I-HARF: Intelligent and Hierarchical Framework for Adaptive Resource Facilitation in Edge-IoT Systems

Ismail AlQerm and Jianli Pan
Department of Computer Science, University of Missouri-St. Louis, St. Louis, MO, USA 63121
E-mail: alqermi, pan@umsl.edu

***Abstract*—Edge computing is being used to facilitate closer computing, storage and networking resources to support various IoT applications including delay-sensitive ones. It is envisioned that the future Edge-IoT systems will incorporate heterogeneous IoT devices distributed over multiple geographical zones of certain institution with edge resource demands that vary according to time and location. Edge servers (resource facilitators) are with limited resources and are susceptible to "outlandish" situations such as service overloading, outage, and external attacks; they may also have to handle the roaming of IoT devices among different zones. These situations induce the need for alternative edge servers using an adaptive resource facilitation scheme to fulfill the demands of the IoT applications. In this paper, we develop a novel intelligent and hierarchical resource facilitation framework named *I-HARF* that adapts to dynamic Edge-IoT situations including outlandish situations, mobility, application's sensitivity and varying resource demand of IoT applications based on time and location. *I-HARF* achieves an adaptive facilitation and holistically addresses the facilitation technical barriers by: 1) adopting hierarchical structure which efficiently migrates the resource facilitation from intra-zone to inter-zone levels; 2) extending novel intra-zone and inter-zone optimization models to boost the utilities of the edge servers and the IoT applications; and 3) developing a novel and unique actor dual-critic and collective actor-critic Deep Reinforcement Learning (DRL) designs that intelligently facilitate the edge resources in both intra-zone and inter-zone respectively. The evaluation results demonstrate *I-HARF*'s capability enabling adaptive resource facilitation that adjusts according to the dynamic Edge-IoT situations.**

***Index Terms*—Edge computing; Internet of Things; Intra-zone facilitation; Inter-zone facilitation; Heterogeneous IoT devices; Deep reinforcement learning.**

I. Introduction

Traditional edge facilities such as 4G/5G base stations, Wi-Fi access points, and wireline central offices are being revamped into small data centers or "edge cloud" to facilitate closer computing, storage and networking resources to support various IoT applications that can be delay-sensitive, bandwidth/data intensive, or require closer resources for machine intelligence. These heterogeneous Edge-IoT systems will shape the future of our daily life, work and productivity, as envisioned by the NSF "10 Big Ideas" [1]. The Edge-IoT environment is expected to connect and provide edge resources (CPU, storage and bandwidth) for heterogeneous IoT applications with various QoS requirements and priorities. These applications run on IoT devices which are distributed over multiple geographical zones of certain institution such as university campus, corporation premises, and residential property.

The resource facilitation at the edge must be adaptive to handle dynamic situations from both IoT and edge sides. From IoT side, mobility of IoT devices is considered as a recurrent problem for resource facilitation as some of the devices such as smart vehicles may roam from one zone to another around the institution. In addition, dynamic zone situations might exist such as more connected devices, various application types with different QoS requirements and priorities, variation of resource demand according to time and location, optimization objectives, and multiple types of resources. From the edge side, edge resource providers (servers) might experience outlandish situations such as server malfunction due to maintenance problem, server overloading because of instant surge in the resource demands from the IoT devices, and hacking of vulnerable server owing to weak security. Technically, the edge resource facilitator must: 1) adapt to the dynamic zone situations in its intra-zone facilitation and fulfill the application demands with QoS guarantee; 2) adapt to the edge dynamic situations and devices mobility through transition from the intra-zone facilitation to inter-zone facilitation in which edge facilitators coordinate with neighboring edge zones to scale up or down resource facilitation; 3) support sensitive applications such as emergency response that cannot tolerate extra latency in the inter-zone resource facilitation given the outlandish situations.

Most of existing works either focus on "cloud-edge" integration or involve multiple edge servers for offloading [2], [3], [4], [5], [6]. Most of the current work on resource facilitation across multiple edge zones did not jointly consider edge outlandishness and device roaming. Moreover, the current intra-zone edge resource facilitation research either focuses on a specific application, or optimizes specific operations such as mobile offloading, migration, and orchestration from latency or energy efficiency perspectives [7], [8], [9], [10]. None of the existing schemes address resource facilitation in the intra-zone by considering multi-dimensional factors including application priorities, dynamic demands for different resources, and heterogeneous QoS requirements for various IoT systems. Advanced designs for resource facilitation are needed to coordinate and jointly optimize intra-zone and inter-zone resource facilitation.

In this paper, we develop a novel intelligent and hierarchical resource facilitation framework named *I-HARF* that adapts to dynamic Edge-IoT situations including outlandish situations, mobility, application's sensitivity and varying resource demand of heterogeneous applications based on time and location. To holistically address the above technical challenges, and align with *I-HARF* goals, we:

- Develop an adaptive intra-zone resource facilitation scheme that includes: 1) An intra-zone facilitation model that aims to boost the edge and IoT application utilities and maximize the edge server functionality by incorporating dynamic application priority assignment according to the current application data volume; and 2) a novel off-policy actor dual-critic policy gradient (ADCPG) DRL scheme, which tackles the dynamic situations in the intra-zone facilitation by leveraging an additional specialized critic network. This critic will help training the actor to generate better resource facilitation actions to achieve the intra-zone objectives and overcome the stability and slowness problem of the classic actor-critic in such context.
- Develop an inter-zone resource facilitation scheme that tackles the possible edge outlandish situations and mobility of the devices. It incorporates: 1) an inter-zone facilitation model which aims to migrate the resource facilitation from the intra-zone to the inter-zone level by coordinating between different zone facilitators and optimizing intra-zone goals under edge outlandish and devices roaming situations; and 2) a novel collective actor-critic policy gradient (CACPG) DRL scheme to solve the inter-zone model, which integrates with the intra-zone DRL to support learning of the inter-zone resource facilitation policies.
- Design a resource reservation mechanism for sensitive applications that exploits prediction of resource demands of these applications to fulfill their delay requirements as they cannot tolerate the delay encountered due to the coordination between the resource facilitators.

The rest of the paper is organized as follows. Section II describes the motivation of *I-HARF* and a typical resource facilitation environment example with the possible resource facilitation scenarios. The related work, its limitations and the remedy using *I-HARF* are presented in Section III. Section IV illustrates the architecture of *I-HARF* and the proposed resource facilitation schemes of *I-HARF* for each scenario including system models and DRL techniques. The performance evaluation is shown in Section V and the paper concludes in Section VI.

II. Motivation and Facilitation Environment

In this section, we present the motivation for the intelligent and hierarchical design of *I-HARF* for adaptive resource facilitation in Edge-IoT. Moreover, we provide a typical example of Edge-IoT environment where the presence of *I-HARF* is vital.

A. Motivation of I-HARF

Inter-zone facilitation and through coordination between edge resource facilitators is necessary to achieve adaptive resource facilitation which is not a trivial task with multiple edge facilitators involved. Efficient intra-zone facilitation is a key factor that is required to achieve an efficient inter-zone facilitation given the dynamic devices' activities and the heterogeneous resource demands of their applications. Therefore, we adopt hierarchical resource facilitation in which the lower-level "intra-zone" facilitation aims to facilitate the resources from the intra-zone edge servers. The latest environment situation is captured and adaptive decisions are made to assign applications priorities and allocate multi-dimensional edge resources to maximize edge servers and applications utilities. The upper level "inter-zone" adopts a collective facilitation approach and follow a "distributed-to-centralized" cooperative pattern which maintains the edge and the IoT application utilities given the multiple resource facilitators involved. To support sensitive applications such as emergency response which are delay sensitive and cannot tolerate the extra latency encountered due to coordination in the inter-zone facilitation, *I-HARF* extends its inter-zone facilitation and reserves resources for these sensitive applications to provide real-time response. Due to lack of consistent models, it is difficult to optimize the resource facilitation decision-making for the highly heterogeneous and dynamic Edge-IoT environment. Thus, model-free DRL [19] is considered as a good candidate because it learns and improves itself from experience and does not require prior knowledge of the system's behavior. We exploit an actor-critic [47] DRL framework in order to leverage the simplicity of the value-based DRL and the ability to handle continuous action space of the policy-based DRL.

B. Smart Campus: A Typical Example of Resource Facilitation Environment

Smart campus is a typical future Edge-IoT environment where various heterogeneous applications exist and span over multiple zones. Each zone including academic, community, and recreation are composed of a building or a complex of buildings and their outdoor area. The envisioned IoT applications for smart campus as in Table I are distinct in terms of their QoS requirements, mobility, priority, device type, and resource demands. The application's resource demand varies according to the time and location context. For example, resource requests are expected to be intensive during the working hours in comparison to the evening time. The volume of resource requests becomes extraordinary at certain campus event such as conference or sport games. Moreover, the location has a key role that impacts the resource demands. For instance, academic zone is anticipated to have more resource requests in contrast with the community zone. Fig. 1 presents an abstract of smart campus that shows the zones, their buildings, and the typical applications in each zone.

There are multiple resource facilitation scenarios to tackle in smart campus based on the environment variations and possible occasions. 1) Typical Scenario: this is the case when the edge facilitators are in healthy condition and able to fulfill the requested resources for the stationary IoT devices within their zones. In this scenario, application priority is an essential factor that affects the resource facilitation process. For example, smart transportation has significant volume of data that requires processing at the edge due to a sport event at the stadium and large number of attendees. However, the face recognition application running on the academic campus is supposed to have lower priority in this occasion. It is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3151667, IEEE Internet of Things Journal

TABLE I
EXAMPLES OF IOT APPLICATIONS IN SMART CAMPUS.

