping is used, [6]. In this method, the episode reward is broken down between subframes, such that their summation yields the original utility function. The following approximation is used here to implement reward shaping [13]:

$$\ln(n) \approx H_n = \sum_{k=1}^{n} \frac{1}{k}, \tag{10}$$

where it can be shown that $\frac{1}{2n+1} \leq H_n - \ln(n) - \gamma \leq \frac{1}{2n}$, where $n \in \mathbb{N}$ and γ is the Euler-Mascheroni constant. The bounds are quite tight for large values of n.

In order to use this approximation for $D_i[t]$, which is neither integer nor large, we first rewrite (6) as:

$$\begin{aligned}U = \sum_{i=1}^{M} \ln(D_i[T]) &= \sum_{i=1}^{M} \Big(\ln(\Delta.D_i[T]) - \ln(\Delta) \Big) \\ &\approx \Big(\sum_{i=1}^{M} \ln(\lfloor \Delta D_i[T] \rfloor) \Big) - M \ln(\Delta),\end{aligned} \tag{11}$$

where Δ is a tuning parameter that determines the precision of the approximation, and $\lfloor x \rfloor$ is the floor of x. This approximation is valid for large values of Δ. Equation (10) can now be used to further simplify (11):

$$U \approx \Big(\sum_{i=1}^{M} \sum_{j=1}^{\lfloor \Delta.D_i[T] \rfloor} \frac{1}{j} \Big) - M \ln(\Delta), \tag{12}$$

which can be expanded as:

$$\begin{aligned}U = &\sum_{i=1}^{M} \sum_{t=1}^{T} \Big(\sum_{j=\lfloor \Delta.D_i[t-1] \rfloor}^{\lfloor \Delta.D_i[t] \rfloor} \frac{1}{j} - \frac{1}{T} \ln(\Delta) \Big) \\ = &\sum_{t=1}^{T} \rho[t],\end{aligned} \tag{13}$$

where

$$\rho[t] = \sum_{i=1}^{M} \sum_{j=\lfloor \Delta.D_i[t-1] \rfloor}^{\lfloor \Delta.D_i[t] \rfloor} \frac{1}{j} - \frac{M}{T} \ln(\Delta). \tag{14}$$

$\rho[t]$ is the *short term reward* in each *subframe*, derived from the *reward function* of *the whole frame*.

2) $\alpha = 0$: Similar to the previous case, distributing the rewards between subframes as opposed to waiting until the end of a frame to deliver the rewards to the agents improves convergence rate. Reward shaping is simple in this case as with $\alpha = 0$, the α-fair utility reduces to the sum rate of all links in all subframes. The per subframe reward value can be evaluated as:

$$\rho[t] = \sum_{i=1}^{M} r_i[t]. \tag{15}$$

It is easy to verify that the sum of $\rho[t]$ over all subframes is equal to 0-fair utility function.

C. Exploration - Stepwise training and exploration procedure

The success of a reinforcement learning system is heavily dependent upon the effectiveness of the employed exploration strategy. The main challenge in the link level throughput maximization problem is that the reward function of each agent depends not only on its own actions, but also on the actions of its neighbors. The reward-action mapping of this problem is very difficult to explore, and classical exploration techniques fail to converge to optimal policies.

To explore the problem space efficiently, we have applied the idea presented in [9], where the authors observe that agents usually fail to learn a complex task when they start from random initial conditions. They propose to simplify the task and let the agents learn it, then make it gradually more complex. Motivated by this approach, we simplify the task of our agents by omitting MCS and defining rate function as:

$$r_i[t] = \log_2(1 + \gamma_i[t]), \tag{16}$$

Training of the agents starts with this reward function. When the agents learn this task, we switch the reward function from eqn. (16) to eqn. (17). The first term in equation (17) is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LNET.2020.3000334, IEEE Networking Letters

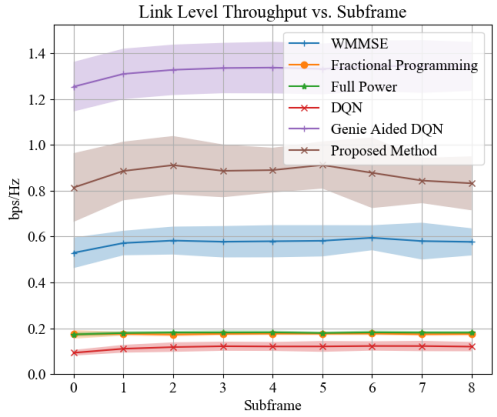


Fig. 2: d_{max} = 4.0 Km

capacity (as in (16)), and the second term ($R(m_i, \gamma_i)$) is the achievable data rate in SINR γ_i with MCS scheme m_i.

$$r_i[t] = (1-\lambda)\log_2(1+\gamma_i[t]) + \lambda R(m_i[t], \gamma_i[t]). \qquad (17)$$

where λ is gradually increased from zero to one during the training procedure, changing the reward function slowly from capacity to throughput. Such gradual increase in the complexity of the environment enables the RL agents to learn the new environment better, as they have already trained for an environment that is just a little simpler.

