Network Slicing and Resource Allocation in an Open RAN System

***Abstract*—**

I. Introduction

A. Main Contributions

In this paper, as depicted in Figure 1, we aim at solving the problem of dynamic network slicing in the O-RAN system. Here, we examine a single-cell downlink system involving multiple network slices that share existing radio sources. The main contributions of this research are summarized as follow:

- The system has a two-time scale and should be solved in two layers. On a large-time scale, the problem of obtaining an optimal number of VNFs and the VNF placement is performed. Also, the assignment of PRB to the slices is obtained. On a small-time scale, the problem of power allocation, and the assignment of PRB to UE in each slice is applied.
- We consider two types of services that requires specific QoS: URLLC, which requires low latency and high reliability; and eMBB, which needs high data rate. The problem of RAN slicing for different types of services is studied in this research.
- We consider an intelligent resource allocation in the O-RAN architecture for the different services. Since conventional models do not perform well here because of the heterogeneous QoS requirements of each service type and the complexity of dynamic RAN slicing. So we must switch to machine learning and dynamic methods and find the most suitable approach. A dynamic strategy of resource allocation is required to solve this problem to achieve the specific QoS for each type of service in the O-RAN architecture. In the small-time scale, the actor-critic algorithm such as DDPG is applied to the system that is based on control policy search. This method directly searches the optimal control policy by estimating of the gradient with respect to the parameters of the control policy. In the large-time scale, the deep Q-Network is implemented which is a value based algorithm and it can find best solution for discrete action-state.

II. Current State of the Research

A. System Model

Assume we have S preallocated slices serving S services contains eMBB, and URLLC services; We consider S_1 slices for the eMBB service type and S_2 slices for the URLLC service type. Therefore, we have $S = S_1 + S_2$. Each Service $s \in \{1, 2, ..., S\}$ consists of U_s single-antenna user equipments (UEs) which require certain QoS to be able to use the requested program. There are different application requests which fall into one of these service categories. Each application request requires specific QoS. Assume each slice $s \in \{1, 2, ..., S\}$, consists of K_s, pre-allocated physical resource blocks (PRBs) obtained in the large-time scale, M_s^d VNFs for the processing of O-DU, M_s^c VNFs for the processing of O-CU-UP and M_s^u VNFs for the processing of UPF. Virtual network functions (VNFs) are functional blocks of the system. Each VNF instance is running on a virtual machine (VM) using resources from the data centers. Each VM, requires enough resources of CPU, memory, storage and network bandwidth. Moreover, we assume the system has R single-antenna O-RU that serverd UEs cooperatively. Each O-RU $r \in \{1, 2, ..., R\}$ is transmitting and receiving the data of all UEs using the coordinated multipoint transmission (COMP) technology. Moreover, all O-RUs, have access to all PRBs.

B. The Achievable Rate

The SNR of i^{th} UE requesting served at slice s is obtained from

$$\rho_{u(s,i)} = \frac{\sum_{k=1}^{K_s} e_{u(s,i)}^k p_{u(s,i)}^k |\mathbf{h}_{R,u(s,i)}^{H\,k}|^2}{BN_0}, \tag{1}$$

where $p_{u(s,i)}^k$ represents the transmission power allocated by O-RUs to i^{th} UE served at slice s on PRB k. $\mathbf{h}_{R,u(s,i)}^k \in \mathbb{C}^R$ is the vector of channel gain of a wireless link from O-RUs to the i^{th} UE in s^{th} slice. Moreover, $e_{u(s,i)}^k \in \{0, 1\}$ is a binary variable that illustrates whether PRB k is assigned to the i^{th} UE allocated to s^{th} slice or not. Also, BN_0 denotes the power of Gaussian additive noise. Moreover, we assume that each PRB is assigned to no more than one UEs. So we have

$$\sum_{s=1}^{S} \sum_{u=1}^{U_s} e_{u(s,i)}^k = 1 \tag{2}$$

The achievable data rate for the i^{th} UE request in the s_1^{th} application of service type 1 (eMBB) can be written as $\mathcal{R}_{u(s_1,i)}^e$.

$$\mathcal{R}_{u(s_1,i)} = B \log_2(1 + \rho_{r,u(s_1,i)}), \tag{3}$$

where B is the bandwidth of system. $\mathcal{R}_{u(s_1,i)}^{e,r}$ is the achievable rate of each RU r to UE i in slice s_1. Since the blocklength in URLLC and mMTC is finite, the achievable data rate for the i^{th} UE request in the s_2^{th}, application of service type 2 (URLLC) is not achieved from Shannon