Application	Location/Zone	Priority	Mobility	Device Type	Sensitivity
Face Recognition	Academic Buildings, Recreation	Dynamic	Sationary	Camera	Medium
Smart Transportation and Delivery	Around campus outdoor	Dynamic	Mobile	Motion and obstacle detection	High
Emergency Response (gunshot, fire)	Around Campus indoor and outdoor	High	Stationary	Camera	High
Education and Research Services (e.g. AI apps, VR apps)	Academic Buildings, Community	Dynamic	Stationary/ Mobile	Multiple	Medium/ Low

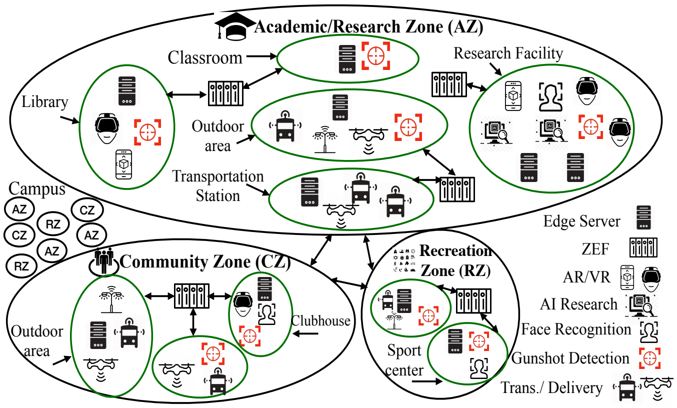


Fig. 1. Smart Campus Zones.

inefficient to assign a fixed priority for these applications with varying volume of resource requests. 2) Complicated Scenario: the edge servers at certain zone may not be capable to fulfill the resource demands and require the intervention of the upper-level of facilitation or seek the neighbor servers support. This inability can be due to: a) the edge server is overloaded because of specific event at certain time and location such as academic conference or sport event in which the IoT applications generate large volume of data that requires processing; b) the server is hacked or experience malfunction as a result of certain maintenance issue; c) mobility of the IoT devices which is the case in the smart transportation and delivery application. It creates the need for inter-zone resource facilitation as the devices (vehicles) may hop from one zone to the other. 3) Exceptional Scenario: this scenario is typically a complicated scenario with sensitive IoT applications. The sensitive applications such as emergency response require extra measures to avoid additional latency. The second and third scenarios induce the inter-zone resource facilitation.

III. RELATED WORK, ITS LIMITATIONS AND REMEDY USING *I-HARF*

In this section, we present the related work, its limitations and how *I-HARF* tackles these limitations.

A. *Related Work*

Many edge computing research have focused on specific operations such as offloading, migration, chaining and orchestration [7] [8] [9] [10], and among others, delay and energy efficiency have been the key optimization goals [11] [12] [13] [14]. Various game theory based, bio-inspired, or economic pricing based optimization methods have been tried [15] [16][17] [18]. Deep Reinforcement Learning (DRL) [19] has attracted attention for small-scope problems such as edge offloading and management [20] [21] [22] [23] [24] [25], with edge content caching in smart vehicle as a typical application [26] [27]. Value-based DRL schemes such as Deep Q-network (DQN) were explored for resource management problem in the edge computing context [28] [29] [30] [31] [32] [33]. In [31], authors exploited edge load dynamics, and formulated a task offloading problem to minimize the expected long-term cost using model-free DQN. The work in [33] proposed an improved deep Q-network (DQN) algorithm to learn the policy of edge resource allocation, where multiple replay memories are applied to separately store the experiences with small mutual influence. Basic actor-critic based DRL methods were recently explored for edge offloading and resource allocation [34] [35] [36] [37]. In particular, the work [41] aimed to improve the typical deep deterministic policy gradient (DDPG) method [42] for edge offloading by considering a critic with double TD neural networks for better value function estimation. The Work in [35] targets delay and energy performance in a multi-user system with an online solution using actor-critic DRL to deal with time-varying user requests and channel conditions. The authors in [39] proposed a DDPG-based algorithm to solve the edge resource management problem using two architectures in vehicular Network context. Inter-zone resource management in Edge-IoT was tackled in a narrow range with very few proposals. The work in [43] proposed a load balancer that accounts for the dynamicity and limitation of edge clouds. In [44], the authors designed an *AutoScaler* to handle offloading in large-scale IoT deployments. Hierarchical "cloud-edge" structure have been explored [45], [46] in which the cloud acts as a supportive resource backup.

B. *Related Work Limitation and I-HARF's Remedy*

Given the related works, they neither addressed the need for inter-zone facilitation in their designs nor built hierarchical model that takes both intra-zone edge resource facilitation and possible "outlandish" and roaming conditions in the inter-zone facilitation into consideration, while both conditions can be the norms for future Edge-IoT. The current work focuses on resource facilitation at the intra-zone level. However, it did not account for the dynamicity of resource demands from the IoT application in different context. The lower-level facilitation of the hierarchical design of *I-HARF* adopts a dynamic application priority concept in which the priority is determined in real-time according to application data volume and its QoS requirements. Dynamic priority accounts for the impact of dynamic resource demands of the IoT applications occurred due to variation in data volume according to time and location contexts. We formulate an intra-zone optimization model characterizing environment situations such as the application types, QoS requirements and resource demands, and incorporating dynamic priority.

The basic DRL schemes such as DQN and actor-critic used in most of the existing schemes do not fit the dimensionality of the considered resource facilitation problem as it involves multiple objectives, multiple resources, various QoS requirements, and heterogeneous applications. None of the existing schemes that use the classic actor-critic methods addressed their potential slowness and stability issues encountered while solving such resource facilitation problem. Thus, we develop an off-policy actor dual-critic policy gradient (ADCPG) DRL that integrates an additional critic network aiming to help criticizing and training the actor. It generates better resource facilitation actions and overcome the stability problem of the classic actor-critic. The additional critic in ADCPG provides an additional loss function that improves the classic actor-critic generated action given the dynamic Edge-IoT environment and achieve better sampling efficiency, faster convergence, and more stabilized performance. The design is generic and can be used with any off-policy DRL [42], [48], [49].

In addition, the current work did not exploit the intra-zone facilitation as a building block for the inter-zone facilitation and treat them as separate problems despite the fact that the inter-zone facilitation relies on the current resource facilitation in the zones. We introduce an inter-zone optimization model that incorporates the utilities optimization of the intra-zone model and coordination costs introduced at the inter-zone level. For the upper-level DRL, we develop a CACPG DRL that exploits the intra-zone resource facilitation action policies as seeds to train the upper-level critic network. Moreover, the value function objective is crafted based on the inter-zone facilitation model optimization objective. This distinguishes our scheme from the federated learning [50] based schemes that rely on handcrafted features to characterize Edge-IoT conditions and simply aggregate individual value function of each intra-zone policy.

IV. Resource Facilitation using *I-HARF*

In this section, we present the architecture of *I-HARF*, a glimpse of *I-HARF* adaptation mechanism that adapts to the facilitation scenarios in Section II.B, and the facilitation schemes of *I-HARF*. The facilitation schemes include: the intra-zone resource facilitation for the typical scenario, the inter-zone resource facilitation for the complicated scenario and the inter-zone resource facilitation with the resource reservation for the exceptional scenario.

A. Architecture and Adaptation Mechanism of I-HARF

I-HARF's architecture is presented in Fig. 2. The key components include: 1) the IoT environment; 2) zone edge facilitators (ZEFs) and edge servers; and 3) the institution edge facilitator (IEF). The IoT environment is complex due to high heterogeneity and dynamicity of devices and applications. The ZEFs act as managers for their associated edge servers and run the DRL-based function facilitating resources on the edge servers for the IoT applications. ZEFs comprise the intra-zone optimization model and lower-level DRL module. The intra-zone optimization model is solved by the lower-level DRL module using the proposed novel ADCPG approach, which consists of the key components including O-Critic, S-Critic and Actor that work together to achieve the ADCPG design goals. The intra-zone comprehensive MDP (CMDP) used by the ADCPG incorporates the intra-zone situation information represented by the static space of applications QoS and the stochastic space of the dynamic resource demands. The IEF

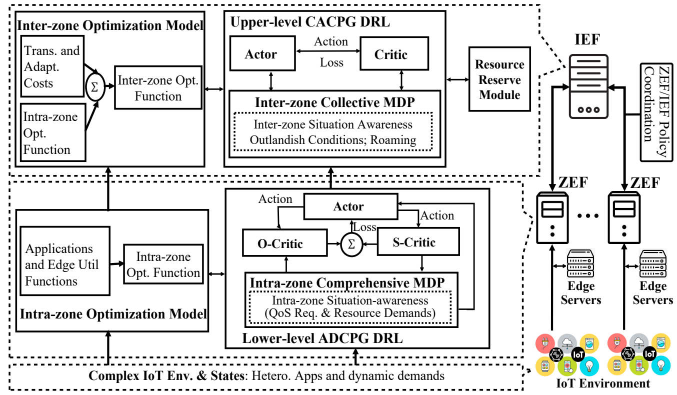


Fig. 2. *I-HARF* Architecture

incorporates the inter-zone optimization model, upper-level DRL module, and resource reservation modules. The inter-zone optimization model is solved by the upper-level DRL module that is based on a collective actor-critic DRL operating in conjunction with the inter-zone MDP. The inter-zone MDP includes the aggregated intra-zone CMDP in addition to the state outlandish conditions to characterize the inter-zone situation-awareness. The resource reservation module acts when the inter-zone facilitation engages to reserve the resources for the sensitive IoT applications.

Fig. 3 maps the functions of *I-HARF* to the scenarios of the smart campus and their respective applications. *I-HARF*

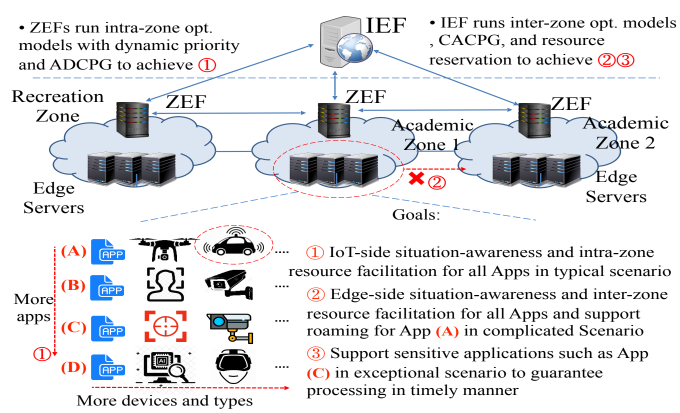


Fig. 3. Intelligent, hierarchical, and adaptive facilitation mechanism.

accounts for the three resource facilitation scenarios in smart campus as follows. 1) For the typical scenario: ZEFs conceive IoT situation information at their respective zones and run the intra-zone optimization models and ADCPG scheme to achieve the intra-zone resource facilitation for all IoT applications. 2) For the complicated scenario: IEF uses edge situation information from its associated ZEFs and run the inter-zone optimization model and CACPG scheme to achieve the inter-zone resource facilitation for all IoT applications and support mobility for smart transportation. 3) For the exceptional scenario: IEF runs the resource reservation mechanism along with CACPG to support sensitive applications such as emergency response such that it guarantees their data processing in timely manner without disruption. The integration between the inter-zone and intra-zone facilitation is triggered if the resource request of the IoT application cannot be handled at the ZEF due to the IoT device mobility or an outlandish situation. For mobility, it is assumed that each IoT device reports its location when it makes the resource request. The corresponding ZEF asks for IEF intervention for resource facilitation using inter-zone facilitation if it finds that the device is moving out of its proximity. For outlandish situations, the ZEF asks for support of the IEF if it is unable to process request in timely manner due its limited resource capacity. This is verified by comparing the application load against its capacity before facilitation. If the IoT device did not receive any response from its corresponding ZEF for certain time which indicates that the ZEF is down, it sends its request to the IEF through the gateway, thus IEF can intervene and find a replacement ZEF. The IEF only coordinates the facilitation of special outlandish and roaming occasions among the affected ZEFs for a limited period of time and do not replace the role of individual ZEFs in resource facilitation. After the special situation or transition is accomplished, the assigned ZEFs will handle the affected devices. Thus, the IEF will not become a new single performance bottleneck.

