Elastic O-RAN Slicing for Industrial Monitoring and Control: A Distributed Matching Game and Deep Reinforcement Learning Approach

Sarder Fakhrul Abedin, *Member, IEEE,* Aamir Mahmood, *Senior Member, IEEE,* Nguyen H. Tran, *Senior Member, IEEE,* Zhu Han, *Fellow, IEEE,* and Mikael Gidlund, *Senior Member, IEEE,*

***Abstract*—In this work, we design an elastic open radio access network (O-RAN) slicing for the industrial Internet of things (IIoT). Unlike IoT, IIoT poses additional challenges such as severe communication environment, network-slice resource demand variations, and on-time information update from the IIoT devices during industrial production. First, we formulate the O-RAN slicing problem for on-time industrial monitoring and control where the objective is to minimize the cost of fresh information updates (i.e., age of information (AoI)) from the IIoT devices (i.e., sensors) while maintaining the energy consumption of those devices with the energy constraint as well as O-RAN slice isolation constraints. Second, we propose the intelligent O-RAN framework based on game theory and machine learning to mitigate the problem's complexity. We propose a two-sided distributed matching game in the O-RAN control layer that captures the IIoT channel characteristics and the IIoT service priorities to create IIoT device and small cell base station (SBS) preference lists. We then employ an actor-critic model with a deep deterministic policy gradient (DDPG) in the O-RAN service management layer to solve the resource allocation problem for optimizing the network slice configuration policy under time-varying slicing demand. While the matching game helps the actor-critic model, the DDPG enforces the long-term policy-based guidance for resource allocation that reflects the trends of all IIoT devices and SBSs satisfactions with the assignment. Finally, the simulation results show that the proposed solution enhances the performance gain for the IIoT services by serving an average of 50% and 43.64% more IIoT devices than the baseline approaches.**

***Index Terms*—Industrial IoT, distributed management and computation, O-RAN slicing, age of information, game theory, deep reinforcement learning.**

I. Introduction

The fifth-generation (5G) and beyond of wireless communications have ignited a wide range of industrial Internet of Things (IIoT) use-cases for Industry 4.0, such as process automation monitoring and predictive maintenance. However, such IIoT use-cases have posed a significant challenge of diverse Quality-of-Services (QoS) requirements regarding data rate, latency and reliability, and priority, as shown in Table I. As a result, the network softwarization [1] is considered as potential trend that transforms the existing IIoT networks into the software-based network slicing solutions which can operate on the general-built physical network infrastructures [2]. Within the current virtual RAN (vRAN) and cloud RAN (C-RAN) concepts, the network slicing offers a natural solution to accommodate diverse QoS requirements of the IIoT applications. However, the traditional RAN slicing involves large vendors creating proprietary hardware, and software [3]. As a result, in 2018, the O-RAN alliance is formed to realize the next-generation cellular networks with flexible multi-vendor network infrastructure to the telecom operators [4]. By deploying O-RAN, the network operators can significantly reduce the operational cost in a dense IoT environment compare to vRAN and C-RAN [5].

TABLE I: QoS requirements for Industrial use-cases [6].

Use-cases	Latency (ms)	Reliability	Pakcet size (B)	Priority level
Emergency systems and action	50	$1-10^{-6}$	24	1
Scale reading	100	$1-10^{-6}$	512	2
Mobile robots	500	$>>$ $1-10^{-6}$	250	3

The benefit of network slicing and QoS management for the IIoT environment under O-RAN differs from the traditional RAN slicing and management solutions in several critical aspects. First, the IIoT devices are mostly deployed in a harsh factory environment to support time/mission-critical applications (Table I). Therefore, to enforce up-to-date control decisions, the remote factory management system requires on-time and updated environmental data collection [7]. Meanwhile, the transmission path between the IIoT devices and a nearby base station (BS) or gateway can be obstructed by large physical objects causing multi-path fading and signal attenuation [8]. Consequently, intelligent O-RAN slice and QoS management for mission and safety-critical industrial services can significantly enhance the 5G key performance indicators (KPIs) as well as the data freshness metric such as age of information (AoI) [9]. Besides, the intelligent O-RAN slicing enforces policy-based resource management and RIC (RAN Intelligent Controller) near-RT functions to achieve the desired performance gain for the IIoT. Second, the IIoT network resource demand changes frequently during the production time in a factory that leads towards the requirement of elastic network slicing and management. Therefore, unlike the traditional RAN slicing, the O-RAN in the IIoT environment can provide near-RT RIC control loops that are highly responsive to enable elastic IIoT network slicing, and resource scheduling [10] for the services that require frequent and fresh environmental data collection to invoke up-to-date industrial control decisions.

Most of the works on vRAN and C-RAN based network slicing for industrial IoT and IoT [11–15] mainly focus on

Sarder Fakhrul Abedin, Aamir Mahmood, and Mikael Gidlund are with Department of Information Systems and Technology, Mid Sweden University, 851 70 Sundsvall, Sweden (E-mail: {sarder.abedin, aamir.mahmood, mikael.gidlund}@miun.se.

Nguyen H. Tran is with the School of Computer Science, The University of Sydney, Sydney, NSW 2006, Australia (E-mail: nguyen.tran@sydney.edu.au)

Zhu Han is with the Electrical and Computer Engineering Department, University of Houston, Houston, TX 77004 (E-mail: : zhan2@uh.edu).

a single objective such as energy efficiency, where the data freshness parameter is overlooked. However, for different types of industrial monitoring and control services, the age of information (AoI) [16] is a key performance metric that allows the factory management application to take up-to-date remote monitoring-based control decisions. Besides, unlike the conventional QoS management [17], the process automation monitoring in the IIoT also requires fresh information updates from different low-power and often battery-operated IIoT sensors/actuators for efficient predictive maintenance. Such a situation further intensifies the QoS management and O-RAN network slicing for the industrial communication where *a balance should be maintained between the transmission energy consumption of the IIoT devices and the fresh information updates for sustainable on-time industrial process automation monitoring and control.*

Under the above circumstances, we focus on optimizing the elastic O-RAN slicing in a dynamic IIoT environment with QoS and contextual constraints. The main contributions of the paper are summarized as follows,

- First, we formulate the optimization problem in the IIoT network under contextual constraints such as energy, data rate, elastic O-RAN slicing, and AoI metric. We also design a non-linear age-penalty function that reflects the priority-aware data update behavior while incorporating the multi-hop transmission and queuing delays for IIoT monitoring and control applications. Then, we show that the formulated problem is NP-hard.
- Second, to deal with the NP-Hard problem, we model the IIoT association problem as a Hospitals/Residents problem that provides the radio connection management for the O-RAN control and solves it using the one-to-many matching game. The proposed distributed approach ensures stable associations between the IIoT devices and SBSs by enforcing two-sided decision-making to optimized resource allocation for the elastic network slices.
- Third, we employ an actor-critic reinforcement learning-based deep deterministic policy gradient (DDPG) with *buffer*, which can ensure a trade-off between energy-efficient IIoT communications and optimized AoI metric. In the proposed trained model, an elastic slicing configuration for the IIoT network is obtained where one-to-many matching-based IIoT associations are considered as an input for modeling the observable network states over time. Besides, we design the state, observation, action space, and reward explicitly for the proposed DDPG, effectively enforcing the long-term policy-based guidance for coordinating resource demands of multiple SBSs.
- Finally, we perform an extensive experimental analysis to evaluate the proposed approach's performance. We also conduct an extensive simulation analysis to evaluate the proposed approach's learning efficiency concerning the key performance indicators. The results show that the association and allocation policy that is obtained by applying the proposed approach achieves significant performance gain by increasing service fairness while reducing the energy and AoI costs for different types of industrial monitoring and control services compared to the baseline approaches.

The remainder of the paper is organized as follows. In Section II, we present an extensive literature review based on the current research. In Sections III, we present the system model and problem formulation, respectively. Section IV explains in detail how we solve the proposed optimization problem with game theory and DRL techniques. In Section V, we present the simulation analysis to validate the performance and efficiency of our proposed approach for elastic network slicing. Finally, in Section VI we conclude the discussion.

II. Literature Review

This section has classified the related works in the following sub-sections based on network slicing, energy efficiency, AoI, and machine learning and artificial intelligence in IIoT, respectively. We also provide a summary of challenges that we face to archive the goal of seamless elastic O-RAN network slicing for IIoT.