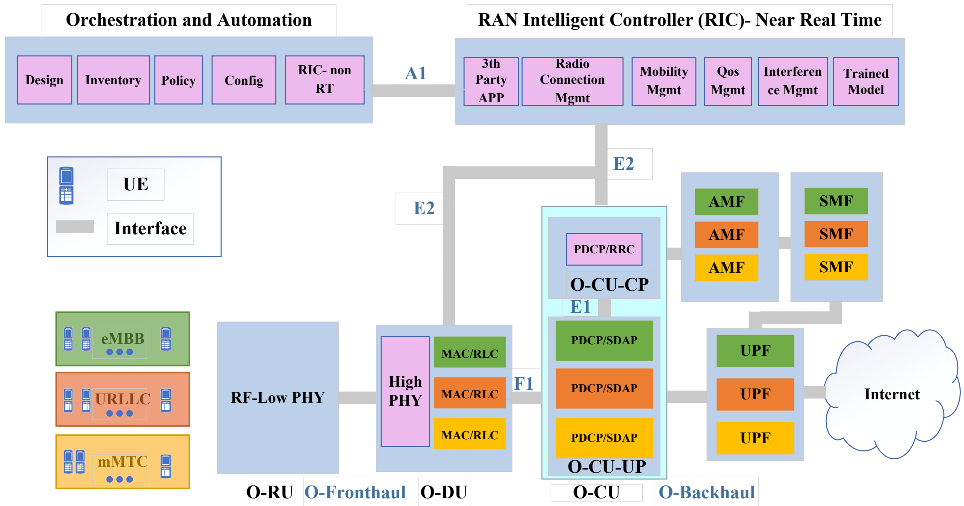

Fig. 1: Network sliced ORAN system

Capacity formula. So, for the short packet transmission the achievable data rate is approximated as follow

$$\mathcal{R}^{\mathrm{u}}_{u(s_2,i)} = B(\log_2(1+\rho_{u(s_2,i)}) - \zeta_{u(s_2,i)}), \qquad (4)$$

where $\zeta_{u(s_2,i)} = \log_2(e)Q^{-1}(\epsilon)\sqrt{\frac{C_{u(s_2,i)}}{N_{u(s_2,i)}}})$ where ϵ is the transmission error probability, Q^{-1} is the inverse of Q function (i.e., Gaussian), $C_{u(s_2,i)} = 1 - \frac{1}{(1+\rho_{u(s_2,i)})^2}$ depicts the channel dispersion of UE i at slice s_2, experiencing PRB k and $N_{u(s_2,i)}$ represents the blocklength of it. $\mathcal{R}^{e,r}_{u(s_1,i)}$ is the achievable rate of each RU r to UE i in slice s_2.

C. Mean Delay

In this part, the mean processing delay for each service is obtained. Suppose the mean processing delay is depicted as T_{proc},

$$T^{\text{proc}} = T^{RU} + T^{DU} + T^{CU}, \qquad (5)$$

Assume the packet arrival of UEs follows a Poisson process with arrival rate $\lambda_{u(s,i)}$ for the i^{th} UE of the s^{th} service (or slice). Therefore, the mean arrival data rate of the s^{th} slice in the UPF layer is $\alpha_s^U = \sum_{u=1}^{U_s} \lambda_{u(s,i)}$. Assume the mean arrival data rate for each slice s (α_s) is approximately equal to the mean arrival data rate of the O-CU-UP layer (α_s^C) and the O-DU (α_s^D). so $\alpha_s \approx \alpha_s^C \approx \alpha_s^D$, Because the amount of data traffic transferred along the route (regardless of frame changes) is constant. Since, by using Burke's theorem, the mean arrival data rate of the second and third layers, which are processed in the first layer, is still poisson with rate α_s. It is assumed that there are load balancers in each layer for each service to divide the incoming traffic to VNFs equally. Suppose the baseband processing of each VNF is depicted as M/M/1 processing queue. Each packet is processed by one of the VNFs of a slice. So, the mean delay for the s^{th} slice in the O-DU, the O-CU, and the UPF is modeled as M/M/1 queue, is formulated as follows, respectively [17]–[19],

$$\begin{aligned} T_s^{DU} &= \frac{1}{\mu_s^d - \alpha_s/M_s^d}, \\ T_s^{CU} &= \frac{1}{\mu_s^c - \alpha_s/M_s^c}, \end{aligned} \qquad (6)$$

where M_s^d, M_s^c and M_s^u are the variables that depict the number of VNFs in O-DU, O-CU-UP and UPF, respectively. Moreover, $1/\mu_s^d$, and $1/\mu_s^c$ are the mean service time of the O-DU, and O-CU layers, respectively. Besides, α_s is the arrival rate which is divided by load balancer before arriving to the VNFs. The arrival rate of each VNF in each layer for each slice s is α_s/M_s^i $i \in \{d, c\}$.