B. Intelligent Intra-zone Resource Facilitation Scheme

In this subsection, we illustrate the intra-zone facilitation scheme which is customized to handle the typical facilitation scenario. It includes the system model and its associated DRL. It tackles the dynamic zone situation challenges including the multi-dimensional factors in the intra-zone resource facilitation and serves as a solid base for an efficient inter-zone resource facilitation. The scheme incorporates an optimization model that aims to maximize system utility. The optimization model is integrated with the ADCPG DRL to generate actions that align with optimization goals. The ADCPG DRL features CMDP characterizing the complex and dynamic Edge-IoT environment in both static and stochastic state sub-spaces and includes the novel actor dual-critic policy gradient design.

1) Intra-zone Resource Facilitation Model: The model aims to maximize the utilities of Edge-IoT based on the resource facilitation and application priority decided according to the current intra-zone situation. For the edge, the edge server utility is found based on the edge server data processing rate (DPR). DPR is the number of tasks that can be processed using the facilitated resources. It is a key factor that evaluates the efficiency of the edge servers as low value indicates that the server is not functioning properly. For the IoT, the IoT applications' utility is defined based on multiple QoS metrics including latency (LAT) and data loss rate (DLR). LAT includes both the network delay and processing delay at edge. DLR is the packet loss rate due to network issues or queuing at the edge server.

We formulate the intra-zone model as follows. At time t, the ZEF decides the application p to be prioritized. The IoT devices running application p are selected for resource facilitation. Application priority is dynamic and changes according to the current application's demands. The ZEFs receive all the applications' requests in certain decision cycle including all low and high priority applications and compare them against the available resources to determine whether the current capacities of the servers are sufficient to handle all the requests. If they are sufficient and all the applications can be served, the ZEFs determine the applications' priorities according to their data volume and QoS requirements. They facilitate resources to the applications according to the assigned priority with the condition that all applications will be served. Otherwise, ZEFs will request the IEF support through the inter-zone facilitation. The resource facilitation decision of each resource R of type z for each IoT device $d \in D_p$ that belongs to application $p \in \mathbf{P}$ is evaluated in terms of the edge and IoT application utilities. The resources are allocated to the IoT application p such that the edge utility function Γ_j and the application utility function Ω_p are maximized. We define a set of three performance metrics: $B = \{b_1, b_2, b_3\}$, where $b_1 = DPR$, $b_2 = LAT$, $b_3 = DLR$. Let us assume that $x_{b,d}$ is the achieved value for each performance metric $b \in B$ and $x'_{b,d}$ is its threshold. At the application level, the metric vector $x_p = [x_{b_1,p,d}, x_{b_2,p,d}, x_{b_3,p,d}]$ must meet its corresponding x'_p.The edge utility function $\Gamma_j(x_p)$ is defined as a function of $x_p = b_1$ for the edge server j that processes the tasks of application p. The application utility function is $\Omega_{p,R_z}(x_{p,d})$, where $R_z \in R = \{R_1, R_2\}$ represents the set of resources ($R_1 =$ CPU, $R_2 =$ memory) allocated to the IoT application p and x_p is the performance metric vector for QoS metrics b_2 and b_3. The intra-zone facilitation model related parameters are defined as: 1) the resource allocation variable $y_{d,r}(t) = 1$ if resource $r \in R_z$ is allocated to IoT device d and 0 otherwise. R_z is the set of available resources of certain type; 2) the application priority indicator $P_p(t) = 1$ if the application p is prioritized at time t and 0 otherwise. The intra-zone model optimization function (ZM) is formulated as:

$$ZM = \max_{y,P} \sum_{p \in \mathbf{P}} \sum_{d \in D_p} \sum_{r \in R_z} P_p(t)\ y_{d,r}(t)\ \Gamma_j(x_p(t))\ \Omega_p(x_{p,d}(t)) \tag{1}$$

$$s.t.$$

$$\textit{\textbf{(C.1)}} \sum_d y_{d,r}(t) \leq 1, \quad r \in R_z, \tag{2}$$

$$\textit{\textbf{(C.2)}} \sum_p P_p(t) \leq 1, \tag{3}$$

$$\textit{\textbf{(C.3)}} \sum_d P_{p*}(t) = D_{p*}, p* \in \mathbf{P}, \tag{4}$$

$$\textit{\textbf{(C.4)}} \sum_d P_{p'}(t) = 0, \forall p' \in \mathbf{P} \tag{5}$$

$$\textit{\textbf{(C.5)}}\ x_{b,p,d}(t+1) \leq x'_{b,p,d} \tag{6}$$

$$\textit{\textbf{(C.6)}} \sum_{p \in \mathbf{P}} \sum_{d \in D_p} \sum_{r \in R_z} y_{d,r}(t) \leq C_j\ \forall j, \tag{7}$$

$$\textit{\textbf{(C.7)}} \sum_{R_z \in R} \sum_{r \in R_z} y_{d,r}(t) \geq W_d \tag{8}$$

The solution of (1) is to select the best priority and facilitation action at t such that the utility functions for IoT and edge are maximized. The constraints in (***C.1***) indicates that each unit of resource can only be allocated to one device. (***C.2***) specifies that only one application will be prioritized at time t to obtain resources. The devices running the prioritized application $p*$ will qualify for resources as in (***C.3***). The devices that run

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3151667, IEEE Internet of Things Journal

other applications p' will be blocked at t (***C.4***). (***C.5***) enforces the QoS requirements of the applications. (***C.6***) guarantees that the allocated resources for all applications are below the edge server capacity C_j. (***C.7***) ensures that the total allocated resources for each IoT device satisfy its demand W_d for all the resource types.

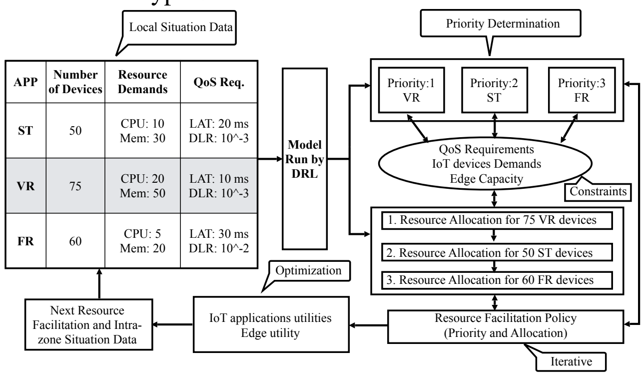


Fig. 4. Intra-zone resource facilitation example.

We illustrate the intra-zone facilitation via an example of three applications: (a) Smart Transportation (ST), (b) Virtual Reality (VR), and (c) Face Recognition (FR) as shown in Fig. 4. First, zone situation information is acquired at t including: the number of devices, applications' dynamic resource demands of CPU and memory, and the applications' QoS requirements $B = \{LAT, DLR\}$. Then, the DRL runs with the model to determine: 1) the real-time application priority, in this example, the highest priority is given to the VR as it currently has high demand with considerable number of devices and moderate QoS requirements; 2) for prioritized VR, resource facilitation policy is also generated to allocate resources for the IoT devices under the constraints in (***C.1-C.7***). The process is repeated for the ST and FR applications.

2) Novel ADCPG Approach for Intra-zone Facilitation: We plan to base our method on the actor-critic framework [47] to benefit from both value-based and policy gradient DRL. In addition, given the multi-dimensional factors of the intra-zone resource facilitation and the dynamicity of Edge-IoT environment, the proposed ADCPG method features a unique critic structure and process to improve and expedite the action policy learning process comparing with other existing actor-critic methods [42], [48], [49]. We first formulate the CMDP that features two-dimensional space of attributes: (static and dynamic) and consists of the following components: **1) State:** The state reflects static and dynamic zone situation information and is defined at t as $s(t) = (s^{\epsilon}(t) = (x', d, p), s^{\upsilon}(t) = W_p) \in S$, where $s^{\epsilon}(t)$ is the static attribute with x' representing the QoS requirements of the application p and d is the IoT device that belong to application p. The dynamic state attribute $s^{\upsilon}(t)$ is defined as the dynamic demand of the application (W_p) found according to the application data volume. **2) Action:** The action at t is defined jointly with two attributes as $a(t) = (P_p, Y_{r,p,d})$, where P_p is the priority of application p and Y is the resource facilitation. Given the current application demand W_p, the state $s(t)$ will evolve based on $a(t)$ as $s^*(t)$ and record the achieved performance metric x^* as a result of the action $a(t)$. The state transition function is defined as $f : S \times A \rightarrow S^*$, $s^*(t) = f(s, P, Y)$. **3) Reward:** The reward value evaluates how application and edge server utilities will improve at $t+1$ in comparison to t with action $a(t)$, and is expressed as $RW(s^*, s) = RW(s^*) - RW(s)$ where RW is associated with the joint applications and server utilities found in equation (1). **4) Action Policy and Value Function:** we define the resource facilitation policy generated by the actor as $\pi : S \times A$ that maps the Edge-IoT system state over the action space. We define the value function under the given facilitation policy π as $V^{\pi_\phi}(s) = E_\pi[\sum_t^\infty \gamma^t RW_{t+1} | s(0) = s]$ representing the sum of rewards from the initial state, where $\gamma^t \in [0, 1]$ is the discount factor. The reward function RW is defined as: $RW = ZM$, where ZM is defined in Equation (1) as a function of the edge and IoT application utilities given the resource facilitation action taken by the ADCPG-DRL mechanism.

