Action Elimination-assisted Deep Reinforcement Learning for B5G Cell Selection and Network Slicing

Sunwoo Kim, Seungnyun Kim, Kyungjoo Suh, and Byonghyo Shim
INMC, Department of Electrical and Computer Engineering, Seoul National University, Korea
Email: {swkim, snkim, kjsuh}@islab.snu.ac.kr, bshim@snu.ac.kr

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10001669

***Abstract*—With the emergence of 5G era, network slicing has received much attention due to its ability to support various services. Network slicing is an approach to partition a single physical network into multiple slices supporting separate services and has been extended to the handover scenario (where the UE moves from one cell to another) recently. In this paper, we propose a deep reinforcement learning (DRL)-based handover-aware network slicing technique for the cell selection and network slicing. Key ingredient of the proposed technique is to use action elimination to reduce the size of slice allocation decision space. In our work, we first determine the target cell providing the maximum user-requested services to the handover UE, and then assign network slices to the handover UE by exploiting action elimination-assisted DRL. From the numerical results, we demonstrate that the proposed technique outperforms the conventional network slicing techniques in terms of throughput.**

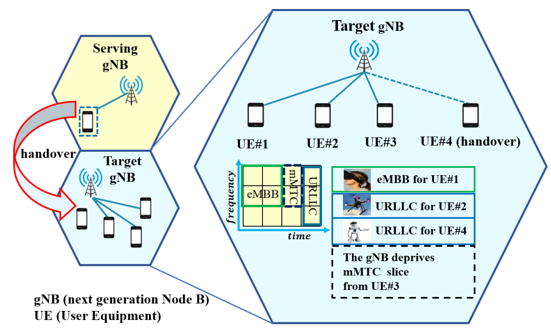

Fig. 1: Illustration of network slicing operation in 5G network handover scenario.

I. Introduction

Fifth generation (5G) wireless communication needs to support a wide variety of services such as enhanced mobile broadband (eMBB), ultra reliable and low latency communications (URLLC), and massive machine-type communications (mMTC) [1]. To accommodate various emerging services, network slicing, an approach to partition a physical network into multiple (logically) isolated networks, has received great deal of attention recently [2]. Since the network slicing operations are performed in the common physical infrastructure, the operation and maintenance costs can be saved significantly [2], [3]. Recently, network slicing has been extended from the static scenario to the handover scenario, where the UE moves from one cell (i.e., gNB, data center [4]) to another (see Fig. 1). In fact, handover occurs far more frequently due to smaller cell coverage of 5G compared to that of LTE, caused by the use of high frequency band (i.e., 3.5 GHz in FR1 and 28 GHz in FR2)[1].

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2022-2017-0-01637) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation)

[1]The coverages of pico cell and femto cell are around 200 m and 30 m, respectively whereas the coverages of macro cell and micro cell are around 20 km and 2 km, respectively.

Over the years, various handover-aware network slicing techniques have been suggested to maximize the system throughput while satisfying the QoS requirements [5]–[9]. In [5], a regression tree-based approach that sequentially decides slice ratio to maximize the system throughput has been proposed. In [6], a proportional fair-based network slicing technique has been proposed to pursue a balance between the throughput and the quality of service (QoS) of user. For the efficient network slicing, approaches exploiting the deep reinforcement learning (DRL) [10] have been proposed in [7], [8]. In [9], a multi-agent DRL-based network slicing technique, where each user equipment (UE) equipped with its own DRL agent takes care of the association between the UE and the local cells to minimize the long-term handover cost, has been proposed.

The primary purpose of this paper is to propose a handover-aware network slicing technique, called action elimination-assisted deep reinforcement learning for the cell selection and network slicing (AEDRL-CSNS). Operations of AEDRL-CSNS are divided into two main steps: 1) selection of a target cell for the seamless handover and 2) network slicing to allocate slices to the newly joined UE. In the cell selection process, we first select the set of candidate cells based on the received signal strength indicator (RSSI) of UE and then choose the cell providing the maximum throughput to the newly joined UE.