IV. Simulation Results

A cellular network with 25 eNBs and 100 users (4 users per cell) is simulated using Tensorflow 1.9.0, Python 3.6.7 and NVIDIA GK107 GPU. All scenarios are simulated with a frame size of 9 subframes, acceptable powers in range of 0-6 Watts and 15 MCS options obtained from LTE standards. The proposed method is compared to the following baseline algorithms:

- **Proposed method**: A two dimensional DDPG system that assigns power and MCS.
- **WMMSE**: Weighted minimum mean squared error method.
- **FP**: Fractional programming method.
- **Full-Power**: All agents transmitting with full power.
- **DQN**: A DQN that assigns power and MCS, trained using the conventional exploration method.
- **Genie-aided DQN**: A DQN that just assings power, to which the ideal MCS is provided by a genie.

To calculate the link level throughput of the benchmark algorithms, WMMSE/FP/Full-power/DQN are first utilized to allocate power levels, and then the MCS leading to the highest link level throughput is assigned to each agent. Although such MCS selection is in favor of the benchmark algorithms, the results show the superiority of the proposed method. The RL methods are all trained first, and their weights are fixed after convergence. They are tested with these fixed weights.

Fig. 2 presents the link-level throughput (per subframe) that users can achieve *on average*, using each method. The results are obtained assuming $\alpha = 0$, $d_{min} = 0.5Km$ and $d_{max} = 4Km$. Since rates depend on users' positions, we have simulated 20 different randomly initiated cellular networks and averaged users' achievable data rates. Fig. 2 depicts the averages of these rates with one standard deviation of the results highlighted on each curve. It can be seen that the proposed method outperforms all benchmarks except the genie-aided DQN upper bound. It can be also verified that DQN with conventional exploration strategy cannot learn the state-action-reward mapping which leads to a inferior performance.

The following figures are provided to investigate the effect of reward function parameters on the behavior of the agents. Figure 3 presents the histograms of the rates of different links with different reward function parameters. Simply put, Fig. 3 shows the rates that each percentile of users achieve in each case. Additionally, in each case, the average (over all users) of the achievable throughput is presented in Fig. 4. Like Fig. 2, the mean and one standard deviation of the results (obtained from multiple random user drops) are plotted for each case.

When $\alpha = 0$ the utility function is simply sum rate. In this case, as can be seen in Fig. 3a, the majority of agents do not transmit data at all, and about 25% of the agents achieve non-zero data rates. More specifically, only the UEs closest to the eNBs are chosen to be active and no service is provided to the rest. Since eNBs can transmit with higher rates to closer UEs, this strategy results in a high link level throughput for the whole network, (Fig. 4, the $\alpha = 0$ case). The issue, however, is that some UEs always receive data in all subframes while others are always inactive, hence fairness is not achieved among users.

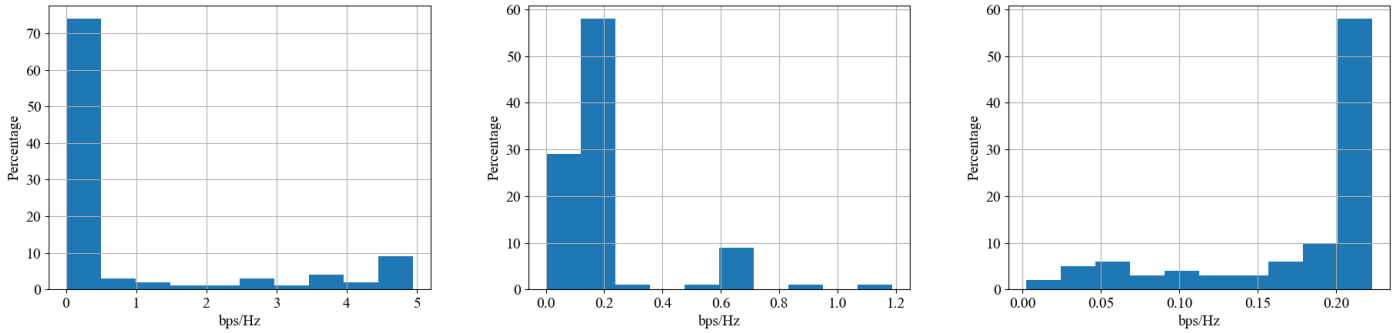
When $\alpha = 1$ and $\Delta = 4$, the objective function is an approximation of the sum of log rates (which guarantees fairness). As can be seen in Fig. 3b, the strategy that the RL agents use in this case leads to a significant drop in the number of users that do not receive transmissions at any point during a frame (from 75% to 30%). By increasing Δ to 20, the users that do not receive any transmissions drops down to only about 5% (Fig. 3c). It is safe to conclude that the increase in Δ improves fairness without any effect on average data rate (Fig. 3). This stems from the fact that a larger Δ results in a better approximation of the network utility function, subsection III-B.