$T^{RU}_{u(s,i)}$ is the mean transmission delay of the i^{th} UE of the s^{th} service on the wireless link. The arrival data rate of wireless link for each UE i of service s is $\lambda_{u(s,i)}$ As a result, we have $\sum_{i=1}^{U_s} \lambda_{u(s,i)} = \alpha_s$. Moreover, The service time of transmission queue for UE i requesting service s has an exponential distribution with mean $1/R_{u(s,i)}$ and can be modeled as a M/M/1 queue [17]–[19].

Therefore, the mean delay of the transmission layer for UE i in slice s is

$$T^{RU}_{u(s,i)} = \frac{1}{R_{u(s,i)} - \lambda_{u(s,i)}}. \qquad (7)$$

D. Reliability of URLLC

As we know, UEs request URLLC services, require services with low latency. For the M/M/1 system, the probability of the delay for each application s_2 in the O-RU is as follow,

$$P_r\{T^s_{RU} \geq T^{max}_{RU}\} = e^{-(R_{tot_s} - \alpha_s)T^{max}_{RU}} \qquad (8)$$

Also, we do not consider the reliability for O-CU and O-DU.

E. Physical Data Center Resource

Each VNF requires physical resources that include memory, and CPU. Let the required resources for VNF f in slice s is represented by a tuple as

$$\bar{\Omega}_s^f = \{\Omega_{M,s}^f, \Omega_{C,s}^f\}, \tag{9}$$

where $\bar{\Omega}_s^f \in \mathbb{C}^2$ and $\Omega_{M,s}^f, \Omega_{C,s}^f$ indicate the amount of required memory, and CPU, respectively. Moreover, the total amount of required memory, storage, CPU and Network Bandwidth of all VNFs of a slice in O-DU, and O-CU is defined respectively as follows

$$\begin{aligned} \bar{\Omega}_{\mathfrak{z},s}^{tot,d} &= \sum_{f=1}^{M_s^d} \bar{\Omega}_{\mathfrak{z},s}^{f,d}, \\ \bar{\Omega}_{\mathfrak{z},s}^{tot,c} &= \sum_{f=1}^{M_s^c} \bar{\Omega}_{\mathfrak{z},s}^{f,c}, \end{aligned} \tag{10}$$

$\forall \mathfrak{z} \in \{M, C\}$, where, $\bar{\Omega}_{\mathfrak{z},s}^{f,d}$, and $\bar{\Omega}_{\mathfrak{z},s}^{f,c}$ are the amount of resource that a VNF required in O-DU, and O-CU, respectively. Then,

$$\bar{\Omega}_{\mathfrak{z},s}^{tot} = \bar{\Omega}_{\mathfrak{z},s}^{tot,d} + \bar{\Omega}_{\mathfrak{z},s}^{tot,c} \tag{11}$$

Also, there are D_c data centers (DC), serving the VNFs. Each DC contains several servers that supply VNF requirements. The amount of memory, and CPU is denoted by τ_{M_j}, and τ_{C_j} for the j^{th} DC, respectively

$$\tau_j = \{\tau_{M_j}, \tau_{C_j}\},$$

In this system model, the assignment of physical DC resources to VNFs is considered. Let $y_{s,d}$ be a binary variable indicating whether the d^{th} DC is allocated the resources to the VNFs of s^{th} slice or not.