A. Network Slicing

Network slicing for 5G enabled industrial IoT networks has received much attention in recent years. In [14], the authors considered the advantages of deploying the private 4G and 5G networks in industrial environments, and provide insight into the significance of network slicing for performance guarantees in multiservice co-existence scenarios. In [13], the authors proposed a federated-orchestrator (F-orchestrator) that coordinates the spectrum and computational resources without sharing the local data and resource information from BSs. The distributed resource allocation algorithm is based on the Alternating Direction Method of Multipliers with Partial Variable Splitting (DistADMM-PVS) to minimize the network's average service response time. In [18], the authors demonstrated network slicing as an enabler for flexible and efficient IIoT networks. In [6], the authors adopted a multi-objective approach and proposed online Gaussian mixture model clustering (OGMMC) and dynamic mini-batch gradient descent (MBGD) algorithms to ensure the allocation of the resources to the slices depending on the bandwidth, delay, and reliability requirements. *However, most of the work on the network slicing for industrial IoT focuses on the resource allocation and QoS enhancement objectives where the data freshness metric is critical for industrial predictive maintenance and control decisions.*

B. Energy Efficient Network Slicing

Energy efficiency has been a core issue in the IIoT environment. In [19], the authors formulated a joint uplink and downlink energy-efficient resource management decision-making problem (i.e., network selection, subchannel assignment, and power management) as a Markov decision process (MDP). Subsequently, the authors proposed a deep post-decision state (PDS)-based experience replay and transfer (PDS-ERT) reinforcement learning algorithm to learn the optimal policy. In [20], the authors formulated an optimization-based RAN slicing problem where the objective is to maximize the total resource allocation success probabilities of all IoT devices to provide energy-efficient ultra-reliable and low

latency (URLLC) services. In [21], the authors considered providing QoS-aware urgent and reliable communications and proposed a game-theoretic distributed slicing strategy over an SDN-based Long Range Wide Area Network (LoRaWAN) architecture. *However, further study on the trade-off between the energy efficiency and the fresh information update from the IIoT devices is required under the industrial communication channel modeling.*

C. Age of Information

AoI is a vital parameter for objectively measuring timeliness of information updates for industrial monitoring and control services. In [22], the authors considered minimizing the long-term average AoI under the limited average transmission power at the source by formulating a Constrained Markov Decision Process (CMDP) problem. To solve such problem, the author proposed recast the original problem to an unconstrained MDP through Lagrangian relaxation. In [23], the authors studied an age minimization problem over a wireless broadcast network to keep many users updated on timely information, where only one user can be served at a time. The formulated MDP objective is to find dynamic scheduling algorithms to minimize the long-run average age. In [24], the authors formulated an energy-efficient trajectory optimization problem in which the objective is to maximize the energy efficiency by optimizing the UAV-BS trajectory policy under the energy and age of information (AoI) constraints to ensure the data freshness at the ground BS. *Unlike the linear AoI functions in the existing works, in this paper we propose an age-penalty function that incorporates the backhaul transmission and queuing delays to facilitate the multi-hop data update from the IIoT devices to multi-access edge server (MEC) via the small cell network.*

D. ML/AI based Network Slicing

Machine learning and artificial intelligence have been widely used to solve different challenges in the IIoT environment. In [25], the authors proposed a proactive, dynamic network slicing scheme that utilizes a deep-learning-based short-term traffic prediction approach for 5G transport networks. In [26], the authors proposed a deep reinforcement learning (DRL) approach to minimize long-term system energy consumption in a computation offloading scenario with multiple IIoT devices and multiple fog access points (F-APs). In [15], the authors proposed a novel deep RL scheme to provide federated and dynamic QoS-aware network management and slice resource allocation (i.e., transmission power and spreading factor) for differentiated QoS services in future IIoT networks. In [27], the authors evaluated the performance of different supervised-learning algorithms with varying complexity for the inference of the radio-link state in the IIoT. *Unlike the existing ML/AI-based solutions for IIoT, in this work, we have focused on optimizing the elastic network slicing policy that captures the priority-aware data update behavior of IIoT applications (i.e., Table I) under O-RAN.*

III. System Model

In Fig. 1, we consider one macro-cell base station (MBS) co-located with the multi-access edge server (MEC-BS) and

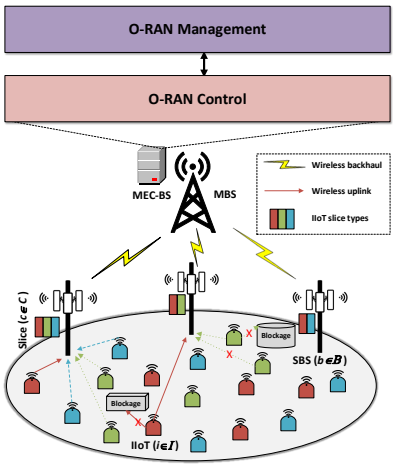

Fig. 1: O-RAN system model for IIoT environment.

a set of small-cell base stations (SBS), $\mathcal{B} = \{1, 2, \cdots, B\}$ which are all managed by the network infrastructure provider (InP). We consider an O-RAN architecture that consists of a set of $\mathcal{C} = \{1, 2, \cdots, C\}$ IIoT slice instances for industrial monitoring and control slice type and priorities, where c represent the slice number. These slices are built on the unified physical infrastructure and share the same resources, consisting of the MEC-BS network. For the O-RAN slicing, the set of physical resource blocks (PRBs) are defined as, $\mathcal{P} = \{1, 2, \cdots, P\}$. Moreover, each $b \in \mathcal{B}$ serves a set of IIoT devices $\mathcal{I} = \{1, 2, \cdots, I\}$ that belongs to different types of slices $c \in \mathcal{C}$. We assume a contract between the network operator, factory, and InP for the on-demand and orthogonal PRB deployment to the network slices in the IIoT environment. The factory receives on-demand slice resources within the contract based on the number of IIoT devices and their network resource demands at a particular industrial production period. Furthermore, the association between the IIoT devices and the SBSs may also change frequently according to the channel conditions (e.g., blockage, Non-line-of-sight (NLoS) propagation). In such a dynamic association scenario, the number of PRB demands of the IIoT devices to the corresponding SBSs may also vary during the industrial production period. Therefore, the InP re-configures the slice $s \in \mathcal{S}$ (i.e., increasing/decreasing the number of PRB) deployed at $b \in \mathcal{B}$ based on the dynamic demand (i.e., increasing/decreasing the number of IIoT devices). In practice, the InP will charge the operator based on the slice reconfiguration request and maintain the QoS of the IIoT devices $i \in \mathcal{I}$ by limiting the number of IIoT devices of each SBS slice type $c \in \mathcal{C}$. Such slice reconfiguration based on the PRB dynamic demands of the IIoT devices is essential for the operator to prevent the QoS degradation for that particular slice's IIoT devices. This approach will also provide necessary slice isolation even in the dynamic and modular industrial production systems.

A. Traditional RAN to O-RAN in IIoT

Unlike the traditional RAN network functions, in the O-RAN architecture, the network functions are equipped with embedded intelligence in two layers: the O-RAN management layer and the O-RAN control layer. The O-RAN management layer supports the intelligent RAN optimization based on RAN

data collected from the O-RAN nodes. More specifically, this layer receives highly reliable data from the modular centralized unit (CU) and distributed unit (DU) in a standardized format. The core RAN optimization solution operates at this layer and is owned by the network operator. The layer also provides the O-RAN system's capability to optimize the network operator's PRB allocation policies and other objectives. On the other hand, the O-RAN control layer provides connectivity and QoS management services to the network operators. The detailed design of the O-RAN control and management layers are provided in the later section.

B. QoS Model

This sub-section provides the detailed QoS model for the O-RAN control and management layers based on transmission capacity and energy consumption of the IIoT devices, backhaul transmission and queuing delay analysis, and AoI of the IIoT devices.

1) Transmission capacity and energy consumption: The transmission capacity for the IIoT device $i \in \mathcal{I}$ that uses slice $c \in \mathcal{C}$ of SBS $b \in \mathcal{B}$ at t is defined as,

$$R_i(\theta^c_{i,p}, \alpha^c_{i,b}, t) = \begin{cases} \theta^c_{i,p}(t) \cdot \beta^c_{p,b} \cdot \alpha^c_{i,b}(t) \log\Big(1 + \gamma^c_{i,b}(t)\Big), \\ \qquad\qquad\qquad \text{if } \alpha^c_{i,b}(t) = 1, \\ 0, \quad \text{otherwise.} \end{cases} \tag{1}$$

In (1), at time t, $\alpha^c_{i,b}(t)$ is the binary indicator variable where $\alpha^c_{i,b}(t) = 1$ indicates if the user i is served by the slice $c \in \mathcal{C}$ at the SBS $b \in \mathcal{B}$; otherwise $\alpha^c_{i,b}(t) = 0$. $\beta^s_{p,b}$ is the bandwidth of PRB $p \in \mathcal{P}$ for the slice c at SBS $b \in \mathcal{B}$. $\theta^c_{i,p}$ is the number of PRB allocated for slice $c \in \mathcal{C}$ to serve corresponding $i \in \mathcal{I}$ to maintain the QoS. The residual PRB for the slices in $b \in \mathcal{B}$ is, $\hat{\theta}_b(t) = \theta^{max}_b - \sum_{c=1}^{C}\sum_{p=1}^{P}\sum_{i=1}^{I} \alpha^c_{i,b}(t) \cdot \theta^c_{i,p}(t)$ where θ^{max}_b is the maximum number of PRB that can be allocated for the slices of SBS $b \in \mathcal{B}$. $\gamma^b_{i,p}(t) = \frac{p_{i,b}(t) h^b_{i,p}(t)}{\sigma^2}$ is the SNR where $p_{i,b}(t)$, $h^b_{i,p}(t)$, and σ are the transmission power, channel grain, and noise, respectively. The channel gain between $i \in \mathcal{I}$ and $b \in \mathcal{B}$ for using the PRB $p \in \mathcal{P}$ at t is given as,

$$h^b_{i,p}(t) = 10^{\frac{-PL_i[dB]}{10}}, \tag{2}$$

The NLoS path loss of the IIoT device $i \in \mathcal{I}$ in dB is calculated as [28],

$$PL_i[dB] = A \log_{10}(d_{i,b}) + B + E \log_{10}\left(\frac{f_c}{5}\right) + X. \tag{3}$$

In (3), $d_{i,b}$ is the distance between IIoT device $i \in \mathcal{I}$ and SBS $b \in \mathcal{B}$. The carrier frequency is denoted as f_c GHz. The constants A, B, E depend on the propagation model. X indicates the wall penetration loss in the NLoS IIoT scenario.

Using (1), the transmission energy consumption of $i \in \mathcal{I}$ at time t is calculated as,

$$\mathcal{E}_i(t) = \sum_{b=1}^{B}\sum_{c=1}^{C}\sum_{p=1}^{P} \frac{p_{i,b}(t) M^c_{i,b}}{R_i(\theta^c_{i,p}, \alpha^c_{i,b}, t)}. \tag{4}$$

In (4), $M^c_{i,b}$ is the data length that is received from $i \in \mathcal{I}$ of slice $c \in \mathcal{C}$ to SBS $b \in \mathcal{B}$. When $\alpha^c_{i,b} = 1$, the energy consumption of each $i \in \mathcal{I}$ is proportional to the amount of uplink data to the SBS $b \in \mathcal{B}$.