Our work is distinct from the conventional DRL-based network slicing techniques in that we employ the action elimination-assisted DRL training process to ignore network slicing decisions violating the QoS requirements of UEs among all possible decisions in the training phase. While the conventional DRL-based schemes explore infeasible actions due to the large action space, AEDRL-CSNS ensures (with high probability) that the exploration of the agent is directed toward the decisions satisfying the QoS requirements. Exploiting the action elimination technique to assist the training process of DRL, we can improve the chance of experiencing the optimal or near-optimal network slicing decision.

From the numerical evaluations on the realistic 5G handover scenario, we demonstrate that the proposed AEDRL-CSNS outperforms the conventional network slicing schemes by a large margin in terms of throughput. For example, AEDRL-CSNS achieves 28% throughput improvement over the proportional fair-based technique. Even when compared with the conventional vanilla DQN-based scheme, AEDRL-CSNS achieves more than 13% throughput improvement.

II. System Model

We consider a downlink transmission system consisting of $\mathcal{M} = \{1, \cdots, M\}$ single antenna cells where each cell serves the single antenna UEs in its coverage. We denote the set of UEs in time slot t as $\mathcal{K}^{(t)}$ where $K^{(t)} = |\mathcal{K}^{(t)}|$. We assume the handover scenario where in each time slot t, a $K^{(t)}$-th UE is newly arrived in this system so that $\mathcal{K}^{(t)} = \mathcal{K}^{(t-1)} \cup \{K^{(t)}\}$. Then it performs the handover from the current serving cell to the target cell. After the handover operation, the target cell assigns network slices to the handover UE. Specifically, the physical network of each cell is divided into N slices where each RB can be assigned to an eMBB, URLLC, or mMTC network slice.

In this setting, the data rate $R^{(t)}_{m,k,n}$ of the k-th UE in the n-th slice of m-th cell at time slot t is

$$R^{(t)}_{m,k,n} = \log_2\left(1 + \frac{|h^{(t),*}_{m,k,n} w^{(t)}_{m,k,n}|^2}{\sigma_k^2}\right), \tag{1}$$

where $h^{(t)}_{m,k,n}$ is the channel coefficient from the m-th cell to the k-th UE in the n-th network slice at time slot t and $w^{(t)}_{m,k,n}$ is the downlink precoding coefficient and σ_k^2 is the noise power.

III. Proposed AEDRL-CSNS Technique

The primary goal of the proposed AEDRL-CSNS technique is to find out the optimal network slicing policy that maximizes the long-term system throughput in the handover scenario. A major issue in this task is that the network slicing problem is a combinatorial optimization problem so that we need to try every combination of the target cell and network slicing to find out the optimal solution. To address this issue, AEDRL-CSNS uses two-stage processing: 1) the serving cell initially chooses the target cell providing the maximum user-requested services to the handover UE, and 2) the target cell newly assigns network slices to the handover UE using the action elimination-assisted DRL.

A. Target Cell Selection

The target cell selection process is divided into three main steps: 1) candidate cells selection and 2) expected throughput calculation, and 3) target cell selection.

1) Candidate Cells Selection: In this step, the serving cell chooses the candidate cells among the neighboring cells based on the measurement report of UE. Initially, when the signal strength from the serving cell is below the pre-defined threshold, the UE sends measurement reports[2] and requests handover to the serving cell. Then the serving cell picks the set of candidate cells $\mathcal{M}_{\text{cand}}$ whose RSSI values are above the pre-defined threshold τ as $\mathcal{M}_{\text{cand}} = \{m \in \mathcal{M} \mid \text{RSSI}(m) \geq \tau\}$. For example, if the RSSI reports of 5 neighboring cells is $[-80\,\text{dBm}, -57\,\text{dBm}, -62\,\text{dBm}, -70\,\text{dBm}, -50\,\text{dBm}]$ and the RSSI threshold value is set to $-65\,\text{dBm}$, then the candidate cell set is $\mathcal{M}_{\text{cand}} = \{2, 3, 5\}$.