The proposed ADCPG method workflow is presented in Fig. 5. It begins as ADCPG interacts with the environment via the defined CMDP and experience samples are stored in the replay memory. Training batch will be fetched from the memory to train both critics. The specialized additional critic (S-Critic) supports the original critic (O-Critic) to improve the action policy learning process. The S-Critic provides an additional loss value denoted by L_ζ with network parameter ζ optimized during the learning process. This loss guides the actor and it is explicitly trained to find a resource facilitation action given the multi-dimensional factors involved in the intra-zone facilitation instead of merely estimate the value function as in the typical actor-critic. The O-Critic provides the loss value L_θ in addition to L_ζ to train the actor to generate the action policy π_ϕ using stochastic gradient descent, where ϕ is the actor network parameter. The action policy is updated by defining the actor loss in terms of the expected return $J(\phi)$ and taking its gradient $\nabla_\phi J(\phi)$, where $J(\phi)$ is evaluated according to the value function $V^{\pi_\phi}(s)$. We formulate the actor network ϕ learning process using the gradient of both critics as follows,

$$\phi^* = \min_\phi (L_\theta(DA_{trn}; \phi) + L_\zeta(DA_{trn}; \phi)) \tag{9}$$

L_θ of the O-Critic is found using a training batch DA_{trn} sampled from the memory as $L_\theta = -J(\phi) = -E_s V^{\pi_\phi}(s_t; \theta)$, where $E_s = \gamma RW_t$. The O-Critic uses the estimated value function to update its network parameter θ. The S-Critic consists of the network $g_\zeta(DA_{trn}; \phi)$ which takes ϕ and the state/action in DA_{trn} as input and outputs a scalar value.

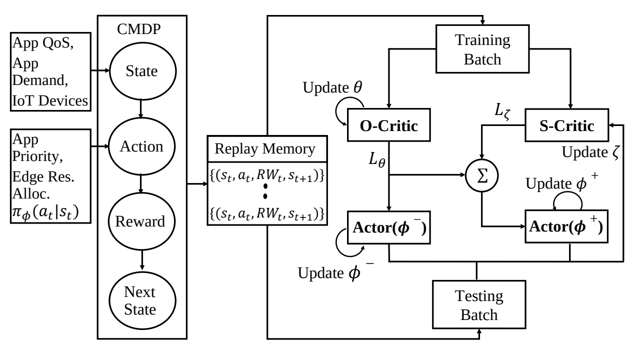


Fig. 5. The ADCPG DRL scheme.

This value represents the loss value L_ζ which is differentiable with respect to ϕ. The actor network parameter ϕ is updated using both critics losses as follows: 1) O-Critic loss

L_θ: ϕ is updated to increase the probability of actions that achieve higher value functions as $\phi^- = \phi - \rho \frac{\partial L_\theta(DA_{trn})}{\partial \phi}$; 2) S-Critic and O-Critic losses (L_θ, L_ζ): the parameter ϕ is updated to stabilize and improve the learning performance as $\phi^+ = \phi - \rho \frac{\partial L_\theta(DA_{trn})}{\partial \phi} - \rho \frac{\partial L_\zeta(DA_{trn})}{\partial \phi}$, where ρ is a constant. To guarantee that the S-Critic will improve the learning performance and will not overfit, a testing batch DA_{tst} is sampled from the memory and used to find the S-Critic intrinsic loss as $L^S(DA_{tst}; \phi^-, \phi^+) = \tanh(L_\theta(DA_{tst}; \phi^+) - L_\theta(DA_{tst}; \phi^-))$. The S-Critic network parameter ζ is updated to minimize L^S and consequently maximize the performance on the testing batch. Thus, the real actor network is updated using both critics. This also guarantees that S-Critic and actor networks are evolved online and in parallel. We exploit the Optimistic Actor Critic exploration method proposed in [51], which approximates a lower and upper confidence bound on the value function. This method performs direct exploration using the upper bound while still using the lower bound to avoid overestimation. The optimistic exploration tackles the problem of combining a greedy actor update with a pessimistic estimate of the critics thats leads to the avoidance of new actions. In addition, it avoids sampling actions with equal probability in opposite directions from the mean as we need actions taken in certain directions much more than others.

C. Intelligent Inter-zone Resource Facilitation Scheme

In this subsection, we develop the inter-zone facilitation model and the associated DRL which tackles the complicated scenario represented by the outlandish edge situations and mobility of IoT devices. Specifically, we develop an inter-zone optimization model for the inter-zone facilitation that responds to the complicated situations by searching and assigning a new ZEF. Moreover, we expand CACPG DRL to generate facilitation policies that maximize the institution-level objectives via cooperation between ZEFs and the IEF. The inter-zone situation information and the intra-zone DRL policies are exploited to train the institution level critic and generate actions that determine the inter-zone resource facilitation at the institution scale.

1) Inter-zone Resource Facilitation Model: Since the inter-zone facilitation extends the decision of the intra-zone facilitation and assigns certain ZEFs to accommodate the affected IoT devices, the proposed inter-zone model's optimization objective incorporates the intra-zone optimization function ZM, adaptation cost and transiting cost functions for the outlandish conditions. We define an adaptation cost (AC) as the cost of powering certain edge server components to provide resources to the affected IoT devices. Transiting cost (TC) is the cost of transiting the affected IoT devices to another edge server more suitable to provide the requested resources. The inter-zone model leverages a holistic view of the environment to make intelligent collective decisions achieving lower costs comparing with greedy or conservative strategies. We illustrate its collective decision effects via a simple example in Fig. 6 which involves server A and B connected to two ZEFs, and an IoT device needs alternative resource provider due to roaming (scenario 1) and edge failure (scenario 2). In scenario 1, the device roams from server A to B at $t = 1$ and move back at $t = 2$. Greedy strategy suggests transiting the device's task from A to B at $t = 1$ and transiting back at $t = 2$. The inter-zone model calculation shows that it is better to keep the workload on server A (utility gain: 20 vs. 7.5). In scenario 2, server A suddenly overloaded/fails for one time unit and the workload is transited to server B. Conservative strategy suggests keeping workload on server B at $t = 2$, while the inter-zone model suggests transiting workload back (utility gain: 16 vs. 11.5). In both scenarios, the holistic collective inter-zone model achieves better results.

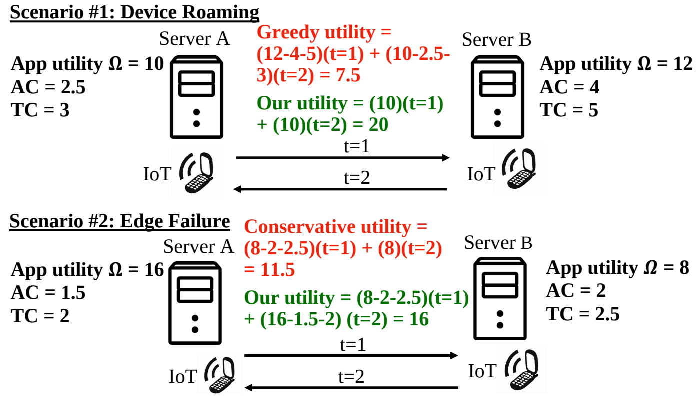


Fig. 6. Inter-zone facilitation example

We formulate the inter-zone model as follows. Suppose $ZEF = \{m_1, m_2, ...m_M\}$ are interconnected in an institution scale and the set of IoT devices D_p is distributed and free to roam. The IEF selects the most suitable ZEF or distribute loads over multiple ZEFs to align with the inter-zone optimization goal. The intra-zone optimization function in (1) is re-formulated with latency $LAT(loc_{d,t}, m_{d,t})$ between device location loc and the corresponding ZEF,

$$ZM^* = \sum_{d \in D_p} [LAT(loc_{d,t}, s_{d,t}) +$$

$$\sum_{s \in ZEF} \sum_{p \in \mathbf{P}} \sum_{r \in R_z} P_p(t)\ y_{s,d,r}(t)\ \Gamma_j(t)\ \Omega_p(x_p(t))] \tag{10}$$

The AC is proportional to the workload (data volume) and the amount of resources allocated to the affected devices,

$$AC = \sum_{j \in servers} c_j^a \Big(\sum_{d \in D_p} Y_{j,d}(t) - \sum_{d \in D_p} Y_{j,d}(t-1) \Big)^+ \tag{11}$$

where $c_j^a > 0$ is the cost of increasing unit resource for the server j. $(\sum_{d \in D_p} Y_{j,d}(t) - \sum_{d \in D_p} Y_{j,d}(t-1))^+$ captures the increase of the device workload on server j from time $t-1$ to t, and $Y_{j,d}$ is the amount of resource allocated to device d, and $(V)^+ = \max\{V, 0\}$. TC is determined according to the required bandwidth and the incurred delay for transiting,

$$TC = \sum_{j \in servers} B_j^{out} W_j^{out}(t) + B_j^{in} W_j^{in}(t) \tag{12}$$

where the data moving in and out of j are B_j^{out} and B_j^{in}. $W_j^{out} = (\sum_{d \in D_p} Y_{j,d}(t-1) - \sum_{d \in D_p} Y_{j,d}(t))^+$ and $W_j^{in} = (\sum_{d \in D_p} Y_{j,d}(t) - \sum_{d \in D_p} Y_{j,d}(t-1))^+$ are the device load that is transited out and received by the new server. The institution-level optimization function (IM) maximizes ZM and minimizes all the associated costs as follows,

$$\max_{j,P,y} IM = ZM^* - AC - TC \quad s.t. \tag{13}$$

$$\textbf{\textit{(C.8)}} \quad \sum_{d \in D_p} \sum_{r \in R_z} y_{d,r}(t) \leq C_j, \tag{14}$$

$$\textbf{\textit{(C.9)}} \quad \sum_{j \in servers} Y_{j,d}(t) \geq W_d \tag{15}$$

where $Y_{j,d} = \sum_{r \in R_z} y_{d,r}(t)$ is the amount of resources allocated by the corresponding ZEF at server j to device d. The optimization problem here is subjected to all the constraints of the intra-zone model in addition to the ones in (14) and (15), which takes the selected server associated with the ZEF of the zone in consideration for the capacity and workload constraints.

2) Novel CACPG Approach for Inter-zone Facilitation: The novel CACPG DRL exploits the intra-zone policies generated by ADCPG from multiple ZEFs to achieve the inter-zone facilitation policies. It involves multiple zones cooperating with their IEF through ZEF/IEF resource facilitation policy updates. We formulate the inter-zone collective MDP for the upper DRL as follows. **1) Global State:** it includes an aggregation of the intra-zone states defined in Section IV.B.2 for the affected zones with outlandish conditions and the their situation information as $s_g(t) = \{s_{m(t)}\}_{\forall m \in ZEF}$, where m is the index of the ZEF. **2) Global Action:** the action determines the new ZEF that facilitates the resources to the IoT devices affected by outlandishness/roaming in addition to the intra-zone resource facilitation decision by ZEF defined in Section V.A.2. It is expressed as $a_g(t) = \{(P_p, Y_{r,p,d})\}_{\forall m}$. **3) Global Reward:** it incorporates the intra-zone reward $RW(s)$ and the associated costs $C_g = AC + TC$ and defined in equation (13) as a modified reward $RW_g(t) = f(RW(s), C_g)$. The total inter-zone reward is defined as an aggregation of the modified intra-zone rewards as $RW_G(t) = \sum_{m \in ZEF} RW_g(t)$. The system evolves to the next state $s'_g(t)$ and the transition function is given as $(s_g(t), a_g(t), RW_G(t), s'_g(t+1))$. The inter-zone facilitation action selection is defined by an inter-zone policy π_g which is a mapping from a given inter-zone state to an inter-zone action. The inter-zone value function is given as $V^{\pi_{\phi_g}}(s_g) = E_\pi[\sum_t^\infty \gamma^t RW_G(t+1) | s_g(0) = s_g]$.