2) Backhaul transmission and queuing delay: We model the MEC-BS as M/M/k queue [29] where the MEC-BS has k parallel virtual computing resource blocks (vCRB) and single queue $\mathcal{Q}_{mec}$. We assume that the arrival and service processes of the traffic from the SBSs follow the Poission distribution. The arrival rate and the service rate are denoted as λ_{mec} and μ_{mec}, respectively. In this case, the backhaul queuing delay of each SBS $b \in \mathcal{B}$ depends on the aggregated backhaul traffic of other SBSs $b' \in \mathcal{B}\backslash\{b\}$, which can be computed by combining the traffic arrival rates as, $\lambda_{mec} = \lambda_b + \sum_{b' \in \mathcal{B}} \lambda_{b'}$. The MEC utilization is calculated as,

$$\rho_{mec} = \frac{\lambda_{mec}}{k \times \mu_{mec}}. \tag{5}$$

Using (5), to transmitting data size $M_b = \sum_{t=0}^{T}\sum_{c \in \mathcal{C}}\sum_{p \in \mathcal{P}}\sum_{i \in \mathcal{I}} R_i(\theta^c_{i,p}, \alpha^c_{i,b}, t)$, the total delay including the backhaul transmission delay and queuing delay is calculated as,

$$d_{b,mec} = \begin{cases} \frac{M_b}{R^{mec}_b(t)} + \frac{\mathcal{C}(k,\rho_{mec})}{k \cdot \mu_{mec} - \lambda_{mec}}, \\ \qquad \text{if } \sum_{c \in \mathcal{C}}\sum_{i \in \mathcal{I}} \alpha^c_{i,b}(t) \geq 1, \\ 0, \quad \text{otherwise.} \end{cases} \tag{6}$$

Here, $\mathcal{C}(k, \rho_{mec})$ is known as the Erlang's C formula [30]. In this case, Erlang's C is well-suited for meeting the computational service demands from the SBSs in $\mathcal{B}$. With an increasing vCRB service requests from the SBSs; some SBS service requests fail to access free vCRB at the MEC-BS. As a result, without rejecting the vCRB service requests from the SBSs, the MEC-BS adds the SBSs service requests to the queue $\mathcal{Q}_{mec}$ until there are free vCRBs at the MEC-BS. The wireless backhaul capacity between $b \in \mathcal{B}$ and the MEC-BS is defined as,

$$R^{mec}_b(t) = \frac{\beta_{mec}}{|\mathcal{B}|} \log\Big(1 + \gamma_{b,mec}(t)\Big). \tag{7}$$

In (7), $\gamma_{b,mec}(t)$ is the SNR between SBS $b \in \mathcal{B}$ and the MEC-BS and the wireless backhaul bandwidth, β_{mec} is equally distributed to the SBSs.

3) AoI: The AoI of IIoT $i \in \mathcal{I}$ for using SBS $b \in \mathcal{B}$ evolves and calculated at MEC-BS as,

$$\begin{aligned} \delta_i(t+1) = (1 - \alpha^c_{i,b}(t))\left[\frac{1}{\xi_i} \cdot f(\delta_i(t))\right] \\ + \alpha^c_{i,b}(t)\left[\frac{M^c_{i,b}}{R_i(\theta^c_{i,p}, t)} + d_{b,mec}\right]. \end{aligned} \tag{8}$$

In (8), $f(\cdot)$ is the age penalty function which is a non-negative and non-decreasing function where ξ_i is the priority of the IIoT i as per Table I. Usually, the age penalty function is an exponential function and useful for such control applications where the desire for data refreshing grows quickly with respect to the age [31]. Overall, the first part penalizes the system when the high priority IIoT $i \in \mathcal{I}$ is not scheduled frequently. The first part states that, if $i \in \mathcal{I}$ is not scheduled at $b \in \mathcal{B}$ for information update at the MEC-BS, the AoI of $i \in \mathcal{I}$ at the time t is equal to the exponential age penalty and AoI of IIoT $i \in \mathcal{I}$. Otherwise, the AoI of $i \in \mathcal{I}$ is the transmission delay between $i \in \mathcal{I}$ and $b \in \mathcal{B}$, and the total delay $d_{b,mec}$.

C. Problem Formulation

Based on the above model, the energy consumption of the IIoT devices and AoI functions for the industrial environment's predictive maintenance are controlled by two decision variables, θ, and α. More specifically, the objective of the problem is to minimize the sum of probability that the AoI of each IIoT device exceed a predefined threshold ξ_i. Furthermore, the objective is mapped with respect to the QoS constraints for the elastic network slicing. Therefore, the optimization problem is defined as follows,

$$\min_{\alpha,\theta} \sum_{i}^{I} \mathbf{Pr}\{\delta_i > \xi_i\}, \tag{9}$$

subject to

$$\frac{1}{T \cdot I} \sum_{t}^{T} \sum_{i}^{I} \mathcal{E}_i(t) \leq \mathcal{E}', \forall i \in \mathcal{I}, \tag{10}$$

$$0 \leq \sum_{i \in \mathcal{I}} \alpha_{i,b}^{c}(t) \leq \alpha_{b,c}^{max}, \forall b \in \mathcal{B}, \forall c \in \mathcal{C}, t = 0, 1, \cdots, T, \tag{11}$$

$$\alpha_{i,b}^{c}(t) \in \{0, 1\}, \forall i \in \mathcal{I}, \forall b \in \mathcal{B}, \forall c \in \mathcal{C}, t = 0, 1, \cdots, T \tag{12}$$

$$\sum_{p \in \mathcal{P}} \theta_{i,p}^{c}(t) \geq \theta_{i}^{min}, \forall c \in \mathcal{C}, \forall i \in \mathcal{I}, t = 0, \cdots, T, \tag{13}$$

$$\sum_{p \in \mathcal{P}} \theta_{i,p}^{c}(t) \leq 1, \forall c \in \mathcal{C}, t = 0, \cdots, T, \tag{14}$$

$$\sum_{c \in \mathcal{C}} \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{I}} \alpha_{i,b}^{c}(t) \cdot \theta_{i,p}^{c}(t) \leq \theta_{b}^{max}, \forall b \in \mathcal{B}, t = 0, \cdots, T. \tag{15}$$

Constraint (10) ensures restricting the objective of minimizing the average energy consumption of the IIoT devices that do not exceed the maximum energy threshold $\mathcal{E}'$. The energy threshold $\mathcal{E}'$ ensures a trade-off between two objectives that are receiving fresh updates from the IIoT devices while restricting the energy drainage of the battery-operated IIoT devices. (11)-(15) are the constraints for the decision variables $\alpha_{i,b}^{c}$ and $\theta_{i,p}^{c}$ of problem (9). At each t, constraints (11) and (12) ensure limiting the number of IIoT devices that is served by each slice $c \in \mathcal{C}$ assigned to $b \in \mathcal{B}$ to maintain the QoS of the slice $c \in \mathcal{C}$. Constraints (13)-(15) provide an elastic way of O-RAN slice isolation and management. Constraint (13) ensures the number of PRB allocations for the slice types should meet the minimum demands θ_i^{min} of the IIoT devices. Constraints (14) indicate each PRB $p \in \mathcal{P}$ can not be deployed to multiple slices belonging to multiple SBSs, and can only be assigned to at most one $c \in \mathcal{C}$ at each time t. Constraint (15) ensures that the amount of elastic PRB allocation $\theta_{i,p}^{c}(t)$ for the associated IIoT devices $\alpha_{i,b}^{c}(t)$ belong to the slice $c \in \mathcal{C}$ at SBS $b \in \mathcal{B}$ should not exceed the maximum number of PRBs per SBS, θ_b^{max} at time t. The decision problem in (9) is a stochastic optimization problem with the corresponding constraints (10)-(15) where constraints (11) and (12) are integer (non-convex) constraints. In a practical setting, the dynamic channel conditions and PRB demands per IIoT slice show the non-casual behavior in the O-RAN system for fixed θ_b^{max} and $\alpha_{b,c}^{max}$. In addition, the elastic slicing decision for the problem (9) also requires considering different IIoT slice types $c \in \mathcal{C}$ which are competing over the available PRBs in $\mathcal{P}$ while maintaining the O-RAN slice isolation and management constraints (13)-(15). Such kind of integer-constrained problem is known to be NP-hard even in a deterministic setting. As a result, very limited combinatorial problems in this class of discrete optimization are known to be solvable and can achieve optimal results in polynomial time.

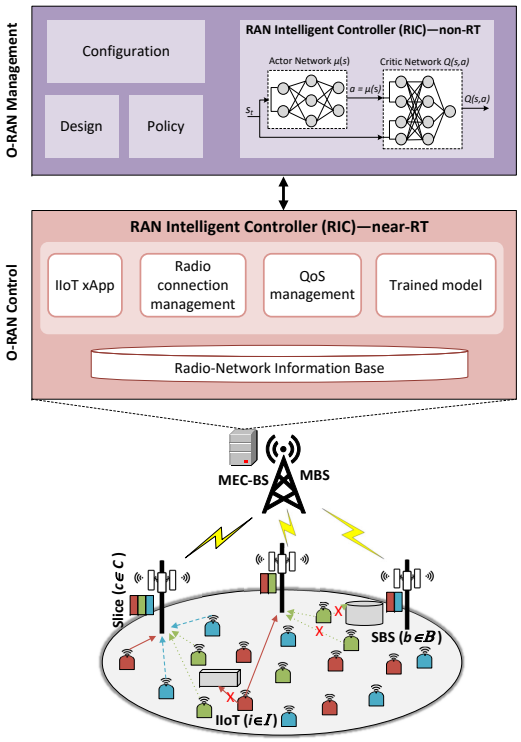

Fig. 2: Intelligent RAN modular design for elastic slicing and QoS management.