To analyze the sensitivity of the proposed method to cell size, throughputs of networks with different d_{max} are compared in Fig. 5. When the difference between d_{max} and d_{min} is high enough, cell size doesn't affect the performance of the proposed method considerably. However, when d_{max} and d_{min} are close, e.g. $d_{max} = 1Km$ and $d_{min} = 0.5Km$, the cell turns into a ring (all users have roughly the same UE-eNB distance) and the agents cannot learn to well differentiate among users, leading to inferior performance.

V. Conclusion

In this paper, a multi-agent RL framework is presented for a cellular network where: a) link level throughput is maximized instead of capacity. b) Power allocation and MCS selection are performed simultaneously using DDPG agents. c) The information of each agent is limited to its neighboring cells. Due to the complexity of this problem, RL methods with conventional exploration strategies cannot outperform random resource allocation. This problem is addressed by utilizing the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LNET.2020.3000334, IEEE Networking Letters



(a) $\alpha = 0$ (b) $\alpha = 1$ and $\Delta = 4$ (c) $\alpha = 1$ and $\Delta = 20$

Fig. 3: Histogram of data rates on all links

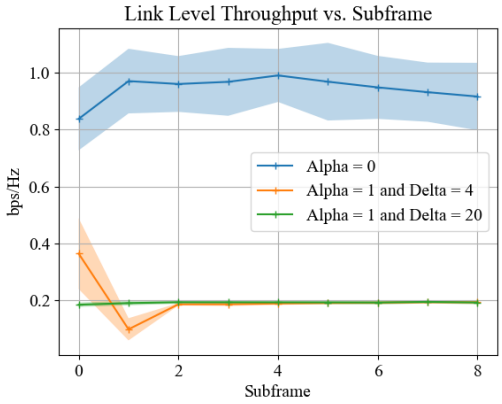


Fig. 4: The effect of reward function on throughput

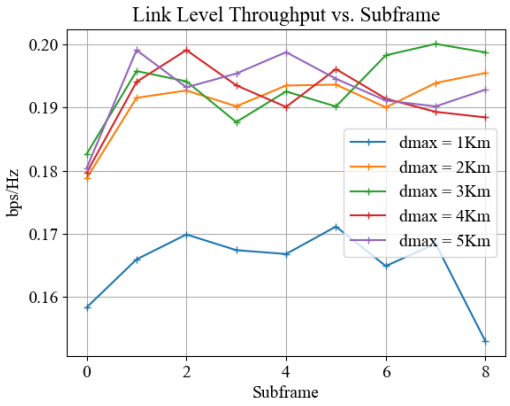


Fig. 5: The effect of d_{max} on throughput with $d_{min} = 0.5Km$

following techniques: a) Deriving short term reward functions from the long term utility function, and b) Designing a step by step training procedure. The performance of the proposed RL framework is then evaluated and its superiority to conventional methods is shown empirically.

REFERENCES

[1] O. Naparstek and K. Cohen, "Deep multi-user reinforcement learning for distributed dynamic spectrum access," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 310–323, 2018.

[2] O. Naparstek and K. Cohen, "Deep multi-user reinforcement learning for dynamic spectrum access in multichannel wireless networks," in *2017 IEEE Global Communications Conference*, pp. 1–7, IEEE, 2017.

[3] F. Meng, P. Chen, L. Wu, and J. Cheng, "Power allocation in multi-user cellular networks: Deep reinforcement learning approaches," *arXiv preprint arXiv:1901.07159*, 2019.

[4] K. Shen and W. Yu, "Fractional programming for communication systemspart i: Power control and beamforming," *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2616–2630, 2018.

[5] Y. S. Nasir and D. Guo, "Deep reinforcement learning for distributed dynamic power allocation in wireless networks," *arXiv preprint arXiv:1808.00490*, 2018.

[6] O. Naparstek and K. Cohen, "Deep multi-user reinforcement learning for distributed dynamic spectrum access," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 310–323, 2018.

[7] C. Zhong, M. C. Gursoy, and S. Velipasalar, "A deep reinforcement learning-based framework for content caching," in *2018 52nd Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–6, IEEE, 2018.

[8] K. I. Ahmed and E. Hossain, "A deep q-learning method for downlink power allocation in multi-cell networks," *arXiv preprint arXiv:1904.13032*, 2019.

[9] M. Tesch, J. Schneider, and H. Choset, "Expensive function optimization with stochastic binary outcomes," in *International Conference on Machine Learning*, pp. 1283–1291, 2013.

[10] A. S. Pagès, "Link level performance evaluation and link abstraction for lte/lte-advanced downlink," *Universitat Politècnica de Catalunya, Barcelona*, 2015.

[11] J. Fan, Q. Yin, G. Y. Li, B. Peng, and X. Zhu, "Mcs selection for throughput improvement in downlink lte systems," in *2011 Proceedings of 20th international conference on computer communications and networks (ICCCN)*, pp. 1–5, IEEE, 2011.

[12] R. Srikant and L. Ying, *Communication networks: an optimization, control, and stochastic networks perspective*. Cambridge University Press, 2013.

[13] J. Havil, "Gamma: exploring euler's constant," *The Australian Mathematical Society*, p. 250, 2003.