2) Expected Throughput Calculation: Once $\mathcal{M}_{\text{cand}}$ (set of candidate cells) is chosen, the candidate cells feed the number of available network slices corresponding to the requested service type of UE back to the serving cell. Using the network slice information fed back from the candidate cells and the measurement reports of UE, the serving cell calculates the expected throughput $T^{(t)}_{m,\text{ho}}$ of $K^{(t)}$-th handover UE for each candidate cells as

$$T^{(t)}_{m,\text{ho}} = \sum_{n=1}^{N} \upsilon^{(t)}_n f_{m,n} R^{(t)}_{m,K^{(t)},n}, \quad m \in \mathcal{M}, \tag{2}$$

where $f_{m,n}$ is the frequency bandwidth allocated to the n-th network slice of m-th cell and $\upsilon^{(t)}_n \in \{0, 1\}$ indicates whether the n-th network slice is available or not.

3) Target cell Selection: In the final step, the serving cell selects the target cell $m^{(t)}_{\text{ho}}$ providing the maximum expected throughput to the $K^{(t)}$-th handover UE as

$$m^{(t)}_{\text{ho}} = \arg\max_{m \in \mathcal{M}^{\text{cand}}} T^{(t)}_{m,\text{ho}}. \tag{3}$$

Once the target cell $m^{(t)}_{\text{ho}}$ is chosen, then the serving cell notifies the chosen target cell to prepare for the handover operation. The target cell then newly assigns network slices to the $K^{(t)}$-th handover UE.

B. DRL-based Network Slice Allocation

1) Problem Formulation: After the target cell selection, we allocate network slices to the newly joined $K^{(t)}$-th UE and conventional UEs $\mathcal{K}^{(t-1)}$ in a way to maximize the long-term system throughput under the QoS constraints. Note that in this subsection, we omit the target cell index m^* for notational simplicity. To indicate the allocation of network slices to the

[2]Measurement reports include RSSI that represents the received signal strength and network slice selection assistance information (NSSAI) that represents the type of requested service of UE.

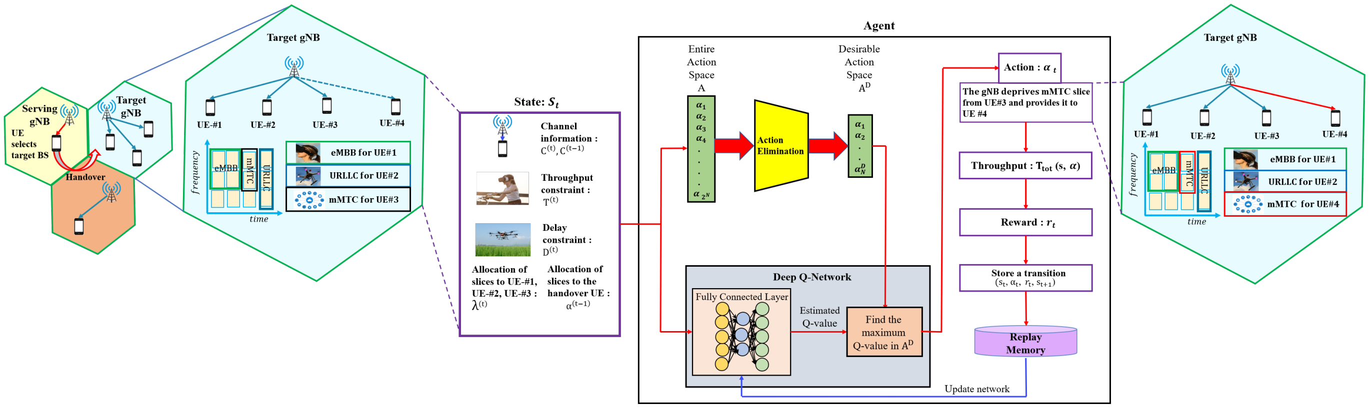

Fig. 2: The proposed DRL-based network slicing scheme.