In the next section, we solve the problem (9) with the corresponding constraints (10)-(15) using the matching game to solve the IIoT association problem and the reinforcement learning for solving the O-RAN slicing problem in the RIC design.

IV. Matching Game and Deep Reinforcement Learning for Elastic O-RAN Slicing

The problem in (9) is solved under the RICs' modular design[1]. In Fig. 2, the intelligent O-RAN modular design for elastic slicing and QoS management is comprised of two layers, the O-RAN control layer, and the O-RAN management layer. In Section IV(A) and IV(B), we provide a detailed discussion of the O-RAN layers' modules and interactions.

A. O-RAN Control Layer

First, we perform the radio connectivity management using the *radio connection management* module in the RAN near-RT, where we find the stable IIoT association distributedly using matching game. Using the matching game outcomes, we measure the IIoT slice requests' demand and generate IIoT service-specific slice templates considering the slicing

[1]xApps can be deployed by multiple sources (e.g., network operator, factory management) to include new features as 5G network control applications (e.g., intelligent security applications [32])

constraints at the *QoS management* module in the RIC near-RT. The reason behind applying the matching game is to generate a stable radio connection between the IIoT devices and the SBSs that enable the O-RAN optimization process to take two-sided decision-making distributedly by each network entities. The standardized interfaces[2] between the O-RAN management and O-RAN control layers and the observational data from the SBSs are stored in the radio-network information base (RNIB). Later, the SBS measurement data are sent to the RIC non-RT at the O-RAN management layer.

B. O-RAN Management Layer

The O-RAN management layer hosts the DDPG, which combines Q-learning and policy gradients to solve the resource allocation problem with the corresponding QoS constraints. The *design, policy, and configuration* modules are used to design and select the learning model (e.g., the actor-critic model) and send the trained module from RIC non-RT to the RIC near-RT at the O-RAN control layer. Once the trained model is received at the RIC near-RT, it transmits corresponding resource allocation actions to the SBSs. The primary benefit of combining the two-sided matching game and DDPG for O-RAN optimization is that the matching game helps the DDPG enforce the long-term policy-based guidance for resource allocation while taking advantage of a guaranteed and stable radio connection, reflecting the trends and patterns of all IIoT devices and SBSs satisfaction with the assignment. Furthermore, once the learning module performs the model training by observing the matching game's output, the radio connection management does require to execute frequently with dynamic change in the IIoT environment. In this way, the computational complexity is greatly reduced in the intelligent O-RAN optimization process.

In Section IV(C), we design the radio connection management with a matching game and find stable IIoT association and the slice demand. Subsequently, in section IV(D), we model the state and action spaces and reward function of the actor-critic model with DDPG for elastic O-RAN slicing, which is deployed at the RIC non-RT to solve problem (9) using outcome of the matching game.

C. Radio Connection Management via Distributed Matching

To obtain the IIoT association with the corresponding SBS $\alpha^c_{i,b}(t)$, we model the association problem as an one-to-many Hospitals/Residents (HR) problem [34]. The association problem involves a set of IIoT devices $\mathcal{I}$ as residents and a set of SBSs $\mathcal{B}$ as hospitals with finite PRB capacity $\alpha^{max}_{b,c}$ for the slices in $\mathcal{C}$. Each of IIoT $i \in \mathcal{I}$ has a preference list $\mathcal{P}_i \subset \mathcal{B}, \forall i \in \mathcal{I}$ where IIoT i ranks a subset of $\mathcal{B}$ in a descending order based on SNR. On the other hand, each SBS $b \in \mathcal{B}$ has a preference list $\mathcal{P}_b, \subset \mathcal{I}, \forall b \in \mathcal{B}$ over a subset of IIoT $\mathcal{I}$ in an ascending order based on their priority. A pair $(i, b) \in I \times B$ is defined as acceptable if b and i appear in $\mathcal{P}_i$ and $\mathcal{P}_b$, respectively. A set of acceptable pairs is defined as a set $\mathcal{A}$ and an assignment $\alpha^c_{i,b} \in \tilde{\mathcal{A}} \subset \mathcal{A}$ represents the mutual acceptance between i and b over the slice s when $\alpha^c_{i,b} = 1$, otherwise $\alpha^c_{i,b} = 0$.

[2]The standard interfaces are A1 between the RIC non-RT and RIC near-RT for policy transfer, E2 for IIoT slice specific measurements and data collection [33].

Algorithm 1: Radio connection management with distributed matching game

1 **Step 1: Initialization**
2 Define $\tilde{\mathcal{A}} \coloneqq \emptyset, \mathcal{P}_i, \mathcal{P}_b$
3 **Step 2: Radio connection management**
4 **while** *some* $i \in \mathcal{I}$ *are unassigned and* i *has a nonempty list* **do**
5 $b \coloneqq$ first in $\mathcal{P}_i$;
6 Update $\tilde{\mathcal{A}} \coloneqq \tilde{\mathcal{A}} \cup \{(i, b)\}$
7 **if** $\sum_{c\in\mathcal{C}} \sum_{i\in\mathcal{I}} \alpha^c_{i,b} > \alpha^{max}_{b,c}$ **then**
8 $i' \coloneqq$ worst in $\tilde{\mathcal{A}}(b)$ in $\mathcal{P}_b$;
9 Update $\mathcal{A} \coloneqq \mathcal{A} \backslash \{(i, b)\}$;
10 **if** $\sum_{c\in\mathcal{C}} \sum_{i\in\mathcal{I}} \alpha^c_{i,b} == \alpha^{max}_{b,c}$ **then**
11 $i' \coloneqq$ worst in $\tilde{\mathcal{A}}(b)$ in $\mathcal{P}_b$;
12 **for** *each successor* i'' *of* i' *in* $\mathcal{P}_b$ **do**
13 Remove pair i'', b

Definition 1 (Distributed matching game): A matching $\tilde{\mathcal{A}}$ is an assignment between $i \in \mathcal{P}_b$ and $b \in \mathcal{P}_i$ such that,

(a) $|\tilde{\mathcal{A}}(i)| \leq 1, \forall i \in \mathcal{I}$,
(b) $|\tilde{\mathcal{A}}(b)| \leq \alpha^{max}_{b,c}, \forall b \in \mathcal{B}$.

Condition (a) indicates that each IIoT device $i \in \mathcal{I}$ can be assigned at most one SBS at a time, and no IIoT device will be assigned to any SBS from which it got rejected. Condition (b) states that the SBSs can only assign a certain number of IIoT devices at a time to reduce overloading.

In the step 1 in Alg. 1, the assignment set $\tilde{\mathcal{A}}$ is initialized empty and the preference lists $\mathcal{P}_i$ and $\mathcal{P}_b$ are created for the corresponding $i \in \mathcal{I}$ and $b \in \mathcal{B}$ (lines 1-2, in Alg. 1). The association starts in step 2 in Alg. 1 where each unassigned $i \in \mathcal{I}$ proposes to the most preferred b in its preference list $\mathcal{P}_i$ until i has a nonempty list (line 5, in Alg. 1). The assignment set $\tilde{\mathcal{A}}$ is updated with the pair (i, b) and if the capacity of the SBS b is over-subscribed, $\tilde{\mathcal{A}}$ is updated accordingly (lines 6-9, in Alg. 1). If the SBS b capacity is full, the pair is deleted and removed from the list (lines 10-13, in Alg. 1). Finally, the matching outcome $\alpha^c_{i,b} \in \tilde{\mathcal{A}} \subset \mathcal{A}$ is generated when there is no blocking pair [35].

D. Actor-Critic Model with DDPG for RIC non-RT

The proposed DDPG for elastic slicing is combined with the actor-critic model with four neural networks and deployed at the O-RAN management layer. More specifically, the proposed actor-critic model with DDPG includes a Q-network and corresponding target Q-network for the *critic* part, and for the *actor* network, there are a deterministic policy network and target policy network. In the traditional baseline deep Q-network (DQN) with experience replay, the learning model generates a probability distribution over actions, and the optimal action is taken over the Q-values of all actions. However, in the proposed actor-critic model, the actor maintains a policy network where the state information for elastic slicing is taken as input, and in return, the actor model generates the exact

action (i.e., continuous). On the other hand, the critic maintains a Q-value network that takes the same input as the actor-network state and action to generate the Q-values as output. The main advantage of DDPG over the vanilla actor-critic and baseline DQN is that DDPG is an "off-policy" method in the continuous action setting, and the "deterministic" policy by the actor computes the action directly without considering the probability distribution over actions. Due to such reasons, in this work, we propose the actor-critic with DDPG to solve the elastic O-RAN slicing problem in (9) with constraints (10)-(15) that provides a significant improvement over the vanilla actor-critic and baseline DQN.

In the following sub-sections, we provide detailed discussion on the DDPG with actor-critic model that includes the design of the state and action spaces, and reward function.