F. Problem Statement

Assume the power consumption of baseband processing at each DC d that is connected to VNFs of a slice s is depicted as ϕ_s. So the total power of the system for all active DCs that are connected to slices can be represented as

$$\phi_{tot} = \textstyle\sum_{s=1}^{S} \phi_s + \sum_{d=1}^{D_c} z_d \psi_d.$$

where, z_d is shown that whether the d^{th} DC is active or not and ψ_d is a static cost when a DC is active, i.e.,

$$z_d = \begin{cases} 1 & \sum_{s=1}^{S} y_{s,d} \geq 1 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Here, we assume that if any VNF placed in a server d is used, the server is on and active, otherwise, it is off. In addition, ϕ_s is obtained from below

$$\phi_s = M_s^u \phi_s^u + M_s^c \phi_s^c + M_s^d \phi_s^d \tag{13}$$

where, ϕ_s^u, ϕ_s^c and ϕ_s^d are the static cost of energy in UPF, CU and DU, respectively. Here, we want to maximize the energy efficiency η. So the optimization problem is formulated as follow

$$\max_{\boldsymbol{M},\boldsymbol{Y},\boldsymbol{E},\boldsymbol{P},\boldsymbol{G}} \eta = \frac{\sum_{s=1}^{S_1}\sum_{i=1}^{U_s} R_{u_{(s,k)}}}{\phi_{tot} + P_r} \tag{14a}$$

$$\begin{aligned}
\text{subject to} \quad & \mathcal{R}_{u_{(s,i)}} \geq \mathcal{R}_{min}^s \quad \forall s, && \text{(14b)}\\
& \mathcal{R}_{u_{(s,i)}} \leq \mathcal{R}_{max}^s \quad \forall s, && \text{(14c)}\\
& \textstyle\sum_{s=1}^{S} y_{s,d} \bar{\Omega}_{\mathfrak{z},s}^{tot} \leq \tau_{\mathfrak{z}_d}, \quad \mathcal{E} = \{M, C\}, && \text{(14d)}\\
& p_{u(s,i)}^k \geq 0 \quad \forall i, \forall r, \forall s, \forall k, && \text{(14e)}\\
& p_{u(s,i)}^k \leq P_s^{max} \quad \forall i, \forall r, \forall s, \forall k, && \text{(14f)}\\
& T_{u(s,i)}^{\text{proc}} \leq T_s^{max} \quad \forall i, \forall s, && \text{(14g)}\\
& \mu_s \geq \alpha_s / M_s \quad \forall s, && \text{(14h)}\\
& \mathcal{R}_{u_{(s,i)}} \geq \lambda_{u_{(s,i)}} \quad \forall i, \forall s, && \text{(14i)}\\
& 0 \leq M_s \leq M_s^{max} \quad \forall s, && \text{(14j)}\\
& \sum_{s=1}^{S}\sum_{i=1}^{U_s} e_{u(s,i)}^k = 1 \quad \forall s, \forall i, \forall r && \text{(14k)}\\
& \phi^{\text{tot}} \leq \phi^{max}, && \text{(14l)}\\
& e_{r,u(s,i)}^k \in \{0, 1\} \quad \forall s, \forall i, && \text{(14m)}\\
& P_r\{T_{RU}^{s_2} \geq T_{RU}^{max}\} = e^{-(R_{tot_{s_2}} - \alpha_{s_2}) T_{RU}^{max}} && \text{(14n)}
\end{aligned}$$

where $\boldsymbol{P} = [p_{u(s,i)}^k]$, $\forall s, \forall i, \forall k$, is the matrix of power for UEs, $\boldsymbol{E} = [e_{u(s,i)}^k]$, $\forall s, \forall i, \forall k$ indicate the binary variable for PRB association. Furthermore, $M = [M_s^d, M_s^c]$, $\forall s$ is the matrix that shows the number of VNFs in each layer of slice. Also $\boldsymbol{Y} = [y_{s,d}]$ $\forall s, \forall d$ is a binary variable shown whether the physical DC is mapped to a VNFs of a slice or not. (14e) and (14f) indicate that the power of each UE is a positive integer value, and the power of each UE in each service does not exceed the maximum power of each service, respectively. Also, (14b) and (14c) shows that the rate of each UE requesting each type of service, i.e., eMBB, and URLLC, has a maximum and minimum, respectively. In addition, in (14d), the constraint supports that we have enough physical resources for VNFs of each slice. (14g) expressed the limited end-to-end delay of the received signal, respectively. (14h) and (14i) denoted the stability of the M/M/1 queue model. (14j) restricted the number of VNF in each slice due to the limited resources. (14k) shows that each PRB are associated to no more than one UE. In addition, (14l) indicates that the fixed cost of energy of VNFs in each slice does not exceed the threshold. Moreover, (14m) depicts that $\boldsymbol{E}$ is a matrix of binary variables. Also, (14n), guarantees the reliability of the URLLC services.

G. Proposed Algorithm and Numerical Results

Problem (14), is a two-time scale problem, i.e., large time scale and small time scale. On a large-time scale, we aim to minimize the power of servers and obtain the QoS of slices. The assignment of PRB to slices is implemented in this time scale. In the small-time scale, the assignment of PRB of slices to UEs is executed, and the optimal power is obtained. In this research, we aim to use the machine learning methods such as deep reinforcement learning and deep learning to train the O-RAN system and have an intelligent system. The deep Q-learning is implemented for the large-time scale and the multi-agent deep reinforcement learning contains DDPG (actor-critic algorithm), correlated q-learning, and the priority proportional fairness algorithm will be implemented for the small-time scale. The deep learning algorithms contain LSTM and recurrent neural networks. The proposed algorithm for the large-time scale is implemented in the following, and part of the numerical results is depicted.

1) Proposed Algorithm: Here we use the reinforcement learning method to solve the above two problems. In the Q-learning method for the large-time scale, an agent tries to find the optimal value in a specific environment. This interactive process is modeled as a Markov decision-making process that includes (S, A, R, P, γ). S represents the state space matrix, and A represents the action vector. R is also the reward of action. $P(.|S, a)$ is the probability of transfer and $\gamma \in (0, 1]$ is the discount factor. The $\Pi(.|S)$ policy is a mapping of the state to the distribution of actions. The value-state function for state s under policy $\Pi(.|S)$ with $V^{\Pi}(s)$ indicates that the expected return value in state s under policy $\Pi(.|S)$). The value of performing operation a in state s under the $\Pi(.|S)$ policy is represented by $Q^{\Pi}(s, a)$. We have the following relations on this basis.

$$V^{\Pi}(s) = \mathbb{E}_{\Pi,P}[\sum_{t=0}^{\infty} \gamma^t R_t | S_0 = s] \tag{15}$$

The Q-value is as below.

$$Q^{\Pi}(s, a) = \mathbb{E}_{\Pi,P}[\sum_{t=0}^{\infty} \gamma^t R_t | S_0 = s, A_0 = a]. \tag{16}$$

$\mathbb{E}$ represented the statistical average. Based on the Bellman equation we have

$$V^{\Pi}(s) = \mathbb{E}_{\Pi,P}[R + \gamma V^{\Pi}(s')] \tag{17}$$

and also,

$$Q^{\Pi}(s, a) = \mathbb{E}_{\Pi,P}[R + \gamma Q^{\Pi}(s', a')] \tag{18}$$

where, s' and a' can be obtained from $\Pi(.|s')$ and $P(.|s, a)$. The goal of reinforcement learnin is to obtain the optimal policy to maximize the $Q^{\Pi}(s, a)$. So, using Bellman equation, we have

$$Q^*(s, a) = \mathbb{E}_{\Pi^*,P}[R + \gamma Q^*(s', a')]. \tag{19}$$

Also, T^* is the Bellman operator

$$T^*Q(s, a) = \mathbb{E}_{\Pi^*,P}[R + \gamma Q(s', a')] \tag{20}$$

By using this operator iteratively, $Q_{t+1}(s, a) \leftarrow T^*Q(s, a)$ the algorithm can converge $Q_t(s, a) \rightarrow Q^*(s, a)$. $t \rightarrow \infty$ [20], [21]. In the Q-learning in each episode we have

$$Q(s_{t+1}, a_{t+1}) = Q(s_t, a_t) + \alpha[r_{t+1} + \gamma \max_{a \in A} Q(s_{t+1}, a) - Q(s_t, a_t)] \tag{21}$$

where, α is the learning rate.

References

[1] L. Gavrilovska, V. Rakovic, and D. Denkovski, "From Cloud RAN to Open RAN." *Wirel. Pers. Commun.*, vol. 113, no. 3, pp. 1523–1539, 2020.

[2] S. Niknam, A. Roy, H. S. Dhillon, S. Singh, R. Banerji, J. H. Reed, N. Saxena, and S. Yoon, "Intelligent O-RAN for beyond 5G and 6G wireless networks," *arXiv preprint arXiv:2005.08374*, 2020.

[3] N. Kazemifard and V. Shah-Mansouri, "Minimum delay function placement and resource allocation for Open RAN (O-RAN) 5G networks," *Computer Networks*, vol. 188, p. 107809, 2021.