$K^{(t)}$-th handover UE at time slot t, we use the binary vector $\boldsymbol{\alpha}^{(t)} = [\alpha_1^{(t)}, \cdots, \alpha_N^{(t)}]^T \in \mathbb{R}^N$ where

$$\alpha_n^{(t)} = \begin{cases} 1 & \text{if } n\text{-th slice is allocated to the handover UE} \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, to indicate the allocation of network slices for each conventional UE in the target cell at time slot t, say k-th UE, we use a binary vector $\boldsymbol{\lambda}_k^{(t)} = [\lambda_{k,1}^{(t)}, \cdots, \lambda_{k,N}^{(t)}]^T$ to indicate the allocation of N network slices before the handover. Then the overall system throughput $T_{\text{total}}^{(t)}$ in the handover scenario at time slot t is given by the sum of throughputs of two major components: 1) throughput $T_{\text{conv}}^{(t)}$ of conventional UEs $\mathcal{K}^{(t-1)}$ and 2) throughput $T_{\text{ho}}^{(t)}$ of the $K^{(t)}$-th handover UE:

$$T_{\text{total}}^{(t)} = T_{\text{conv}}^{(t)} + T_{\text{ho}}^{(t)}. \tag{4}$$

To be specific, once the network slices are newly assigned to the handover UE, the throughput $T_k^{(t)}$ of k-th conventional UE is updated as

$$T_k^{(t)} = \sum_{n=1}^{N} \lambda_{k,n}^{(t)} f_n R_{k,n}^{(t)} (1 - \alpha_n^{(t)}), \quad \forall k \in \mathcal{K}^{(t-1)}, \tag{5}$$

where f_n is the resource bandwidth allocated to the n-th network slice and $R_{k,n}^{(t)}$ is the data rate of k-th conventional UE in the n-th slice. The corresponding system throughput $T_{\text{conv}}^{(t)}$ of conventional UEs is $T_{\text{conv}}^{(t)} = \sum_{k \in \mathcal{K}^{(t-1)}} T_k^{(t)}$. Also, the throughput $T_{\text{ho}}^{(t)}$ of $K^{(t)}$-th handover UE is expressed as

$$T_{\text{ho}}^{(t)} = \sum_{n=1}^{N} \alpha_n^{(t)} f_n R_{K^{(t)},n}^{(t)}. \tag{6}$$

Then the total system throughput $T_{\text{total}}^{(t)}$ is expressed as

$$\begin{aligned} T_{\text{total}}^{(t)} &= T_{\text{conv}}^{(t)} + T_{\text{ho}}^{(t)} \qquad (7) \\ &= \sum_{n=1}^{N} f_n \left(\sum_{k \in \mathcal{K}^{(t-1)}} \lambda_{k,n}^{(t)} R_{k,n}^{(t)} (1 - \alpha_n^{(t)}) + \alpha_n^{(t)} R_{K^{(t)},n}^{(t)} \right) \end{aligned}$$

Other than the maximization of total system throughput, we need to consider two major constraints to fulfill the QoS requirements: 1) data throughput constraint and 2) delay constraint. First, the data throughput constraint requires that the throughput of each UE should be larger than the minimum throughput requirement. That is,

$$T_k^{(t)} \geq T_{k,\text{min}}^{(t)}, \quad \forall k \in \mathcal{K}^{(t-1)} \tag{8}$$

$$T_{\text{ho}}^{(t)} \geq T_{\text{ho,min}}^{(t)}, \tag{9}$$

where $T_{k,\text{min}}^{(t)}$ and $T_{\text{ho,min}}^{(t)}$ are the minimum throughput requirements of the k-th conventional UE and the $K^{(t)}$-th handover UE at time slot t, respectively. In addition, the delay constraint is induced from the requirement that one data packet should be completely transmitted within one frame [11]. That is, the frame duration should be less than or equal to the maximum packet delay:

$$\frac{F_k^{(t)}}{T_k^{(t)}} \leq D_{k,\text{max}}^{(t)}, \quad \forall k \in \mathcal{K}^{(t-1)} \tag{10}$$

$$\frac{F_{\text{ho}}^{(t)}}{T_{\text{ho}}^{(t)}} \leq D_{\text{ho,max}}^{(t)}, \tag{11}$$

where $F_k^{(t)}$ and $F_{\text{ho}}^{(t)}$ are the packet length and $D_{k,\text{max}}^{(t)}$ and $D_{\text{ho,max}}^{(t)}$ are the maximum packet delay of the k-th conventional UE and the $K^{(t)}$-th handover UE at time slot t, respectively.