1) State and action spaces: State $s_t \in \mathcal{S}$ at time period t represents the state of SBS $b \in \mathcal{B}$ that includes all the residual capacities, AoI metric values, and energy consumption of all IIoT service types at SBSs $b \in \mathcal{B}$. Therefore, the state space s_t is defined as,

$$s_t = \{\hat{\theta}(t-1), \delta(t-1), \mathcal{E}(t-1)\}. \tag{16}$$

In (16), state s_t includes AoI of IIoT devices as $\delta = \{\delta_i(t-1)\}_{i\in\mathcal{I}}$ which represents the individual AoI of the IIoT device $i \in \mathcal{I}$ and time $t-1$. The residual capacity of the SBSs in state s_t is defined as $\hat{\theta} = \{\hat{\theta}_b(t-1)\}_{b\in\mathcal{B}}$. The energy consumption of the IIoT devices for associating with corresponding SBSs in time $t-1$ is represented as, $\mathcal{E} = \{\mathcal{E}_i(t-1)\}_{i\in\mathcal{I}}$. In the state space design, $\hat{\theta}$ captures the slice demands from the IIoT associations with corresponding SBSs, total number of PRB allocation to all the IIoT devices of different slice types (i.e., using (1)) generated using the matching game using Alg. 1 at time step $t-1$. At each time t, an action a_t indicates the number of PRB allocation for the slices deployed in SBS $b \in \mathcal{B}$ and defined as,

$$a_t = \{a_{b,t}\}_{b\in\mathcal{B}}. \tag{17}$$

In (17), action space a_t is comprised of PRB allocation actions for all the slice types of each SBS $b \in \mathcal{B}$. The number of PRB allocation per SBS $b \in \mathcal{B}$ is distributed to the different slice types as per the slice requirements of the IIoT devices.

2) Reward function: The reward function design plays a significant role in the learning process to evaluate the decision-making policy's quality. In the proposed framework, the reward at time t is defined as,

$$r_t = \begin{cases} \sum_i^I \omega \cdot (1 - \mathbf{Pr}\{\delta_i > \xi_i\}), \text{ if constrains} \\ \qquad\qquad\qquad\qquad \text{(10)-(15) are true,} \\ -\sum_i^I (1-\omega) \cdot \mathbf{Pr}\{\delta_i > \xi_i\}), \text{ otherwise.} \end{cases} \tag{18}$$

In (18), we define weight $0.1 \leq \omega \leq 1$ for the reward function and the system receives penalty for the constraint violations. The value of ω close to 1 reduces the penalty for any constraint violation. Therefore, we assume ω is dependent on the factory requirements that is set by the factory manager.

3) Deep deterministic policy gradient with actor-critic model: In step 1 of Alg. 2, the actor and critic networks and target networks with the corresponding network weights are randomly initialized for the DDPG training (lines 1-3, in Alg. 2). The DDPG algorithm uses the replay buffer $\mathcal{M}$ to sample the experience and update the network parameters (line 4, in Alg. 2). In step 2 of Alg. 2, at each episode e of the DDPG training, the association list $\alpha^c_{i,b} \in \tilde{\mathcal{A}} \subset \mathcal{A}$ is generated for allocating the network resources (line 7, in Alg. 1). The adaptive slicing will be done based on the associated IIoT devices' demand at each SBSs. Using the actor-network, the learning agent takes action a_t using the policy function $\mu(s|\theta)$ and executes the action a_t according to the slice action vector (lines 8-9, in Alg. 2) at t. Based on the action a_t, the IIoT association, energy consumption, AoI, and instant reward at t are calculated and observed to move to the next state s_{t+1} (lines 10-11, in Alg. 2). All the states' transitions are stored in the memory buffer and the samples from the memory $\mathcal{M}$ are used to update the actor and critic networks (lines 12-13, in Alg. 2). The value network is updated using the Bellman equation as (line 14, in Alg. 2),

$$y_m = r_m + \gamma\big[Q'(s_{t+1}, \mu'(s_{t+1}|\theta^-)|\psi^-)\big]. \tag{19}$$

In DDPG, the next-state Q values are calculated with the target value network and target policy network where θ^- and ψ^- are the weights of the actor's target network and critic's target network, respectively. Then, we minimize the mean-squared loss between the updated Q value and the original Q value using the loss function as (line 15, in Alg. 2),

$$\mathcal{L} = \frac{1}{|\mathcal{M}|}\sum_{m=1}^{|\mathcal{M}|}(y_m - Q(s_m, a_m|\psi))^2. \tag{20}$$

In (20), ψ is the weight of the critic's network. For the policy function, the objective is to maximize the expected return which is calculated as,

$$J(\theta) = \mathbb{E}\Big[Q(s,a)|_{s=s_t, a_t=\mu(s_t)}\Big]. \tag{21}$$

To calculate the policy loss, the derivative of the objective function is calculated with respect to the policy parameter. Since the actor (policy) function is differentiable, the chain rule is as follows,

$$\nabla_{\theta^\mu} J(\theta) \approx \nabla_a Q(s,a) \nabla_{\theta^\mu} \mu(s|\theta^\mu). \tag{22}$$

In (22), θ is the weight of the actor's network. Since the policy update is off-policy with batches of experience, the mean of the sum of gradients calculated from the mini-batch as (line 16, in Alg. 2),

$$\nabla_\theta J \approx \frac{1}{|\mathcal{M}|}\Bigg[\sum_{m=1}^{|\mathcal{M}|} \nabla_a Q(s,a|\psi)|_{s=s_m, a=\mu(s_m)} \nabla_\theta \mu(s|\theta)|_{s=s_m}\Bigg]. \tag{23}$$

A copy of the target network parameters is used to track the learned networks via *soft updates* (line 17, in Alg. 2). The trained networks are stored and later used for testing the DDPG (line 18, in Alg. 2).

E. Complexity Analysis

The complexity of Alg. 1 is measured by the complexity of creating the distributed preference lists by both SBSs and

Algorithm 2: Deep Deterministic Policy Gradient (DDPG) for RIC non-RT

Step 1: Initialization
Define *actor network*, $\mu(s|\theta)$, and target network, $\mu'(s|\theta^-)$ with corresponding weights θ, θ^-
Define *critic network*, $Q(s,a|\psi)$ and target network $Q'(s,a|\psi^-)$ with corresponding weights ψ, and ψ^-
Set batch size, $\mathcal{M}$
Step 2: DDPG Training
for $e = 1, \cdots, E$ **do**
  Using Alg. 1, generate $\alpha^s_{i,b} \in \tilde{\mathcal{A}} \subset \mathcal{A}$ and capture assignment and slice demands to observe initial state s
  for $t = 1, \cdots, T$ **do**
    Select the action $a_t = \mu(s_t|\theta)$ as per the current policy and execute a_t using (17)
    Calculate average energy consumption, AoI, and instant reward r_t using (4), (8), and (18), respectively
    Observe r_t and next state s_{t+1}
    Store transition (s_t, s_{t+1}, a_t, r_t) in buffer
    Sample $\mathcal{M}$ transitions from buffer
    Compute targets using (19)
    Update critic network using (20)
    Update actor policy using the sampled policy gradient using (23)
    Update target actor and critic networks: $\psi^- \longleftarrow \upsilon\psi + (1-\upsilon)\psi^-, \theta^- \longleftarrow \upsilon\theta + (1-\upsilon)\theta^-$
Store the Q-network

IIoT devices. Using standard sorting algorithm, the complexity of creating the preference lists of SBSs and IIoT devices are $\mathcal{O}(|\mathcal{B}|\log(|\mathcal{B}|))$ and $\mathcal{O}(|\mathcal{I}|\log(|\mathcal{I}|))$, respectively. The worst-case time complexity of Alg. 1 is, $\mathcal{O}(|\mathcal{B}||\mathcal{I}|)$ where $|\mathcal{B}|$ and $|\mathcal{I}|$ are the numbers of SBSs and IIoT devices, respectively. Based on the complexity analysis of the proposed matching game, it can also be inferred that the energy consumption of the IIoT devices is tolerable due to a finite number of iteration per IIoT device preference lists. Meanwhile, in Alg. 2, the model training includes four neural networks with buffer memory for storing the past observations. Also, we apply the state normalization at each training epoch to measure δ, and $\mathcal{E}$, and action a_t is performed at each training time step t. As a result, the time complexity of the training is $\mathcal{O}(s)$ (i.e., dependent on the number of variables in the state space in (16)). Meanwhile, the proposed DDPG training uses the buffer memory to store the past behavior of the IIoT environment, and therefore, the space complexity is $|\mathcal{M}|$.

V. Experimental Results and Analysis

This section first analyzes the experimental environment's performance through the key performance metrics (i.e., energy consumption, AoI, and fairness). We then describe the outcomes obtained from the experiment and finally provide an in-depth discussion and critical observations from the simulation results. To generate our elastic slicing algorithm, we train an RL agent that consists of an actor-critic pair to employ an empirically obtained hyper-parameter. The neural network architectures for both the actor and critic networks are $[512, 512]$ hidden layers, and the rectified linear unit (ReLU) activation function is used for the actor and critic hidden layers to avoid the vanishing gradient problem in backpropagation. The actor and critic networks' learning weights are set to $1e{-}3$ and $\gamma = 0.95$. The number of time steps per iteration is set to 50 where the target network update frequency is set to 5. We also set the buffer size, $\mathcal{M} = 1,000$. The simulation results are obtained by averaging and normalizing the values over 100 episodes. We implemented the proposed architecture and the baseline architectures in python using TensorFlow [36], and Ray RLlib: Scalable Reinforcement Learning [37].