The proposed CACPG DRL scheme implemented at the IEF is shown in Fig. 7. The intra-zone CMDP components including the action policies and value functions from the ZEFs are aggregated to initially populate the inter-zone replay memory associated with the inter-zone actor-critic. The inter-zone actor uses the replay memory training data samples DA_{g-trn} and the inter-zone state information to generate the inter-zone action policy π_{ϕ_g} that maximizes the function in equation (13) and denoted by J^*. The parametrized policy π_{ϕ_g} is directly updated by the loss function L_{θ_g} given by the inter-zone critic in terms of the expected return $J^*(\phi_g)$ and taking its gradient $\nabla_{\phi_g} J(\phi_g)$. Thus, the inter-zone actor ϕ_g improves its learning according to the following optimization function $\phi_g^* = \min_{\phi_g}(L_{\theta_g}(DA_{g-trn}; \phi_g))$. The inter-zone

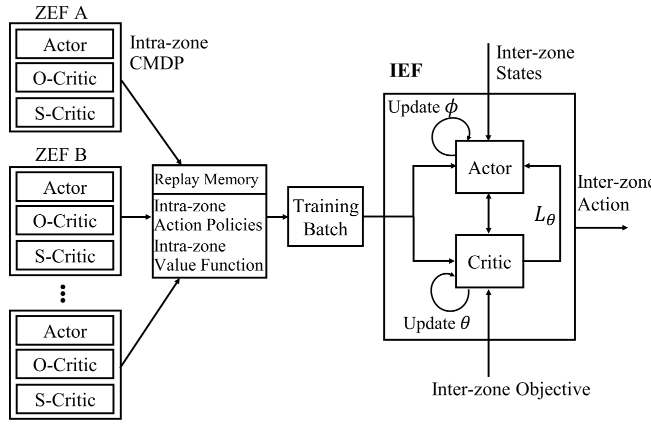


Fig. 7. Collective Actor-Critic DRL.

critic evaluates the actions generated by the inter-zone actor via the following loss function,

$$L_{\theta_g} = -J^*(\phi_g) = -\gamma RW_G(t) V^{\pi_{\phi_g}}(s_g(t); \theta_g) \tag{16}$$

where ϕ_g and θ_g are the actor and the critic parameters respectively. The inter-zone critic estimates the value function and update θ_g as,

$$\theta_g \leftarrow \min_{\theta_g}(V^{\pi_{\phi_g}}(s_g(t); \theta_g) - RW_G(t) - \gamma V^{\pi_{\phi_g}}(s_g(t+1); \theta_g))^2 \tag{17}$$

The iterative procedure of the inter-zone action generation is illustrated in Algorithm 1. The procedure takes the inter-zone state information and the intra-zone actions from the replay memory as inputs. It generates an action set $A = a_g^{(i)}[m'](t) \cup$

Algorithm 1: Inter-zone DRL Procedure

Require: Inter-zone state $s_g(t)$,intra-zone action policies generated by each ZEF $a_{\boldsymbol{m}}(t) = [a_1(t), a_2(t), a_3(t), \ldots., a_{\boldsymbol{M}}(t)]$, $\boldsymbol{M}$ is the # of ZEFs

Ensure: inter-zone action $a_g(t)$

1: $a^{(0)} \leftarrow a_m(t)$
2: **for** $(i = 0,1, \ldots., N)$ **do**
3: **for** $(m = 1,2, \ldots., M)$ **do**
4: Generate an action set $A = a_g^{(i)}[m'](t) \cup a_m(t)$;
5: Train the inter-zone actor and critic;
6: Update the action of ZEF m as:
7: $a_g^{(i)}[m](t) = \max_{a_j} V_g^\pi(s_g)$;
8: **end for**
9: Update the actor-critic parameters ϕ_g and θ_g;
10: **end for**
11: $a_g(t) = a_g^{(N)}(t)$;

$a_m(t)$, for the involved ZEFs. The inter-zone action for the ZEF m is updated such that it has the maximum inter-zone critic value function as shown in line 7 and minimal loss value. The procedure is iterated until it converges and outputs the inter-zone actions for all the affected ZEFs.

The IEF does not need to recompute everything and use the output of the ZEFs as seeds to find the inter-zone resource facilitation policy. The IEF only coordinates the facilitation of special outlandish and roaming occasions among the affected ZEFs for a limited period of time and do not replace the role of individual ZEFs in resource facilitation. After the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3151667, IEEE Internet of Things Journal

special situation or transition is over, the new ZEFs handle the affected devices. Thus, the IEF does not become a new single performance bottleneck. Even if the IEF goes offline, it does not affect the independent functioning of the low-level ZEFs in their zones. The complexity of the DRL in *I-HARF* is found based on the two DRL components: the zone level run at the ZEF and the institution level implemented at the IEF. The complexity of the ADCPG DRL is tied to the number of IoT devices D_p involved and is calculated as $O(D_p R_z)$, where R_z is the amount of resources available at the local ZEF. The training complexity is obtained according to the number of epoch and the training interval for the ADCPG as $O(2N\Omega)$, where Ω is the training interval. For the inter-zone facilitation, the complexity calculation is affected by the number of ZEFs and their operation complexity and found as $O(MD_pR_z)$. The training complexity of CACPG is calculated as $O(2MN\Omega + N\Psi)$ where Ψ is the training interval at the IEF.

D. Resource Reservation Mechanism in Inter-zone Resource Facilitation

This subsection tackles the problem of the exceptional scenario which comprises sensitive IoT applications that cannot tolerate latency introduced due to coordination between ZEFs and IEF in the inter-zone resource facilitation. The sensitive applications include sensitive data based on which extremely urgent responding decisions must be made while coordination latency is avoided. A typical scenario in smart campus is explained as follows: emergency response is sensitive application while academic AI research data processing is less sensitive. In certain campus event (exceptional scenario), the environment changes rapidly and the volume of data that needs processing from different applications is significant which overloads the edge servers. Consequently, the inter-zone resource facilitation engages to handle the applications resource requests. However, these applications may experience extra latency due to ZEFs/IEF coordination. Emergency response does not tolerate such latency as other applications such as AI research does.

The proposed resource reservation mechanism reserves the edge resources for those sensitive applications such that their associated tasks are processed in a timely manner. It relies on prediction of the volume of the tasks for such applications. The sensitive applications are identified by the system based on their latency requirements. The applications with sensitive latency requirement qualify for resource reservation. The sensitivity of the latency requirement is evaluated by comparing it against certain threshold determined according to the applications executed in the Edge-IoT system. The reserved resources are utilized to process the sensitive applications tasks upon arrival which mitigate the coordination latency impact. The CACPG structure presented in Section IV.C.2 is modified such that the exceptional scenario occurred at certain event is handled using the inter-zone DRL with the resource reservation mechanism depicted in Fig. 8. The resource reservation module (RRM) receives the prediction information from the prediction module and execute the reservation mechanism which uses the prediction of the sensitive applications demands, the current demand from all the running applications, and the resource facilitation policy of the ZEFs. The CACPG actor receives the output of the RRM as an additional input that contributes to the inter-zone resource facilitation policy generated by the CACPG. Prediction of the sensitive applications resource demands in such Edge-IoT

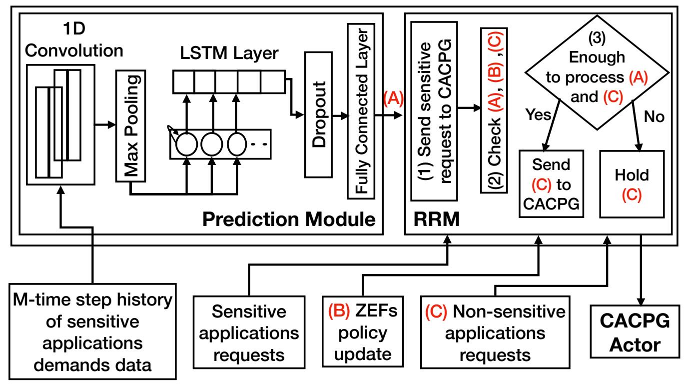


Fig. 8. Resource reservation mechanism.

environment is non-trivial as it comprises complex interplay among the applications data generation patterns and the environment variation considering possible random outlandish events. Thus, we adopt a prediction module that exploits a hybrid prediction approach that relies on convolutional neural networks (CNN) [52] and long short term memory (LSTM) [53]. This hybrid approach achieves accurate prediction results as CNN improves features extraction. The features of the prediction model in the reservation mechanism consist of real-time observable parameters that impact the resource reservation decision. The features include M-time history window of the application requests of resources including CPU and memory, computation and communication loads of the application requests, and requests processing time. With these features history window as input, the prediction model outputs the N time-step ahead of the application resource demands. The prediction method is a multivariate time series forecasting problem that predicts multiple time steps ahead. It exploits the advantage of combining CNN and LSTM to achieve prediction with high accuracy and small training samples. Convolutional layers with pooling layers in the CNN-LSTM prediction module capture the local dependencies and the invariant in the data features. A 1D convolutional layer with multiple filters of certain kernel size is used on the input data to obtain time step-wise information from the input features, and comprehend their local dependencies and invariance over features in every time sample. Max pooling extracts the invariant attributes and feeds the output to the LSTM network. Each feature fed to the LSTM has a dimension equivalent to the number of filters in the CNN. The LSTM considers the time series as a sequence of dimensional feature vectors. A drop out layer is introduced for regularization. Its output is moved to a fully connected layer to output the prediction results. The complexity of the prediction is kept low with choice of small number of convolutional layers which is the case as the prediction is only necessary ahead of certain outlandish occasions. RRM exploits the prediction output to determine the reservation decision as follows. At certain time slot t, it checks the current applications' resource requests and forwards the requests of the sensitive application to the CACPG be allocated first. Then, RRM checks the remaining edge resources, the non-sensitive application requests, and the projected sensitive application requests obtained using the prediction module. If the remaining edge resources are sufficient to process the non-sensitive applications requests and the projected sensitive requests, the RRM pushes the non-sensitive application requests to the CACPG for resource allocation. Otherwise, it holds the non-sensitive requests to the next allocation time slot.