By incorporating (8)-(11), the network slicing problem $\mathcal{P}$ can be formulated as

$$\mathcal{P}: \max_{\{\boldsymbol{\alpha}^{(t)}\}} \sum_{t=1}^{T} T_{\text{total}}^{(t)} \tag{12a}$$

$$\text{s.t. } T_k^{(t)} \geq T_{k,\text{min}}^{(t)}, \quad \forall k \in \mathcal{K}^{(t-1)} \tag{12b}$$

$$T_{\text{ho}}^{(t)} \geq T_{\text{ho,min}}^{(t)} \tag{12c}$$

$$\frac{F_k^{(t)}}{T_k^{(t)}} \leq D_{k,\text{max}}^{(t)}, \quad \forall k \in \mathcal{K}^{(t-1)} \tag{12d}$$

$$\frac{F_{\text{ho}}^{(t)}}{T_{\text{ho}}^{(t)}} \leq D_{\text{ho,max}}^{(t)}. \tag{12e}$$

By plugging (5) and (6), $\mathcal{P}$ can be re-expressed as

$$\mathcal{P}^{'}: \max_{\{\boldsymbol{\alpha}^{(t)}\}} \sum_{t=1}^{T}\sum_{n=1}^{N} f_n\Bigg(\sum_{k\in\mathcal{K}^{(t-1)}} \lambda_{k,n}^{(t)} R_{k,n}^{(t)}(1-\alpha_n^{(t)})+\alpha_n^{(t)} R_{K^{(t)},n}^{(t)}\Bigg) \tag{13a}$$

$$\text{s.t.} \sum_{n=1}^{N} \lambda_{k,n}^{(t)} f_n R_{k,n}^{(t)}(1-\alpha_n^{(t)}) \geq T_{k,\text{min}}, \forall k \in \mathcal{K}^{(t-1)} \tag{13b}$$

$$\sum_{n=1}^{N} \alpha_n^{(t)} f_n R_{K^{(t)},n}^{(t)} \geq T_{\text{ho,min}} \tag{13c}$$

$$\frac{F_k}{\sum_{n=1}^{N} \lambda_{k,n}^{(t)} f_n R_{k,n}^{(t)}(1-\alpha_n^{(t)})} \leq D_{k,\text{max}}^{(t)}, \forall k \in \mathcal{K}^{(t-1)} \tag{13d}$$

$$\frac{F_{\text{ho}}}{\sum_{n=1}^{N} \alpha_n^{(t)} f_n R_{K^{(t)},n}^{(t)}} \leq D_{\text{ho,max}}^{(t)}. \tag{13e}$$

This problem is a binary decision problem and thus very difficult to find out the optimal solution.

2) DRL-based Network Slice Allocation Model: To solve $\mathcal{P}^{'}$, we exploit the action elimination-assisted DRL, where DNN in the DRL agent (i.e., deep Q-network (DQN)) learns the effect of each slice allocation decision over the long-term system throughput. Since AEDRL-CSNS eliminates ineffective actions, we can expect that the DRL agent explore effective actions exclusively (see Fig. 2).

In the proposed DRL framework, the state of the environment at the time slot t consists of several parts:

- channel matrix $\mathbf{C}^{(t)} \in \mathbb{R}^{K^{(t)}\times N}$ whose (k,n)-th element is the channel coefficient $h_{k,n}^{(t)}$
- slice allocation information of conventional UEs $\boldsymbol{\Lambda}^{(t)} = [\boldsymbol{\lambda}_1, \cdots, \boldsymbol{\lambda}_{K^{(t-1)}}] \in \mathbb{R}^{N\times K^{(t-1)}}$
- minimum throughput requirements of UEs $\mathbf{T}_{\text{min}}^{(t)} = [T_{1,\text{min}}^{(t)}, \cdots, T_{K^{(t-1)},\text{min}}^{(t)}, T_{\text{ho,min}}^{(t)}] \in \mathbb{R}^{K^{(t)}}$
- maximum packet delays of UEs $\mathbf{D}_{\text{max}}^{(t)} = [D_{1,\text{max}}^{(t)}, \cdots, D_{K^{(t-1)},\text{max}}^{(t)}, D_{\text{ho,max}}^{(t)}] \in \mathbb{R}^{K^{(t)}}$

In summary, the state can be expressed as

$$s_t = [\mathbf{C}^{(t)}, \mathbf{C}^{(t-1)}, \boldsymbol{\Lambda}^{(t)}, \mathbf{T}_{\text{min}}^{(t)}, \mathbf{D}_{\text{max}}^{(t)}, \boldsymbol{\alpha}^{(t-1)}]. \tag{14}$$

By exploiting the extracted features among $\mathbf{C}^{(t)}$, $\mathbf{C}^{(t-1)}$, $\boldsymbol{\Lambda}^{(t)}$, $\mathbf{T}_{\text{min}}^{(t)}$, and $\mathbf{D}_{\text{max}}^{(t)}$, the agent learns the optimal network slicing policy in the handover scenario. In our work, we set the action as the binary vector indicating the allocation of network slices to the handover UE ($\alpha_t = [\alpha_1^{(t)}, \cdots, \alpha_N^{(t)}]^T \in \mathbb{R}^N$). Lastly, to maximize the overall throughput, we set the reward as the total throughput ($r_t = T_{\text{total}}^{(t)} = T_{\text{conv}}^{(t)} + T_{\text{ho}}^{(t)}$).

3) Action Elimination-assisted Training Process: For each allocation decision $\alpha \in A$, we compute the throughputs of conventional UEs $T_{\text{conv}}^{(t)}$ and the handover UE $T_{\text{ho}}^{(t)}$ (see (5) and (6)). Using $T_{\text{conv}}^{(t)}$ and $T_{\text{ho}}^{(t)}$, we identify the infeasible decisions violating the QoS constraints, and then eliminate them from A to obtain the desirable action space A^D:

$$A^D = \{\alpha \in A | (13\text{b}), (13\text{c}), (13\text{d}), (13\text{e}) \text{ are satisfied.}\}. \tag{15}$$

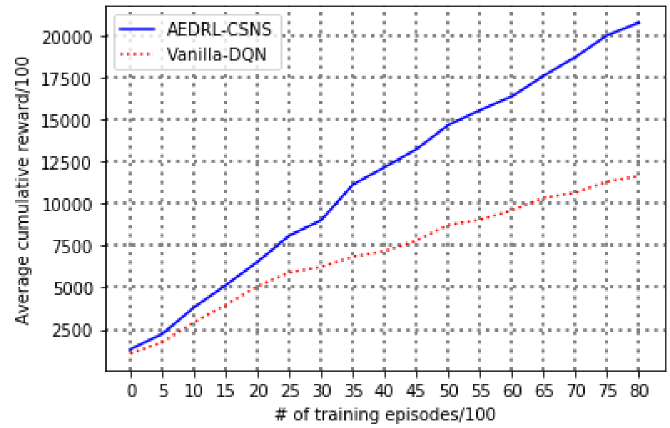

Fig. 3: Average cumulative reward as a function of the number of training episodes.

From A^D, the DRL agent estimates the Q-values of all possible actions and use them for training.

IV. Simulation Results

In this section, we evaluate the throughput performance of AEDRL-CSNS and compare it with the conventional handover-aware network slicing techniques, including round robin-based network slicing [3], regression tree-based network slicing [5], proportional fair-based network slicing [6], and vanilla DQN-based technique where only the basic DQN is used [7].