TABLE II: Simulation Settings

Simulation parameters	Value setting
No. of IIoT devices	$[10, 100]$
No. of slice types, % of IIoT devices per type	3, $[30\%, 30\%, 40\%]$
Max. Tx power of IIoT devices	250 mW [38]
No. of SBSs, MEC-BS, k	5, 1, 3
Carrier frequency, bandwidth, number of subchannels, β_{mec}	2.4 GHz, 10 MHz, 32, 100 MHz [39] [38]
Thermal noise density	-174 dBm/Hz
PRB demands, θ_i^{min}	$[1, 5]$
$\alpha_{b,c}^{max}$	10
$\mathcal{E}'$, ξ, ω	0.7, $[2, 4, 6]$, 0.6

A. Experiment Setting

The simulation settings for the performance evaluation is summarized in Table II. The number of IIoT devices belong to three IIoT slice types, as shown in Table I. The percentage of IIoT service type devices for various IIoT devices is 30% for the emergency systems (i.e., Type 1) and scale reading services (i.e., Type 2) and 40% for mobile robots (i.e., Type 3). We adopt the alpha-beta-gamma (ABG) model [40], defined by the 3GPP standardized 5G channel model for the path loss model of the IIoT scenarios. The AoI parameters for Type 1, 2, and 3 IIoT slices are set to $[2, 4, 6]$. Therefore, the Type 1 imposes a stricter AoI threshold (i.e., $\xi_1 = 2$), and the Type 3 has less strict threshold value (i.e., $\xi_3 = 6$). To ensure a fair performance analysis, we compare the proposed DDPG with matching approach with two state-of-the-art approaches [41] which are,

- *Proximal Policy Optimization (PPO):* PPO's clipped objective supports multiple SGD passes over the same batch of experiences. The training batch size, SGD mini-batch size, and the number of SGD iteration are set to $120, 60, 100$, respectively. The learning rate is set to 0.00005. We also shuffle sequences in the batch during model training.
- *Policy Gradient (PG) :* The policy gradient's central idea is that the policy itself is a function with parameters θ, and thus this function can be optimized directly using gradient descent. The learning rate for PG is set to 0.0004.

B. Learning Efficiency Analysis

The learning parameters' selection determines the convergence rate and efficiency of different learning models. In the learning efficiency analysis, we evaluate the performance of

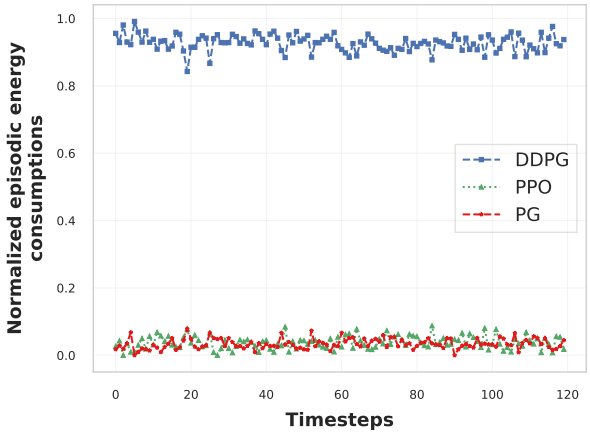

Fig. 3: Evaluating the normalized energy consumption as a function of number of timeslots, $|\mathcal{I}| = 100, |\mathcal{B}| = 5, T = 100$.

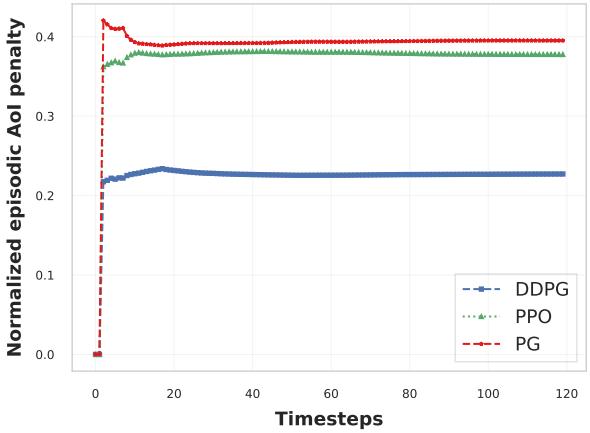

Fig. 4: Evaluating the normalized AoI penalty as a function of number of timeslots, $|\mathcal{I}| = 100, |\mathcal{B}| = 5, T = 100$.

the proposed approach, and other baseline approaches in terms of normalized energy consumption, normalized AoI penalty function, and reward for one episode of training (i.e., episode 100).

In Fig. 3, we observe that the energy consumption of the proposed DDPG is much higher than that of the baseline PPO and PG approaches. The reason behind is that, the proposed DDPG can associate more IIoT devices to the specific slices deployed at the SBSs than the other baseline approaches (explained in Fig. 6). Hence, the energy consumption is 46.10%, 46.5% higher than that of the PPO and PG, respectively. On the other hand, the PPO energy consumption is 2.90% higher than the PG energy consumption. However, the normalized average energy consumption of the IIoT devices in the proposed approach is bounded by the energy consumption threshold. In Fig. 4, the normalized AoI penalty of the PPO and PG are on average 12.56% and 13.52% higher than the proposed DDPG approach. At the beginning of the time step of episode 100, there are no AoI penalty values for all the approaches. However, the AoI penalty values of the PPO and PG approaches increase drastically with increasing time steps (i.e., time step 5) by 12.35% and 50% compare to the proposed DDPG approach. Fig. 5, shows the convergence of all the approaches

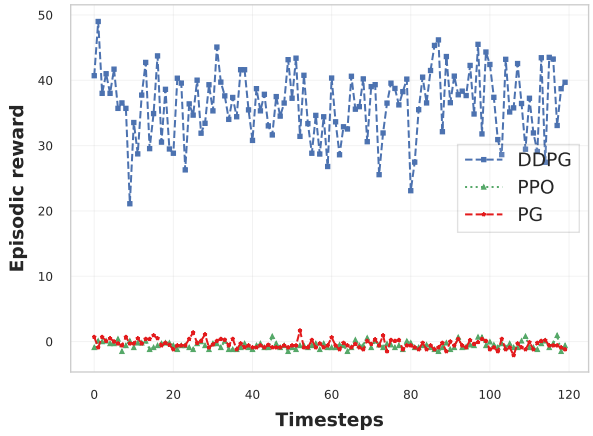

Fig. 5: Evaluating reward as a function of number of timeslots, $|\mathcal{I}| = 100, |\mathcal{B}| = 5, T = 100$.

in terms of episodic reward when the number of IIoT devices is $|\mathcal{I}| = 100$. It can be seen that three of the approaches converge to final levels with different timesteps despite some fluctuations. This is because of the dynamic demand for the different slice types and policy exploration in the network. Among the three approaches, the proposed DDPG achieves a higher convergence rate by returning positive reward at episode 100 whereas the baseline approaches receive higher penalty.

C. Key Performance Analysis

In Fig. 6, the proposed DDPG approach outperforms the baseline approaches significantly to serve different types of IIoT devices. From Fig. 6, we observe that for an increasing number of IIoT devices, the proposed DDPG associate with more diverse IIoT device types than the PPO and PG. However, the percentage of serving different IIoT device types starts reducing after $|\mathcal{I}| = 50$ for a fixed number of SBSs and maximum PRBS in the network. This is because the number of IIoT device associations in the matching game decreases to guarantee stable radio connection management dependent on the channel condition with an increasing number of IIoT devices. The proposed DDPG and the baseline approaches consider such stable matching outcomes to construct the observation space for model training. Therefore, the learning performance efficiencies of those models depend on how well the observations from the experience replay buffer is generalized to learn $Q(s, a)$ function and generate the elastic PRB allocation policy. The baseline approaches exhibit an imbalanced service rate for different IIoT service types due to their observation sample inefficiency than that of the proposed DDPG. As a result, we can infer the proposed DDPG can approximate a more fair and stable radio connection management than the PPO and PG. Overall, the proposed DDPG approach serves an average of 50% and 43.64% more IIoT devices than the PPO and PG, respectively.

In Fig. 7, we evaluate the normalized energy consumption of the proposed DDPG approach compared to the baseline approaches for different IIoT devices. The overall energy consumption of the DDPG is around 50% and 46.50% higher than PPO and PG, respectively. The reason behind that is, with

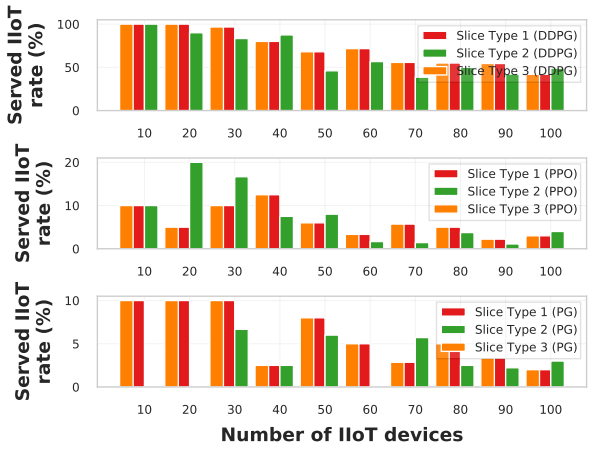

Fig. 6: Service rate comparisons between the proposed DDPG approach and the baseline approaches over different number of IIoT devices, $|\mathcal{B}| = 5$.

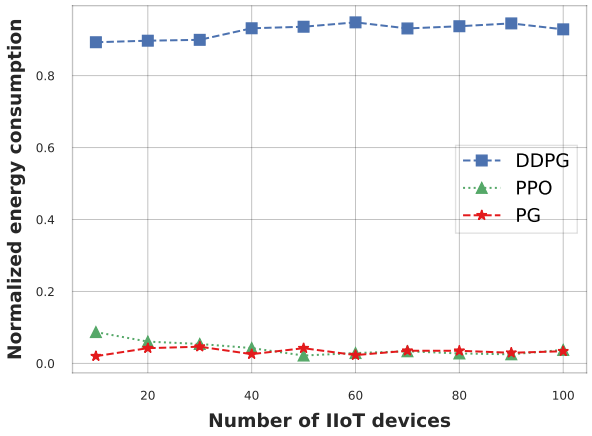

Fig. 7: Normalized energy consumption comparisons between the proposed DDPG approach and the baseline approaches over different number of IIoT devices, $|\mathcal{B}| = 5$.

an increasing number of IIoT devices for a fixed number of SBS, the proposed DDPG approach performs well in terms of serving a higher number of IIoT devices than that of the baseline approaches.