[4] C. B. Both, J. Borges, L. Gonçalves, C. Nahum, C. Macedo, A. Klautau, and K. Cardoso, "System Intelligence for UAV-Based Mission Critical with Challenging 5G/B5G Connectivity," *arXiv preprint arXiv:2102.02318*, 2021.

[5] "O-RAN Architecture Description," O-RAN Alliance, Tech. Rep., 2020.

[6] O.-R. W. G. 2, "AI/ML workflow description and requirements," O-RAN Alliance, Tech. Rep., 2020.

[7] B.-S. Lin, "Toward an AI-Enabled O-RAN-based and SDN/NFV-driven 5G& IoT Network Era," *Network and Communication Technologies*, vol. 6, no. 1, pp. 6–15, 2021.

[8] M. Alsenwi, N. H. Tran, M. Bennis, S. R. Pandey, A. K. Bairagi, and C. S. Hong, "Intelligent resource slicing for eMBB and URLLC coexistence in 5G and beyond: A deep reinforcement learning based approach," *IEEE Transactions on Wireless Communications*, 2021.

[9] M. Yan, G. Feng, J. Zhou, Y. Sun, and Y.-C. Liang, "Intelligent resource scheduling for 5G radio access network slicing," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7691–7703, 2019.

[10] J. Mei, X. Wang, K. Zheng, G. Boudreau, A. B. Sediq, and H. Abou-zeid, "Intelligent Radio Access Network Slicing for Service Provisioning in 6G: A Hierarchical Deep Reinforcement Learning Approach," *IEEE Transactions on Communications*, 2021.

[11] F. Rezazadeh, H. Chergui, L. Christofi, and C. Verikoukis, "Actor-Critic-Based Learning for Zero-touch Joint Resource and Energy Control in Network Slicing," in *ICC 2021-IEEE International Conference on Communications*. IEEE, 2021, pp. 1–6.

[12] M. Setayesh, S. Bahrami, and V. W. Wong, "Joint PRB and Power Allocation for Slicing eMBB and URLLC Services in 5G C-RAN," in *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE, 2020, pp. 1–6.

[13] P. Yang, X. Xi, T. Q. Quek, J. Chen, X. Cao, and D. Wu, "How should I orchestrate resources of my slices for bursty URLLC service provision?" *IEEE Transactions on Communications*, vol. 69, no. 2, pp. 1134–1146, 2020.

[14] F. Saggese, M. Moretti, and P. Popovski, "Power Minimization of Downlink Spectrum Slicing for eMBB and URLLC Users," *arXiv preprint arXiv:2106.08847*, 2021.

[15] P. Korrai, E. Lagunas, S. K. Sharma, S. Chatzinotas, A. Bandi, and B. Ottersten, "A RAN resource slicing mechanism for multiplexing of eMBB and URLLC services in OFDMA based 5G wireless networks," *IEEE Access*, vol. 8, pp. 45 674–45 688, 2020.

[16] H. Zhou, M. Erol-Kantarci, and V. Poor, "Learning from Peers: Transfer Reinforcement Learning for Joint Radio and Cache Resource Allocation in 5G Network Slicing," *arXiv preprint arXiv:2109.07999*, 2021.

[17] J. Tang, W. P. Tay, T. Q. Quek, and B. Liang, "System cost minimization in cloud RAN with limited fronthaul capacity," *IEEE Transactions on Wireless Communications*, vol. 16, no. 5, pp. 3371–3384, 2017.

[18] P. Luong, F. Gagnon, C. Despins, and L.-N. Tran, "Joint virtual computing and radio resource allocation in limited fronthaul green C-RANs," *IEEE Transactions on Wireless Communications*, vol. 17, no. 4, pp. 2602–2617, 2018.

[19] P. Luong, C. Despins, F. Gagnon, and L.-N. Tran, "A novel energy-efficient resource allocation approach in limited fronthaul virtualized C-RANs," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. IEEE, 2018, pp. 1–6.

[20] P. R. Montague, "Reinforcement learning: an introduction, by Sutton, RS and Barto, AG," *Trends in cognitive sciences*, vol. 3, no. 9, p. 360, 1999.

[21] Y. Hua, R. Li, Z. Zhao, X. Chen, and H. Zhang, "Gan-powered deep distributional reinforcement learning for resource management in network slicing," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 2, pp. 334–349, 2019.