In our simulation, we consider a downlink transmission scenario where M cells simultaneously serve K UEs. The small cells are uniformly distributed in the hexagonal area of inter-site distance (ISD) 200m and the UEs move freely at a constant speed $v \in \text{Unif}[v_{min}, v_{max}]$, where v_{max} and v_{min} are the max/min speed of UE. For the fading channel model, we use the small-scale fading coefficient $g_{m,k}$ generated from the complex Gaussian distribution (i.e, $g_{m,k} \sim \mathcal{CN}(0,1)$) and the large-scale fading coefficient $\beta_{m,k}$ generated based on Hata-COST231 model expressed as in [12]. For the training of AEDRL-CSNS, we use the Adam optimizer, a robust gradient-based optimization tool [13].

In Fig. 3, we plot the cumulative reward as a function of the number of training episodes. We observe that the cumulative reward of AEDRL-CSNS is higher than that of the vanilla DQN-based handover-aware network slicing scheme. This is because while the vanilla DQN-based scheme receives penalties by exploring the infeasible slice allocation decisions violating the data throughput and delay constraints, AEDRL-CSNS receives no such penalty as it ignores such decisions via action elimination so that the chance of exploring feasible decisions is increased considerably, resulting in an improvement of cumulative reward.

In Fig. 4, we plot the throughput performance of the proposed AEDRL-CSNS technique as a function of UE's data rate requirement. We observe that AEDRL-CSNS outperforms conventional network slicing techniques by a large margin.

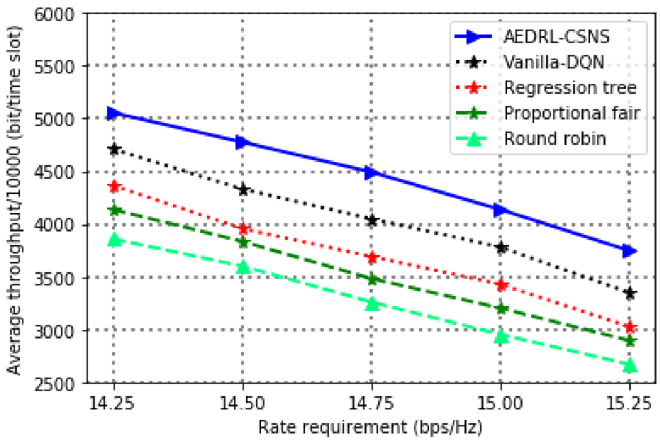

Fig. 4: Average throughput as a function of rate requirement.

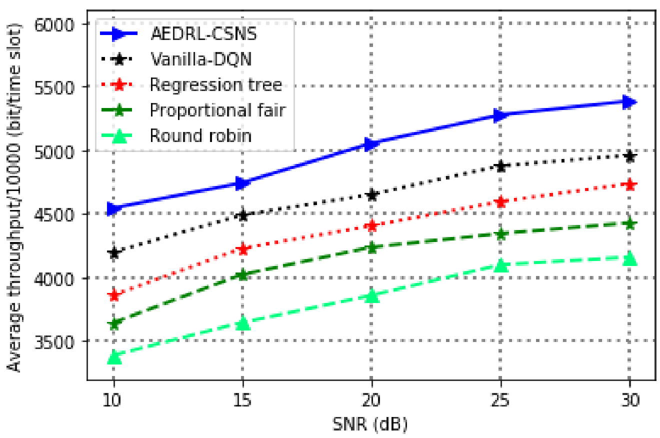

Fig. 5: Average throughput as a function of SNR.

For example, when the rate requirement is 14.75 bps/Hz, we observe that AEDRL-CSNS achieves about 13% improvement in the average throughput over the vanilla DQN-based scheme. Since the allocation decisions that cannot meet the data throughput constraint are ignored, AEDRL-CSNS has a better chance to find out the optimal slice allocation policy.

In Fig. 5, we evaluate the performance of AEDRL-CSNS as a function of signal-to-noise ratio (SNR). We observe that AEDRL-CSNS achieves a significant throughput gain over the conventional techniques in the handover scenario. For instance, AEDRL-CSNS achieves about 9% improvement in the system throughput over the regression tree-based technique at SNR$=20$ dB. We also observe that the system throughput increases with SNR. For example, when SNR$=15$ dB, AEDRL-CSNS achieves about 17% improvement over the proportional fair-based technique but it goes up to 21% when SNR$=25$ dB.

V. Conclusion

In this paper, we proposed a novel cell selection and network slicing technique called AEDRL-CSNS for 5G handover scenario. In AEDRL-CSNS, a target cell providing the maximum user-requested services is chosen first and then the network slices are allocated to the handover UE using DRL-based slicing mechanism. Key ingredient of the proposed handover-aware network slicing technique is the action elimination to remove the undesirable slice allocation decisions violating the QoS requirements. In doing so, the chance of exploring desirable decisions can be enhanced, resulting in a substantial improvement of throughput and QoS requirement satisfaction. Through the simulations on realistic 5G environment, we observed that AEDRL-CSNS brings a significant throughput gain over the conventional network slicing techniques.

References

[1] H. Ji, S. Park, J. Yeo, Y. Kim, J. Lee and B. Shim, "Ultra Reliable and Low Latency Communications in 5G Downlink: Physical Layer Aspects," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 124-130, Jun. 2018.

[2] S. Kim and B. Shim, "Network slicing with deep reinforcement learning for beyond 5G and 6G systems," in *Proc. Asia Pacific Wireless Commun. Symp. (APWCS).*, Aug. 2022.

[3] C. Bektas, S. Bocker, F. Kurtz, and C. Wietfeld, "Reliable software-defined ran network slicing for mission-critical 5G communication networks," in *Proc. IEEE Globecom Workshops (GC Wkshps).*, Dec. 2020.

[4] S. Kim, L. T. Nguyen, J. Kim, and B. Shim, "Deep learning based low-rank matrix completion for IoT network localization," *IEEE Wireless Commun. Lett.*, vol. 10, no. 10, pp. 2115-2119, Oct. 2021.

[5] N. Salhab, R. Rahim, R. Langar, and R. Boutaba, "Machine learning based resource orchestration for 5G network slices," in *Proc. IEEE Global Commun. Conf. (GLOBECOM).*, Waikoloa, HI, USA, Dec. 2019.

[6] M. Leconte, G. S. Paschos, P. Mertikopoulos, and U. Kozat, "A resource allocation framework for network slicing," in *Proc. IEEE INFOCOM.*,Honolulu, HI, USA, Apr. 2018.

[7] K. Suh, S. Kim, Y. Ahn, S. Kim, H. Ju, and B. Shim, "Deep reinforcement learning-based network slicing for beyond 5G," *IEEE Access*, vol. 10, pp. 7384-7395, Jan. 2022.

[8] Y.-J. Liu, G. Feng, J. Wang, Y. Sun, and S. Qin, "Access control for RAN slicing based on federated deep reinforcement learning," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Montreal, QC, Canada, pp. 1–6, 2021.

[9] Y. Sun, W. Jiang, G. Feng, P. V. Klaine, L. Zhang, M. A. Imran and Y. C. Liang, "Efficient handover mechanism for radio access network slicing by exploiting distributed learning," *IEEE Trans. Netw. and Serv. Manag.*, vol. 17, no. 4, pp. 2620-2633, Dec. 2020.

[10] H. Ju, S. Kim, Y. Kim, and B. Shim, "Energy-efficient ultra-dense network with deep reinforcement learning," *IEEE Trans. Wireless Commun.*, Feb. 2022.

[11] J. Tang, B. Shim, and T. Q. S. Quek, "Service multiplexing and revenue maximization in sliced C-RAN incorporated with URLLC and multicast EMBB," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 4, pp. 881–895, Apr. 2019.

[12] S. Kim, J. Son, and B. Shim, "Energy-efficient ultra-dense network using LSTM-based deep neural network," *IEEE Trans. Wireless Commun.*, vol. 20, no. 7, pp. 4702-4715, Mar. 2021.

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.