In Fig. 8, the proposed DDPG approach outperforms the baseline approaches significantly in terms of normalized AoI over different number of IIoT devices. With a relatively small number of IIoT devices (i.e., $|\mathcal{I}| = 50$) in the network, the SBSs can support most of the IIoT devices simultaneously. Hence, the IIoT devices are frequently associated with the respective SBSs without violating the AoI threshold. As the network size grows (i.e., $|\mathcal{I}| = 60$), the SBSs can associate and then allocate PRBs to less IIoT slice type devices, and the AoI penalty increases for the proposed DDPG. On the other hand, the AoI penalty of the PG and PPO fluctuates with an increasing number of IIoT devices. However, the AoI penalties of IIoT devices for the DDPG are still less than the PPO and PG, respectively. In Fig. 9, the proposed DDPG approach outperforms the baseline approaches significantly in terms of cumulative rewards over the different number of IIoT devices. In the case of the proposed DDPG approach, the rate of change in cumulative reward is around 12.83% up to $|\mathcal{I}| = 40$ compare to 2.33% and 17.76% of the PPO and PG, respectively. When the network size is between $|\mathcal{I}| = [50, 80]$,

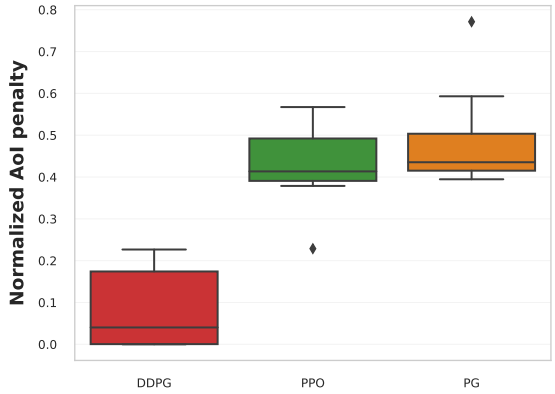

Fig. 8: Normalized AoI penalty comparisons between the proposed DDPG approach and the baseline approaches over different number of IIoT devices.

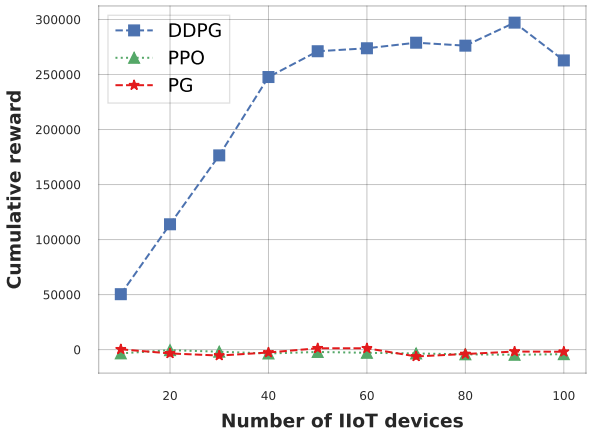

Fig. 9: Cumulative comparisons between the proposed DDPG approach and the baseline approaches over different number of IIoT devices, $|\mathcal{I}| = [10, 100], |\mathcal{B}| = 5$.

the cumulative change rate stabilizes in the case of all the approaches. However, when the network is significantly large (i.e., $|\mathcal{I}| = 100$), the DDPG approach's performance drops slightly due to the increasing discrepancy between the increasing amount of demand and static PRB supply. This is because the QoS of the IIoT applications degrades after a certain point for a fixed number of SBSs and maximum PRBs. It is noteworthy to mention, the higher cumulative reward under different network settings indicates the proposed DDPG strikes a better trade-off between minimizing the AoI violation and maximizing the energy efficiency of the IIoT devices than the other two baseline approaches. Overall, the proposed DDPG shows 51.39% and 51.01% higher performance gain to achieve better long-term reward compared to the PPO and PG, respectively.

In Fig. 10, we evaluate the running time performance of the proposed DDPG approach with the baseline approaches under different IIoT network sizes. The proposed DDPG approach requires 26.15% and 4.48% less running time during training than PPO and PG, respectively. This is because the proposed DDPG approach takes advantage of the IIoT service types' historical demand to make future decisions, whereas the baseline approaches are less efficient in using the buffer.

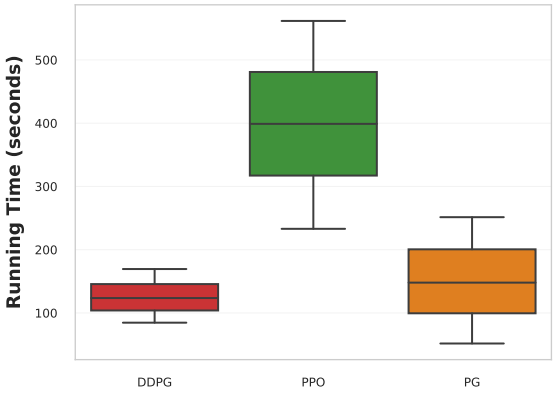

Fig. 10: Mean running time comparisons between the proposed DDPG approach and the baseline approaches over different number of IIoT devices, , $|\mathcal{I}| = [10, 100], |\mathcal{B}| = 5$.

D. Discussion

Based on the experimental analysis, we can summarize the key observations to prove the efficacy of the proposed actor-critic model with DDPG than that of the baseline approaches, which are as below,

- Unlike the PG and PPO, the proposed DDPG more beneficial to compute the policy gradient based on learned value functions rather than raw observed rewards and returns. Therefore, this approach reduces the noise and increases the algorithm's robustness because the learned Q-function can generalize the observed experiences.
- The proposed DDPG combined with the actor-critic incorporates the stabilization techniques introduced in DQN with the experience replay buffer and target networks that allow for complex neural approximators. Most importantly, the DDPG with actor-critic considers a deterministic policy $\pi(s)$, while the PG considers stochastic policies that specify probabilistic distributions over actions $\pi(a|s)$. As a result, the deterministic policy approach has significant performance gains and is generally more sample-efficient because the policy gradient integrates only over the state space but not the action space.

VI. Conclusion

In this work, we address solving an elastic O-RAN slicing problem for industrial monitoring and control in IIoT. We reduce the AoI penalty cost of fresh information updates from different IIoT devices while considering the energy consumption of the IIoT devices. We also incorporated different industrial communication environment aspects and maintained the O-RAN slice's quality of service (QoS). As a result, we introduce a matching game for solving the IIoT association problem and then applied an actor-critic-based deep reinforcement learning model for O-RAN slicing-based resource allocation. We also performed comprehensive simulation analysis to show the proposed approach's efficacy to achieve significant performance gain concerning the essential performance indicators. The simulation results show a strong correlation between the IIoT service rate, energy consumption, and AoI, wherein the proposed approach ensures fairness in achieving a balanced and elastic O-RAN slicing policy. The critical observations in the simulation further verify the proposed approach's effectiveness under different network settings.

References

[1] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca, and J. Folgueira, "Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 80–87, 2017.

[2] F. Song, J. Li, C. Ma, Y. Zhang, L. Shi, and D. Nalin, "Dynamic virtual resource allocation for 5G and beyond network slicing," *IEEE Open Journal of Vehicular Technology*, 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] L. Gavrilovska, V. Rakovic, and D. Denkovski, "From Cloud RAN to Open RAN," *Wireless Personal Communications*, pp. 1–17, 2020.

[5] 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.

[6] S. Messaoud, A. Bradai, and E. Moulay, "Online GMM clustering and mini-batch gradient descent based optimization for industrial IoT 4.0," *IEEE Trans. Ind. Informat.*, vol. 16, no. 2, pp. 1427–1435, 2019.

[7] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial internet of things: Challenges, opportunities, and directions," *IEEE Trans. Ind. Informat.*, vol. 14, no. 11, pp. 4724–4734, 2018.

[8] F. Barac, S. Caiola, M. Gidlund, E. Sisinni, and T. Zhang, "Channel diagnostics for wireless sensor networks in harsh industrial environments," *IEEE Sensors J.*, vol. 14, no. 11, pp. 3983–3995, 2014.

[9] S. Zhang, H. Zhang, Z. Han, H. V. Poor, and L. Song, "Age of information in a cellular internet of UAVs: Sensing and communication trade-off design," *IEEE Trans. Wireless Commun.*, vol. 19, no. 10, pp. 6578–6592, 2020.

[10] L. Bonati, S. D'Oro, M. Polese, S. Basagni, and T. Melodia, "Intelligence and learning in O-RAN for data-driven NextG cellular networks," *arXiv preprint arXiv:2012.01263*, 2020.

[11] J. Kwak, J. Moon, H.-W. Lee, and L. B. Le, "Dynamic network slicing and resource allocation for heterogeneous wireless services," in *IEEE PIMRC*, 2017, pp. 1–5.

[12] Y. K. Tun, M. Alsenwi, N. H. Tran, Z. Han, C. S. Hong *et al.*, "Energy efficient communication and computation resource slicing for eMBB and URLLC coexistence in 5G and beyond," *IEEE Access*, vol. 8, pp. 136 024–136 035, 2020.

[13] Y. Li, A. Huang, Y. Xiao, X. Ge, S. Sun, and H.-C. Chao, "Federated orchestration for network slicing of bandwidth and computational resource," *arXiv preprint arXiv:2002.02451*, 2020.