V. Performance Evaluation

We evaluate the performance of *I-HARF* for intra-zone facilitation, inter-zone facilitation, and resource reservation for sensitive applications in the heterogeneous smart campus environment explained in Section IV with possible outlandish and mobility situations. The performance is evaluated in terms of the rate of successful processing of application requests (SPAR), system utility, average latency of sensitive applications and system convergence.

A. Evaluation Setup

In the following evaluation, we simulate an Edge-IoT environment that includes 200 IoT devices and 20 ZEFs with two types of resources: CPU and memory. These numbers are used in all the simulations unless otherwise indicated. We consider four types of IoT applications with various requirements: Emergency Response (ER), Virtual Reality (VR), Face Detection (FD) and Smart transportation (ST). The QoS in $[LAT, DLR]$ are set for: ER as $[20ms, 10^{-3}]$, VR as $[45ms, 10^{-2}]$, FD as $[60ms, 10^{-2}]$ and ST as $[30ms, 10^{-3}]$. The number of IoT devices deployed in this simulation is variable with ratio of 1/4 for each application. The resource demand (request) for each IoT device is determined according to its application. They are generated following Poisson distribution in the following ranges [0.1, 0.8] for vCPU and [0.8, 4] GB for memory. We normalize the resource capacity of edge servers. Thus, the resource capacity of each edge server is of one unit. The computing capacity of the edge servers is set between 1 GHz and 6 GHz. The average data transmission rate is distributed between 250 Mbps and 1200 Mbps. For the actor and critic networks, we use fully connected DNNs with 2 hidden layers of 250 neurons and ReLU activation function for good performance and adequate complexity. The replay memory capacity is 5000 samples. The learning rates are set as (0.0005, 0.001) for the actor and two critics respectively. Actor-Critics DRL algorithms are executed using double Intel i7 quad core 3.4 GHz CPUs, 16 GB Random Access Memory (RAM), and 512 GB disk. The edge servers are chosen from the set of M4 Amazon EC2 instances [54]. Amazon M4 instance of type M4.10xLarge includes 40 vCPU, 160 GiB of memory, and 4 GHz of bandwidth.

Fig. 9 presents the simulation setup and the steps followed to facilitate resources described as follows.

First Step (Intra-zone resource facilitation in typical scenario): The ZEF acquires the current situation information from the associated IoT environment and runs the intra-zone model integrated with ADCPG DRL to facilitate resources for the IoT applications with the goal of system utility maximization.

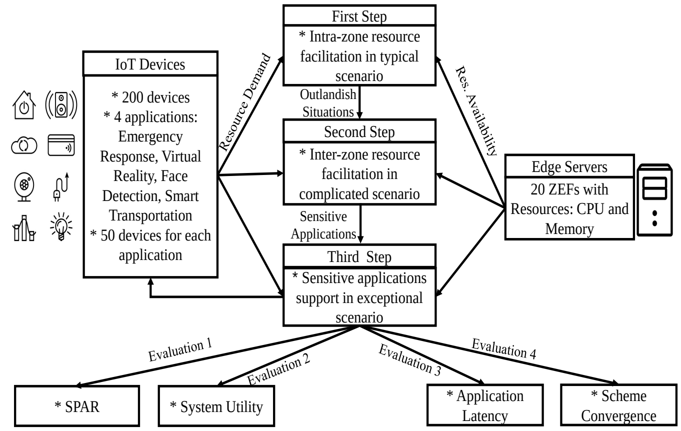


Fig. 9. Evaluation setup and related steps

Second Step (Inter-zone resource facilitation in complicated scenario): This step is required when outlandish situations occur or an IoT device roams from one zone to another. The IEF retrieves the current situation of its associated ZEFs and runs the inter-zone model in conjunction with CACPG DRL to facilitate resources for the IoT applications with the goal of system utility maximization and minimization of the associated costs.

Third Step (Sensitive applications support through resource reservation in exceptional scenario): This step complements the inter-zone resource facilitation in case of existence of an IoT sensitive application that needs further measures to guarantee its latency requirements. The reservation mechanism is executed to assist CACPG in resource facilitation with minimal latency.

Various evaluations detailed in the next subsections are conducted to demonstrate *I-HARF* capabilities. On the one hand, we compare the performance of *I-HARF* to resource allocation methods including DRL-based systems (DQN) in [30] and Actor-Critic in [34] for the intra-zone facilitation. On the other hand, *I-HARF* is compared against resource allocation methods developed for resource facilitation from multiple edge-servers including (RA-QoS) in [55] and DDQN-FL in [56].

B. SPAR Evaluation

SPAR is the normalized successfully processed applications requests at the edge. It is calculated as the ratio of the number of the processed applications requests to the total number requests initiated by the applications. We evaluate the achieved SPAR for each IoT application in the intra-zone facilitation mode and study the impact of the possible outlandish situations on the achieved SPAR in the inter-zone facilitation. Fig. 10 presents SPAR for each IoT application in the intra-zone facilitation. The figure shows that *I-HARF* maintains SPAR at ratio close to 1 in comparison to other schemes. To study the impact of the outlandish situations on the achieved SPAR which mainly impact the capacity of the resources available at the edge, we plot SPAR against a variable number of IoT devices while some ZEFs are with

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3151667, IEEE Internet of Things Journal

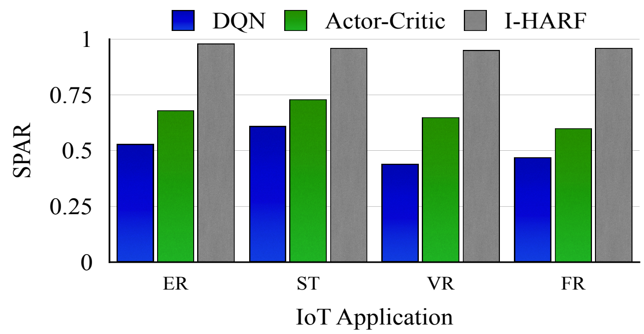


Fig. 10. SPAR of different IoT applications.

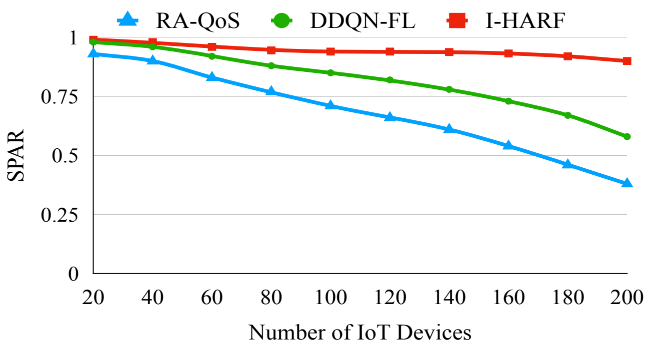


Fig. 11. SPAR vs. number of IoT devices.

randomly changing status. Fig. 11 presents the achieved SPAR vs. the number of IoT devices. Increasing the number of IoT devices leads to significant demand for resources. This overwhelms the corresponding ZEF which becomes incapable to fulfill the demand due to the insufficient resources. Due to the poor management in DDQN-FL and RA-QoS, they are not capable to find the most appropriate alternative ZEF to provide the required resources to the increasing number of IoT devices. However, IEF in I-HARF coordinates with its associated ZEFs to accommodate the resources' requests of the increasing IoT devices. *I-HARF* maintains the SPAR at a high level even when the number of IoT devices is large. We clearly notice that *I-HARF* outperforms other DRL based resource allocation schemes in the intra-zone facilitation and the systems that involve allocation of resources from multiple edge servers specifically at critical system settings with a large number of IoT devices. The achieved SPAR by I-HARF is justified as I-HARF incorporates dynamic priority assignment mechanism supported by the novel DRL scheme, which allows the system to process different application' requests successfully at high rates. With dynamic priority, the system becomes able to accommodate more requests regardless of the application types or load. The hierarchical system structure with two-DRL levels enhance the processing capacity of the applications' requests as multiple servers can be involved to fulfill the demands in a real-time decision-making fashion as in Fig 11.

C. System Utility

The system utility presented in (13) evaluates the efficacy of *I-HARF* for both IoT side and edge side. From the IoT side, it indicates if the QoS requirements of the IoT applications are satisfied while it evaluates the efficiency of tasks processing at the edge side. The system utility includes the IoT application and the associated edge server utilities. The application utility $\Omega_p(x_p)$ is a function of the QoS metrics achieved by the application and the edge utility $\Gamma_j(x_p)$ is found based on the data processing rate of an edge server for certain application. In the following, we evaluate the system utility in the typical and the complicated scenarios. Fig. 12 presents the system utility vs. the normalized task arrival rate in the typical scenario. The task arrival rate follows Poisson distribution from all the IoT applications. Fig. 12 indicates that the utility decreases as the task arrival rate increases as there is more demand for the resources at the edge.

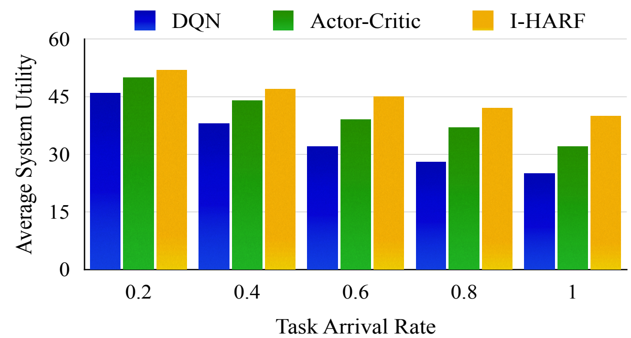


Fig. 12. Average system utility in the intra-zone facilitation (Typical Scenario).

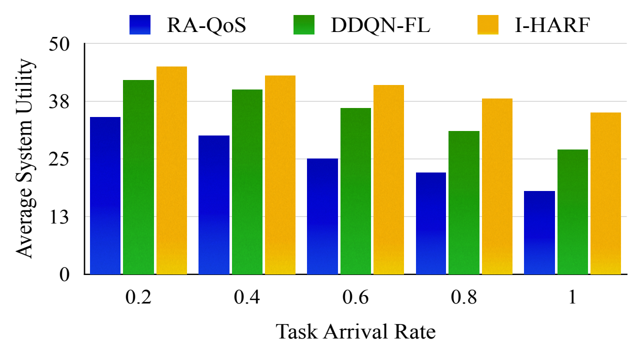


Fig. 13. Average system utility in the inter-zone facilitation (Complicated Scenario).

To study the impact of the outlandish situations and mobility in the complicated scenario on the achieved system utility which mainly impacts the resource availability at the edge, we focus on mobile application (smart transportation) and some of the ZEFs experience malfunctions. Fig. 13 presents the achieved system utility vs. the the task arrival rate. It is shown that the mobility of devices and the failure of the ZEFs lead to a decline in the achieved utility as the resource availability becomes limited and competition for resources between IoT devices escalates. However, *I-HARF* manages to maintain the utility at reasonable level in comparison to other methods and to the typical scenario. We clearly notice that *I-HARF* outperforms other resource allocation methods in the IoT system utility evaluation specifically at critical system settings in the complicated scenario. In addition, we notice that *I-HARF* maintains the system utility in the complicated scenario close to the one in the typical scenario and is within the range of 10% difference. The rationale for the ultimate performance of I-HARF is that it focuses on both application and edge utilities in the developed intra-zone model and incorporates transit and adaptation costs in the inter-zone model.

Considering both utilities makes the system more efficient in matching the applications QoS requirements with the available resources at the edge. Additional costs considered in the inter-zone model causes the utility functions to be more practical as migrating the processing of the applications' requests from one server to another is not granted without considering the time and coordination required for it.

D. Latency of the Sensitive Applications

In this evaluation, we demonstrate *I-HARF* capability to support sensitive applications in the exceptional scenario and guarantee their latency requirements. We focus on the emergency response as a typical sensitive application. Latency is picked for evaluation here as it is the most critical QoS metric for these applications. Fig. 14 presents the average latency of the ER application vs. its number of tasks to be processed. The figure shows that *I-HARF* keeps the latency of the application below the threshold which is 500 ms despite the outlandish situation. The tail (99th percentile) latency of the emergency response application is plotted in Fig. 15. This evaluation clearly demonstrates the efficacy of the resource reservation mechanism. Latency of the sensitive applications such as ER plotted in Fig. 14 demonstrates I-HARF capability to accommodate sensitive application additional requirements as these applications cannot tolerate additional latency that might be encountered due to coordination in the inter-zone facilitation. The advantage of the developed reservation mechanism in I-HARF is clearly demonstrated in the latency evaluation. Reservation of resources for the sensitive applications in the inter-zone facilitation allows these applications to process their data in timely manner in comparison to schemes that leave the resource facilitation to the existing environment conditions that cause intolerable latency by the application.

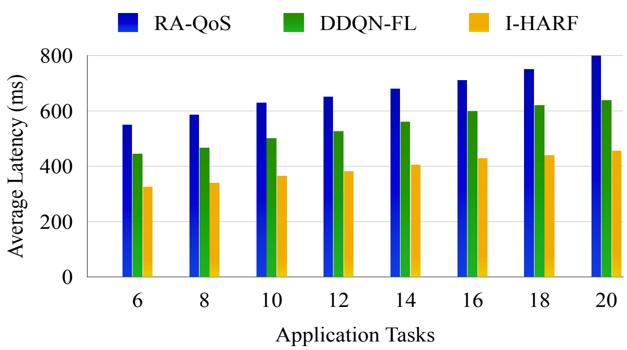


Fig. 14. Average latency of ER application.

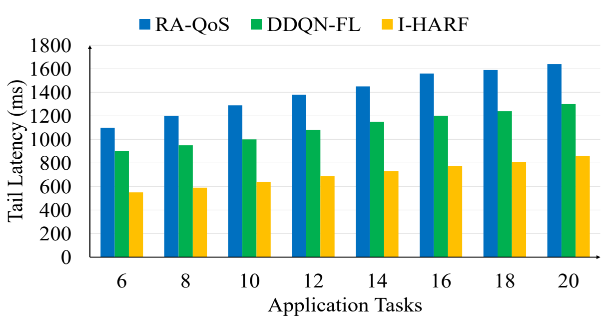


Fig. 15. Tail (99th Percentile) latency of ER application.

E. I-HARF Convergence

We conduct this evaluation to demonstrate the convergence performance of *I-HARF*. The convergence is evaluated in the typical and complicated scenario using the system utility. Fig. 16 shows the achieved system utility against the number of epoches in the typical scenario with task arrival rate of 0.8. We notice that at the beginning, the utility is low because DRL agent does not have enough experience to make rational decisions for resource facilitation. With the increase in the number of epoches, the utility increases gradually until a relatively stable value is reached. Fig. 16 also shows that *I-HARF* converges faster than other DRL based frameworks. The plot in Fig. 17 presents the system utility in the complicated scenario. The figure indicates that *I-HARF* converges faster than the other frameworks and achieve higher system utility.

All the evaluations reveal the advantages of *I-HARF* design principles to build a resource facilitation framework that is adaptive to all possible situations. These principles include: 1) dynamic priority which is determined in real-time to match with the application current data volume; 2) ADCPG DRL which is capable of resolving the multi-dimensional intra-zone resource facilitation problem; 3) hierarchical resource facilitation using CACPG DRL which is able to facilitate resources in complicated scenarios using the inter-zone facilitation; and 4) the resource reservation mechanism which manages to guarantee QoS requirements for sensitive applications in the inter-zone facilitation.

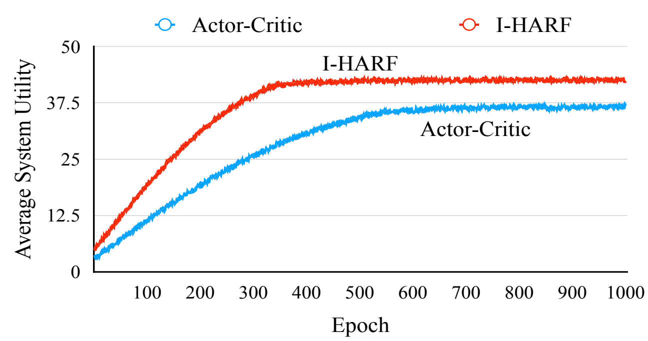


Fig. 16. Convergence of system utility in the typical scenario.

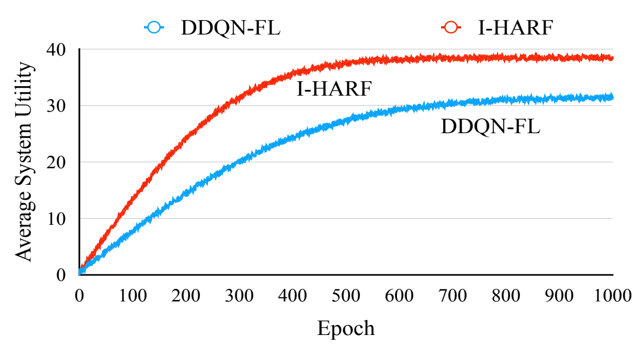


Fig. 17. Convergence of system utility in the complicated scenario.

VI. Conclusion

The paper has tackled the resource facilitation problem in Edge-IoT environment with consideration of outlandish environment situations. We proposed intelligent, adaptive and hierarchical *I-HARF* framework which comprises intra-zone and inter-zone resource facilitation schemes. Intra-zone facilitation adopts dynamic priority based model which is solved by novel ADCPG DRL scheme. Inter-zone facilitation exploits the intra-zone facilitation policy and situation awareness to develop resource facilitation scheme that is adaptive to the experienced mobility and outlandish situations. Both facilitations maximize the application and the edge utilities. In addition, *I-HARF* employs a reservation mechanism that exploit prediction to reserve resources for sensitive applications that cannot tolerate experienced delays due to the outlandish situations and consequence of coordination between ZEFs and IEF. Evaluation results demonstrate *I-HARF*'s capabilities including maximizing SPAR and system utility, guaranteeing latency for sensitive applications, and fast system convergence.

Acknowledgment

The work is supported by National Science Foundation (NSF) CNS core grant No. 1909520.

References

[1] NSF's 10 big ideas, NSF, [Online], available at: https://www.nsf.gov/news/special_reports/big_ideas/index.jsp.

[2] Fagui Liu, Zhenxi Huang, and Liangming Wang, "Energy-efficient collaborative task computation offloading in cloud-assisted edge computing for IoT sensors," Sensors 19.5: 1105, 2019.

[3] Y. Sun, M. Peng and S. Mao, "Deep Reinforcement Learning-Based Mode Selection and Resource Management for Green Fog Radio Access Networks," in IEEE Internet of Things Journal, vol. 6, no. 2, pp. 1960-1971, April 2019.

[4] C. Shu, Z. Zhao, Y. Han, G. Min and H. Duan, "Multi-User Offloading for Edge Computing Networks: A Dependency-Aware and Latency-Optimal Approach," in IEEE Internet of Things Journal, vol. 7, no. 3, pp. 1678-1689, March 2020.

[5] L. Huang, X. Feng, L. Zhang, et al., "Multi-Server Multi-User Multi-Task Computation Offloading for Mobile Edge Computing Networks," Sensors, 2019.

[6] H. Guo, J. Liu, J. Zhang, W. Sun and N. Kato, "Mobile-Edge Computation Offloading for Ultradense IoT Networks," in IEEE Internet of Things Journal, vol. 5, no. 6, pp. 4977-4988, Dec. 2018

[7] Li Lin, et al., "Computation offloading toward edge computing," Proceedings of the IEEE, 107.8: 1584-1607, 2019.

[8] Shiqiang Wang, et al., "Dynamic service migration in mobile edge computing based on markov decision process," IEEE/ACM Transactions on Networking, 27.3: 1272-1288, 2019.

[9] Duong Tuan Nguyen, et al., "Placement and chaining for run-time iot service deployment in edge-cloud," IEEE Transactions on Network and Service Management, 17.1: 459-472, 2019.

[10] Cagatay Sonmez, Atay Ozgovde, and Cem Ersoy, "Fuzzy workload orchestration for edge computing," IEEE Transactions on Network and Service Management 16.2: 769-782, 2019.

[11] Z. Ning, P. Dong, X. Kong and F. Xia, "A Cooperative Partial Computation Offloading Scheme for Mobile Edge Computing Enabled Internet of Things," in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 4804-4814, June 2019.

[12] J. Zhang et al., "Energy-Latency Tradeoff for Energy-Aware Offloading in Mobile Edge Computing Networks," in IEEE Internet of Things Journal, vol. 5, no. 4, pp. 2633-2645, Aug. 2018.

[13] H. Guo, J. Zhang, J. Liu and H. Zhang, "Energy-Aware Computation Offloading and Transmit Power Allocation in Ultradense IoT Networks," in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 4317-4329, June 2019.

[14] A. Abouaomar, S. Cherkaoui, Z. Mlika and A. Kobbane, "Resource Provisioning in Edge Computing for Latency Sensitive Applications," in IEEE Internet of Things Journal, 2021.

[15] B. Baek, J. Lee, Y. Peng and S. Park, "Three Dynamic Pricing Schemes for Resource Allocation of Edge Computing for IoT Environment," in IEEE Internet of Things Journal, vol. 7, no. 5, pp. 4292-4303, May 2020.

[16] P. Wang, C. Yao, Z. Zheng, G. Sun and L. Song, "Joint Task Assignment, Transmission, and Computing Resource Allocation in Multilayer Mobile Edge Computing Systems," in IEEE Internet of Things Journal, vol. 6, no. 2, pp. 2872-2884, April 2019.

[17] H. Rafique, M. A. Shah, S. U. Islam, T. Maqsood, S. Khan and C. Maple, "A Novel Bio-Inspired Hybrid Algorithm (NBIHA) for Efficient Resource Management in Fog Computing," in IEEE Access, vol. 7, pp. 115760-115773, 2019.

[18] Minghui Liwang, Ruitao Chen, and Xianbin Wang, 'Resource Trading in Edge Computing-enabled IoV: An Efficient Futures-based Approach," https://arxiv.org/abs/2101.00775, 2021.

[19] Vincent Francois-Lavet, Peter Henderson, Riashat Islam, Marc G. Bellemare, Joelle Pineau, "An Introduction to Deep Reinforcement Learning," in Foundations and Trends in Machine Learning, Volume 11, Issues 3-4, Now Publishers, 2018.

[20] X. Chen, H. Zhang, C. Wu, S. Mao, Y. Ji and M. Bennis, "Optimized Computation Offloading Performance in Virtual Edge Computing Systems Via Deep Reinforcement Learning," in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 4005-4018, June 2019.

[21] J. S. Pujol Roig, D. M. Gutierrez-Estevez and D. Gunduz, "Management and Orchestration of Virtual Network Functions via Deep Reinforcement Learning," in IEEE Journal on Selected Areas in Communications, vol. 38, no. 2, pp. 304-317, Feb. 2020.

[22] Xianfu Chen, et al., "Multi-tenant cross-slice resource orchestration: A deep reinforcement learning approach," IEEE Journal on Selected Areas in Communications 37.10: 2377-2392, 2019.

[23] J. Chen, S. Chen, Q. Wang, B. Cao, G. Feng and J. Hu, "iRAF: A Deep Reinforcement Learning Approach for Collaborative Mobile Edge Computing IoT Networks," in IEEE Internet of Things Journal, vol. 6, no. 4, pp. 7011-7024, Aug. 2019.

[24] Suzhi Bi, Liang Huang, Hui Wang, Ying-Jun and Angela Zhang. " Lyapunov-guided Deep Reinforcement Learning for Stable Online Computation Offloading in Mobile-Edge Computing Networks ", https://arxiv.org/abs/2010.01370, 2021.

[25] T. Alfakih, M. M. Hassan, A. Gumaei, C. Savaglio and G. Fortino, "Task Offloading and Resource Allocation for Mobile Edge Computing by Deep Reinforcement Learning Based on SARSA," in IEEE Access, vol. 8, pp. 54074-54084, 2020.

[26] G. Qiao, S. Leng, S. Maharjan, Y. Zhang and N. Ansari, "Deep Reinforcement Learning for Cooperative Content Caching in Vehicular Edge Computing and Networks," in IEEE Internet of Things Journal, vol. 7, no. 1, pp. 247-257, Jan. 2020.

[27] M. Zhang, S. Wang, and Q. Gao, "A joint optimization scheme of content caching and resource allocation for internet of vehicles in mobile edge computing," Journal of Cloud Computing 9.1: 1-12, 2020.

[28] C. Lin, D. Deng, Y. Chih and H. Chiu, "Smart Manufacturing Scheduling With Edge Computing Using Multiclass Deep Q Network," in IEEE Transactions on Industrial Informatics, vol. 15, no. 7, pp. 4276-4284, July 2019.

[29] Y. Liu, H. Yu, S. Xie and Y. Zhang, "Deep Reinforcement Learning for Offloading and Resource Allocation in Vehicle Edge Computing and Networks," in IEEE Transactions on Vehicular Technology, vol. 68, no. 11, pp. 11158-11168, Nov. 2019.

[30] L. Lei, H. Xu, X. Xiong, K. Zheng, W. Xiang and X. Wang, "Multiuser Resource Control With Deep Reinforcement Learning in IoT Edge Computing," in IEEE Internet of Things Journal, vol. 6, no. 6, pp. 10119-10133, Dec. 2019.

[31] M. Tang and V. W. S. Wong, "Deep Reinforcement Learning for Task Offloading in Mobile Edge Computing Systems," in IEEE Transactions on Mobile Computing, 2021.

[32] Li D, Xu S, Li P. Deep Reinforcement Learning-Empowered Resource Allocation for Mobile Edge Computing in Cellular V2X Networks. Sensors. 2021, 21(2):372.

[33] X. Xiong, K. Zheng, L. Lei and L. Hou, "Resource Allocation Based on Deep Reinforcement Learning in IoT Edge Computing," in IEEE Journal on Selected Areas in Communications, vol. 38, no. 6, pp. 1133-1146, June 2020.

[34] Y. Wei, F. R. Yu, M. Song and Z. Han, "Joint Optimization of Caching, Computing, and Radio Resources for Fog-Enabled IoT Using Natural Actor-Critic Deep Reinforcement Learning," in IEEE Internet of Things Journal, vol. 6, no. 2, pp. 2061-2073, April 2019.

[35] K. Liu and W. Liao, "Intelligent Offloading for Multi-Access Edge Computing: A New Actor- Critic Approach," ICC 2020 - 2020 IEEE International Conference on Communications (ICC), pp. 1-6, Dublin, Ireland, 2020.

[36] Fang Fu, Zhicai Zhang, Fei Richard Yu, Qiao Yan, "An actor-critic reinforcement learning-based resource management in mobile edge computing systems," International Journal of Machine Learning and Cybernetics, 11:18751889, 2020.

[37] Jia Yan, Suzhi Bi, and Ying-Jun Angela Zhang, "Offloading and Resource Allocation with General Task Graph in Mobile Edge Computing: A

Deep Reinforcement Learning Approach," IEEE Transactions on Wireless Communications, 2020.

[38] Y. He, Y. Wang, C. Qiu, Q. Lin, J. Li and Z. Ming, "Blockchain-Based Edge Computing Resource Allocation in IoT: A Deep Reinforcement Learning Approach," in IEEE Internet of Things Journal, vol. 8, no. 4, pp. 2226-2237, 15 Feb.15, 2021.

[39] H. Peng and X. Shen, "Deep Reinforcement Learning Based Resource Management for Multi-Access Edge Computing in Vehicular Networks," in IEEE Transactions on Network Science and Engineering, vol. 7, no. 4, pp. 2416-2428, 1 Oct.-Dec. 2020

[40] Y. Chen, Z. Liu, Y. Zhang, Y. Wu, X. Chen and L. Zhao, "Deep Reinforcement Learning-Based Dynamic Resource Management for Mobile Edge Computing in Industrial Internet of Things," in IEEE Transactions on Industrial Informatics, vol. 17, no. 7, pp. 4925-4934, July 2021.

[41] Haodong Lu, et al., "Edge QoE: Computation offloading with deep reinforcement learning for Internet of Things," IEEE Internet of Things Journal, 2020.

[42] Timothy P. Lillicrap, et al., "Continuous control with deep reinforcement learning," ICLR 2016.

[43] Ye Yu, Xin Li, and Chen Qian, "SDLB: A Scalable and Dynamic Software Load Balancer for Fog and Mobile Edge Computing," In Proceedings of the ACM Workshop on Mobile Edge Communications (MECOMM 2017), New York, NY, USA, 55-60. 2017.

[44] H. Flores et al., "Large-scale offloading in the Internet of Things," 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), pp. 479-484, Kona, HI, 2017.

[45] C. Zhang, H. Du, Q. Ye, C. Liu and H. Yuan, "DMRA: A Decentralized Resource Allocation Scheme for Multi-SP Mobile Edge Computing," 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), pp. 390-398, Dallas, TX, USA, 2019.

[46] Xuan-Qui Pham, et al., "Joint node selection and resource allocation for task offloading in scalable vehicle-assisted multi-access edge computing," Symmetry 11.1: 58, 2019.

[47] Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction," The MIT Press, 2nd edition, 2018.

[48] Tuomas Haarnoja, et al., "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,"arXiv preprint arXiv:1801.01290, 2018.

[49] Scott Fujimoto, Herke Van Hoof, and David Meger, "Addressing function approximation error in actor-critic methods," arXiv preprint arXiv:1802.09477, 2018.

[50] Qiang Yang, et al., "Federated machine learning: Concept and applications," ACM Transactions on Intelligent Systems and Technology (TIST) 10.2: 1-19, 2019.

[51] Kamil Ciosek and Quan Vuong and Robert Loftin and Katja Hofmann "Better Exploration with Optimistic Actor Critic" in proceedings of Advances in Neural Information Processing Systems (NeurIPS), Volume 32, 2019.

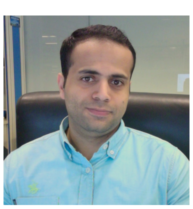
[52] Jrgen Schmidhuber, "Deep learning in neural networks: An overview", Neural Networks, Volume 61, 2015, Pages 85-117.

[53] Sepp Hochreiter and Jrgen Schmidhuber. 1997. "Long Short-Term Memory." Neural Comput. 9, 8 (November 15, 1997), 1735-1780.

[54] Amazon EC2 Instance Types. [Online]. Available: https://aws.amazon.com/ec2/instance-types.

[55] H. A. Alameddine, S. Sharafeddine, S. Sebbah, S. Ayoubi and C. Assi, "Dynamic Task Offloading and Scheduling for Low-Latency IoT Services in Multi-Access Edge Computing," in IEEE Journal on Selected Areas in Communications, vol. 37, no. 3, pp. 668-682, March 2019.

[56] Xiaofei Wang, et al., "In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning," IEEE Network 33.5: 156-165, 2019.



Ismail AlQerm is a postdoctoral research associate in the department of computer science at University of Missouri-Saint Louis (UMSL). He received his PhD in computer science from King Abdullah University of Science and Technology (KAUST) in 2017 and was among the recipients of KAUST Provost Award. His research interests include edge computing, resource allocation in IoT networks, developing machine learning techniques for resource allocation in wireless networks, and software defined radio prototypes. He is a member of IEEE and ACM.

Jianli Pan is currently an Associate Professor in the Department of Computer Science at the University of Missouri, St. Louis, MO USA. He obtained his Ph.D. and M.S. degrees from the Department of Computer Science and Engineering of Washington University in St. Louis, USA. He also holds a M.S. degree in Information Engineering from Beijing University of Posts and Telecommunications (BUPT), China. He is an associate editor for both IEEE Communication Magazine and IEEE Access. His current research interests include Internet of Things (IoT), edge computing, machine learning, cybersecurity, and smart energy.