[14] J. Thota and A. Aijaz, "Slicing-enabled private 4G/5G network for industrial wireless applications," in *Proc. of the 26th Annual International Conference on Mobile Computing and Networking*, 2020, pp. 1–3.

[15] S. Messaoud, A. Bradai, O. B. Ahmed, P. Quang, M. Atri, and M. S. Hossain, "Deep federated Q-learning-based network slicing for industrial IoT," *IEEE Trans. Ind. Informat.*, 2020.

[16] H. H. Yang, C. Xu, X. Wang, D. Feng, and T. Q. Quek, "Understanding age of information in large-scale wireless networks," *IEEE Trans. Wireless Commun.*, 2021.

[17] S. F. Abedin, M. G. R. Alam, S. A. Kazmi, N. H. Tran, D. Niyato, and C. S. Hong, "Resource allocation for ultra-reliable and enhanced mobile broadband IoT applications in fog network," *IEEE Trans. Commun.*, vol. 67, no. 1, pp. 489–502, 2018.

[18] H. Wu, I. A. Tsokalo, D. Kuss, H. Salah, L. Pingel, and F. H. Fitzek, "Demonstration of network slicing for flexible conditional monitoring in industrial IoT networks," in *IEEE CCNC*. IEEE, 2019, pp. 1–2.

[19] H. Yang, A. Alphones, W.-D. Zhong, C. Chen, and X. Xie, "Learning-based energy-efficient resource management by heterogeneous RF/VLC for ultra-reliable low-latency industrial IoT networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5565–5576, 2019.

[20] P. Yang, X. Xi, T. Q. Quek, J. Chen, X. Cao, and D. Wu, "RAN slicing for massive IoT and bursty URLLC service multiplexing: Analysis and optimization," *arXiv preprint arXiv:2001.04161*, 2020.

[21] S. Dawaliby, A. Bradai, and Y. Pousset, "Distributed network slicing in large scale IoT based on coalitional multi-game theory," *IEEE Trans. Netw. Service Manag.*, vol. 16, no. 4, pp. 1567–1580, 2019.

[22] Q. Wang, H. Chen, Y. Li, Z. Pang, and B. Vucetic, "Minimizing age of information for real-time monitoring in resource-constrained industrial IoT networks," in *IEEE INDIN*, vol. 1. IEEE, 2019, pp. 1766–1771.

[23] Y.-P. Hsu, E. Modiano, and L. Duan, "Scheduling algorithms for minimizing age of information in wireless broadcast networks with random arrivals," *IEEE Trans. Mobile Comput.*, 2019.

[24] S. F. Abedin, M. S. Munir, N. H. Tran, Z. Han *et al.*, "Data freshness and energy-efficient UAV navigation optimization: A deep reinforcement learning approach," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–13, 2020.

[25] Q. Guo, R. Gu, Z. Wang, T. Zhao, Y. Ji, J. Kong *et al.*, "Proactive dynamic network slicing with deep learning based short-term traffic prediction for 5G transport network," in *IEEE OFC*, 2019, pp. 1–3.

[26] Y. Ren, Y. Sun, and M. Peng, "Deep reinforcement learning based computation offloading in fog enabled industrial internet of things," *IEEE Trans. Ind. Informat.*, 2020.

[27] A. Bombino, S. Grimaldi, A. Mahmood, and M. Gidlund, "Machine learning-aided classification of LoS/NLoS radio links in industrial IoT," in *IEEE WFCS*, 2020, pp. 1–8.

[28] Y. d. J. Bultitude and T. Rautiainen, "IST-4-027756 WINNER II D1. 1.2 V1. 2 WINNER II channel models," *EBITG, TUI, UOULU, CU/CRC, NOKIA, Tech. Rep., Tech. Rep*, 2007.

[29] S. F. Abedin, A. K. Bairagi, M. Munir, N. H. Tran, and C. Hong, "Fog load balancing for massive machine type communications: A game and transport theoretic approach," *IEEE Access*, vol. 7, pp. 4204–4218, 2019.

[30] K. Nag and M. Helal, "Evaluating Erlang C and Erlang A models for staff optimization: A case study in an airline call center," in *IEEE IEEM*, 2017, pp. 1–5.

[31] Y. Sun *et al.*, "Update or wait: How to keep your data fresh," *IEEE Trans. Inf. Theory*, vol. 63, no. 11, pp. 7492–7508, 2017.

[32] N. I. Mowla, N. H. Tran, I. Doh, and K. Chae, "Federated learning-based cognitive detection of jamming attack in flying ad-hoc network," *IEEE Access*, vol. 8, pp. 4338–4350, 2019.

[33] ORAN Alliance, "O-RAN: Towards an Open and Smart RAN," https://www.o-ran.org/resources, 2018, [Online; accessed 09-03-2021].

[34] R. W. Irving *et al.*, "The hospitals/residents problem with ties," in *Scandinavian Workshop on Algorithm Theory*. Springer, 2000, pp. 259–271.

[35] Y. Gu, W. Saad, M. Bennis, M. Debbah, and Z. Han, "Matching theory for future wireless networks: Fundamentals and applications," *IEEE Commun. Mag.*, vol. 53, no. 5, pp. 52–59, 2015.

[36] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, 2016, pp. 265–283.

[37] E. Liang *et al.*, "Ray RLlib: A composable and scalable reinforcement learning library," *arXiv preprint arXiv:1712.09381*, p. 85, 2017.

[38] H. Yang, A. Alphones, W. Zhong, C. Chen, and X. Xie, "Learning-based energy-efficient resource management by heterogeneous RF/VLC for ultra-reliable low-latency industrial IoT networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5565–5576, 2020.

[39] C.-C. Lai *et al.*, "Data-driven 3D placement of UAV base stations for arbitrarily distributed crowds," in *IEEE GLOBECOM*, 2019, pp. 1–6.

[40] T. Jiang *et al.*, "3GPP standardized 5G channel model for IIoT scenarios: A survey," *IEEE Internet Things J.*, pp. 1–1, 2021.

[41] J. Schulman, F. Wolski, P. Dhariwal, A. Radford *et al.*, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

Sarder Fakhrul Abedin (S'18-M'20) received his B.S. degree in Computer Science from Kristianstad University, Sweden, in 2013. He received his Ph.D. degree in Computer Engineering from Kyung Hee University, South Korea in 2020. Currently, he is serving as an Assistant Professor at Department of IST, Mid Sweden University, Sweden. His research interests include edge computing, machine learning, Industrial wireless networks.

Aamir Mahmood (M'18-SM'19) received the B.E. degree in electrical engineering from NUST, Pakistan, in 2002, and the M.Sc. and D.Sc. degrees in communications engineering from the School of Electrical Engineering, Aalto University, Finland, in 2008 and 2014. He worked as a Research Intern with the Nokia Research Center, Finland, in 2014, a Visiting Researcher with Aalto University, and a Post-Doctoral Researcher with Mid Sweden University, Sweden, from 2015 to 2018, where he has been an Assistant Professor since 2019. His research interests include radio resource allocation, and RF coexistence management.

Nguyen H. Tran (S'10-M'11) received the BS degree from Hochiminh City University of Technology and Ph.D. degree from Kyung Hee University, in electrical and computer engineering, in 2005 and 2011, respectively. Since 2018, he has been with the School of Computer Science, The University of Sydney, where he is currently a Senior Lecturer. He was an Assistant Professor with Department of Computer Science and Engineering, Kyung Hee University, Korea from 2012 to 2017. His research interest is to applying analytic techniques of optimization, game theory, and stochastic modeling to cutting-edge applications such as cloud and mobile edge computing, data centers, heterogeneous wireless networks, and big data for networks. He received the best KHU thesis award in engineering in 2011 and best paper award at IEEE ICC 2016. He has been the Editor of IEEE Transactions on Green Communications and Networking since 2016, and served as the Editor of the 2017 Newsletter of Technical Committee on Cognitive Networks on Internet of Things.

Zhu Han (S'01-M'04-SM'09-F'14) received the B.S. degree in electronic engineering from Tsinghua University, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, in 1999 and 2003, respectively. From 2000 to 2002, he was an RD Engineer of JDSU, Germantown, Maryland. From 2003 to 2006, he was a Research Associate at the University of Maryland. From 2006 to 2008, he was an assistant professor at Boise State University, Idaho. Currently, he is a John and Rebecca Moores Professor in the Electrical and Computer Engineering Department as well as in the Computer Science Department at the University of Houston, Texas. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. Dr. Han received an NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the EURASIP Best Paper Award for the Journal on Advances in Signal Processing in 2015, IEEE Leonard G. Abraham Prize in the field of Communications Systems (best paper award in IEEE JSAC) in 2016, and several best paper awards in IEEE conferences. Currently, Dr. Han is IEEE fellow since 2014, AAAS fellow since 2019 and ACM distinguished member since 2019. Dr. Han is 1% highly cited researchers according to Web of Science since 2017.

Mikael Gidlund (M'98-SM'16) received the Licentiate of Engineering degree in radio communication systems from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2004, and the Ph.D. degree in electrical engineering from Mid Sweden University, Sundsvall, Sweden, in 2005. From 2008 to 2015, he was a Senior Principal Scientist and a Global Research Area Coordinator of Wireless Technologies with ABB Corporate Research, Västerås, Sweden. From 2007 to 2008, he was a Project Manager and a Senior Specialist with Nera Networks AS, Bergen, Norway. From 2006 to 2007, he was a Research Engineer and a Project Manager with Acreo AB, Hudiksvall, Sweden. Since 2015, he has been a Professor of computer engineering with Mid Sweden University, Sundsvall, Sweden. He holds more than 20 patents (granted and pending applications) in the area of wireless communication. His current research interests include wireless communication and networks, wireless sensor networks, access protocols, and security. Dr. Gidlund is an Associate Editor of the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS.