Energy-Efficient Joint Optimization of Sensing and Computation in MEC-Assisted IoT Using Mean-Field Game

Runchen Xu, *Graduate Student Member, IEEE*, Zheng Chang, *Senior Member, IEEE*, Zhu Han, *Fellow, IEEE*, Sahil Garg, *Member, IEEE*, Georges Kaddoum, *Senior Member, IEEE*, and Joel J. P. C. Rodrigues, *Fellow, IEEE*

***Abstract*—Integrating multiaccess edge computing (MEC) with the Internet of Things (IoT) is able to provide IoT sufficient computational resources in addition to its capabilities of sensing and communication. In this article, given the limited computational and energy resources, IoT devices (IDs) are allowed to offload computational tasks to MEC servers for execution. However, as the number of IDs increases dramatically, jointly optimizing the usage of sensing, communication, and computational resources becomes challenging due to the exponential growth in interactions among the IDs. In this article, we address the energy-efficient joint optimization problem for sensing and computation in the MEC-assisted IoT system, aiming to ensure the freshness of the status update and minimize the energy consumption of IDs. To reduce the computation complexity, we introduce the concept of the general mean-field N-player Markov game (GMFG), and reformulate it as a mean-field game (MFG) with teams, leveraging the network structure of states. Considering the advantages of reinforcement learning (RL) for solving dynamic problems, we propose an MFG-based actor–critic algorithm (MFGAC) to minimize the long-term average system cost. Through extensive simulations, we demonstrate that the proposed method is effective and can outperform other schemes under different scenarios.**

***Index Terms*—Energy efficient, Internet of Things (IoT), mean-field game (MFG), multiaccess edge computing (MEC), reinforcement learning (RL).**

Manuscript received 1 May 2024; revised 8 July 2024; accepted 4 August 2024. Date of publication 14 August 2024; date of current version 20 November 2024. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62071105; in part by NSF under Grant CNS-2107216, Grant CNS-2128368, Grant CMMI-2222810, and Grant ECCS-2302469; in part by the U.S. Department of Transportation; in part by Toyota; in part by Amazon; in part by the Japan Science and Technology Agency (JST) Adopting Sustainable Partnerships for Innovative Research Ecosystem (ASPIRE) under Grant JPMJAP2326; in part by the Brazilian National Council for Scientific and Technological Development—CNPq under Grant 306607/2023-9; and in part by Horizon EU under Grant 101086159. *(Corresponding author: Zheng Chang.)*

Runchen Xu is with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: 202222080537@std.uestc.edu.cn).

Zheng Chang is with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China, and also with the Faculty of Information Technology, University of Jyväskylä, 40014 Jyväskylä, Finland (e-mail: zheng.chang@jyu.fi).

Zhu Han is with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77004 USA, and also with the Department of Computer Science and Engineering, Kyung Hee University, Seoul 446-701, South Korea (e-mail: zhan2@uh.edu).

Sahil Garg is with the Electrical Engineering Department, École de technologie supérieure, Montreal, QC H3C 1K3, Canada, and also with the Centre for Research Impact and Outcome, Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura 140401, India (e-mail: sahil.garg@ieee.org).

Georges Kaddoum is with the Electrical Engineering Department, École de technologie supérieure, Montreal, QC H3C 1K3, Canada, and also with the Cyber Security Systems and Applied AI Research Center, Lebanese American University, Beirut 03797751, Lebanon (e-mail: georges.kaddoum@etsmtl.ca).

Joel J. P. C. Rodrigues is with Amazonas State University, Manaus 69850, Brazil (e-mail: joeljr@ieee.org).

Digital Object Identifier 10.1109/JIOT.2024.3443701

I. INTRODUCTION

A. Background

RECENTLY, the research on Internet of Things (IoT) applications has received increasing interests, driven by the advancements in computation, communication, and sensing technologies [1], [2], [3] and the proliferation of IoT devices (IDs). To facilitate various IoT applications, including traffic surveillance, autonomous driving, and environmental monitoring, it is necessary to sense a variety of valid real-time information about the physical environment and transform it into state updates for future system operations. Meanwhile, as a metric for measuring the data freshness in IoT, the Age of Information (AoI), which assesses the currency of data within IoT applications, has been extensively studied and is integral to the functioning of communication systems [4], [5], [6], [7], [8], [9], [10]. However, the processing of sensory data is often constrained by the computational and temporal limitations inherent in IDs, which typically have restricted computing power, storage capability and battery capacity. To address these constraints, the emergence of multiaccess edge computing (MEC) has provided a potential solution by bringing networking, storage and computation capabilities closer to the network edge [9], [11], [12], [13]. By leveraging the computational resources of MEC servers, IDs can offload their computational tasks for efficient execution. Nonetheless, the increasing number of devices within the next-generation IoT system has resulted in an upsurge of interactions, both among devices and between devices and MEC servers [7]. As a result, the joint optimization of sensing and computation strategies becomes extremely complex, presenting a significant challenge in MEC-assisted IoT systems.

B. Related Work

Recently, the demand for effective and real-time information in various IoT applications has become increasingly stringent. IDs have the capability to perceive data from their physical surroundings and transmit both raw data and status updates to a centralized station. To assess the currency of data, the metric of AoI has been conceptualized, denoting the span of time elapsed since the latest state was updated [10]. Several research works have been conducted to optimize AoI in different contexts. For instance, Xu et al. [5] formulated a dynamic status update optimization problem with nonuniform time steps. The aim is to optimize the long-term average cost while balancing energy consumption and currency of data. Song et al. [14] developed a mathematical structure to evaluate the probability of AoI violation and the average AoI. They introduce a localized, adaptive power management strategy aimed at minimizing the overall average AoI across the ID network. Feng and Yang [15] focused on energy harvesting sensors and design an optimal state updating strategy to reduce long-term average AoI. Moreover, the sensed information typically exhibits temporal and spatial correlations in sensor monitoring networks [16]. Kim et al. [6] provided a definition and analysis of the concept of error-tolerable sensing (ETS) coverage, which is influenced by AoI. They formulate an optimization problem aimed at maintaining a specified standard of ETS coverage and minimizing the energy consumption.

The optimization of computational offloading emerges as a critical challenge, particularly in scenarios characterized by resource contention among IDs. In order to reduce the cumulative maximum latency of devices, Hu et al. [17] proposed a joint optimization approach that considers the user scheduling, drone flight paths, and offloaded tasks' distribution. Guo et al. [18] introduced the concept of digital twin (DT) into aerial computing networks, exploring the realm of resource allocation and intelligent unmanned aerial vehicle (UAV) deployment. Guo et al. [19] investigated the intricacies of joint resource allocation and task scheduling, aiming to optimize both the device-to-device (D2D) association and computation offloading strategies. To minimize the system-wide cost, which encompasses both data collection time and UAV energy consumption, Huang et al. [20] formulated a comprehensive joint optimization problem. The approach includes considerations, such as the UAV trajectory, sensing scheduling, communication power, sending power, and the number of time slots. Chen et al. [9] proposed a joint sensing and processing optimization problem aimed at reducing system cost, with a focus on maintaining data currency and energy efficiency. Additionally, Yun et al. [21] put forward an innovative approach by proposing a reinforcement learning (RL)-based algorithm for making joint sensing and computational decisions in edge computing (EC)-enabled wireless sensor networks.

In contrast to traditional optimization methods, RL approaches offer the capability to learn through environmental interactions, enabling continuous improvement of strategies. Moreover, multiagent RL (MARL) has been considered as efficient framework, which can address cooperative games across various domains. For instance, Oubbati et al. [22] considered each UAV as an agent and employ a multiagent deep Q-network (MADQN) to determine optimal trajectories. Wu et al. [23] introduced an MARL-based algorithm for trajectory optimization of multiple UAVs, with the objective to reduce the AoI of UAV data collection. However, the proliferation of IDs within IoT systems has resulted in a significant increase in computational complexity. With a rising count of agents in N-player scenarios, the interactions among these agents escalate exponentially. This exponential rise in interactions presents challenges in solving optimization problems and renders them intractable.

To reduce the complexity arising from interactions and computations, the concept of mean-field game (MFG) has emerged as a important tool in game theory. MFG investigates the collective behavior of all agents by incorporating the distribution of states across the agent population [24]. It has found widespread application in diverse domains, including biology, finance, computer networking, and industry. Guo et al. [25] introduced a general MFG (GMFG) framework for decision making. Zhou and Saad [7] leveraged an MFG framework and investigate the asymptotic performance of IDs in monitoring systems. To address the problem of minimizing AoI in grant-free (GF) communications, Zhang et al. [8] presented a mean-field evolutionary game-based scheme. Gu et al. [26] build an MFG MARL framework and provide computationally efficient algorithms. To reduce the system complexity, Kang et al. [27] redefined the classical computation offloading problem within an MFG framework, proposing a G-prox PDHG algorithm. Zhang et al. [28] developed an MG-aided MARL approach with low complexity to address the joint power and spectrum allocation in the presence of massive vehicle-to-vehicle connections.

C. Motivation and Contributions

As we can see, the integration of joint sensing and computation in IoT systems, specifically in terms of balancing energy consumption and AoI, is still under-investigation. Li et al. [29] optimized the sampling and processing offloading strategy to minimize the average AoI, yet they neglect the systemic implications of energy consumption. Chen et al. [9] put forward a multivariable iterative algorithm for minimizing system costs, aiming to optimize the overall system overhead. However, this approach may not be efficient for realistic dynamic environments. Furthermore, many existing approaches suffer from critical limitations, such as the significant increase in computational complexity with the number of IDs increases. For instance, aiming to improve the joint decision making for energy-restricted sensors within a network, Yun et al. [21] adopted the multiagent deep deterministic policy gradient (MADDPG) algorithm [30] for policy training. However, the computational complexity of this approach is $\mathcal{O}(N^2 \cdot |\mathcal{A}|)$, with N denoting the number of sensors and $\mathcal{A}$ representing the action space. As the agent population expands, the computational complexity grows significantly, complicating the resolution of optimization problem, especially in scenarios involving large arrays of sensors.

Motivated by the observations from the previous analysis, we investigate the joint sensing and computation optimization problem in a MEC-assisted IoT system. Within this system, IDs periodically conduct sensing tasks and have the option of local processing and data offloading. To mitigate the complexity associated with interaction and computation, we employ the MFG approach. This enables us to analyze the collective behavior of all IDs by utilizing the population state distribution, instead of observing each agent individually. Subsequently, we reformulate the general mean-field N-player Markov game (GMFG) into a networked Markov decision process (MDP) involving teams of agents. Focusing on minimizing both the long-term average energy consumption and long-term average AoI of IDs, we propose an MFG-based actor–critic algorithm (MFGAC). In this algorithm, the IDs take mean-field actions based on the state distribution, reducing computational complexity to $\mathcal{O}(|\mathcal{S}| \cdot |\mathcal{A}|)$, where $\mathcal{S}$ and $\mathcal{A}$ represent the state space and action space of the IDs, respectively.

To this end, the major contributions of this article can be summarized as follows.

1) We consider a system consisting of a MEC server and numerous IDs, formulating an energy-efficient joint optimization problem for sensing and computation. The aim of this optimization is to minimize the long-term average cost, which includes ensuring the freshness of status while minimizing the energy consumption of IDs.
2) To address the challenges associated with interaction and computation complexity, we present a comprehensive analysis of the GMFG, which focusing on exploiting the population state distribution. Subsequently, we reformulate the game as MFG with teams according to the network structure of states to further diminish computational complexity.
3) Considering the advantages of RL for solving dynamic problems, we introduce an MFGAC algorithm to solve the optimization problem. Notably, the computational complexity of MFGAC is $\mathcal{O}(|\mathcal{S}|\cdot|\mathcal{A}|)$ and is not affected by the size of ID participation. To the best of our knowledge, it is early attempt to employ an MFG-based RL approach to solve the joint sensing and computation optimization problem in such an IoT system.
4) We execute extensive simulations for the performance evaluation. The simulation results convincingly show the efficacy of our method, outperforming other benchmark schemes in various scenarios.

D. Organization

The reminder of this article is structured as follows. In Section II, we present the system model of the IoT system, along with a quantitative analysis of the original optimization problem. In Section III, we first introduce the GMFG and then reformulate it as an MDP with teams of agents. In Section IV, we present the proposed MFGAC with more details. The performance evaluation is shown in Section V. Finally, Section VI concludes the work.

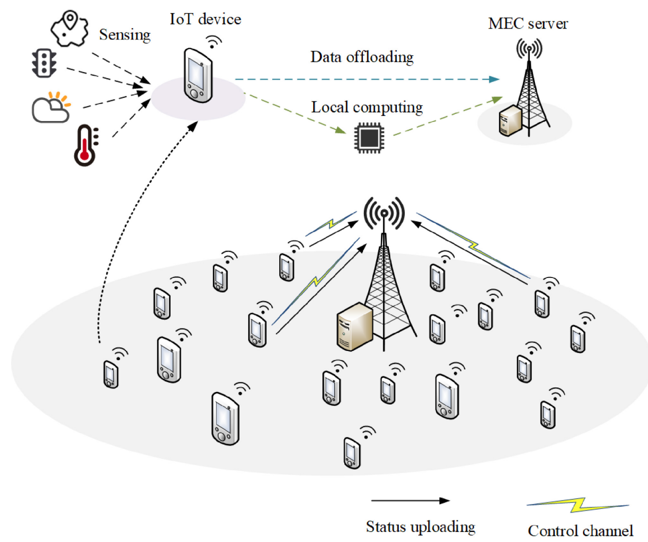

Fig. 1. System model.

II. System Model

A. Network Model

We consider a system comprising of a MEC server and N IDs, which is depicted in Fig. 1. The MEC server can offer computational resources to facilitate the computation tasks of the IDs. The set of the IDs is denoted as $\mathcal{I} = \{1, 2, \ldots, N\}$, where N represents the total number of the IDs. All IDs send their status information to the MEC server. The MEC server aggregates the information and solves the optimal joint sensing and computation policy, then sends the optimal policy back to the IDs over the control channel. Each device can periodically perform sensing operation to gather information about its surrounding environment, such as location, temperature, traffic signals, etc., at regular intervals with a period of T_s. For simplicity, we divide the timeline into T slots, denoted by $\mathcal{T} = \{1, 2, \ldots, T\}$. Additionally, we assume that the data size of information generated from a single sensing operation by the nth ID is quantified as D_n. In order to transform the sensed information into state updates for IoT applications, each ID can choose between offloading the sensed data to the MEC server for further processing or locally processing the task using its own processor and transmitting the updated status to the MEC server. The size of data associated with the updated status of the nth ID is indicated as D'_n, where $D_n > D'_n$. The key notations utilized are listed in Table I.

The whole process is shown in Fig. 2. At the beginning of each time slot, each ID performs one of the three operations: SO, SL, and I. We consider the nth ID adopting a mixed strategy $o_n(t) = o_n^1(t), o_n^2(t), o_n^3(t)$, which denotes the probabilities of choosing SO, SL, and I, respectively.

1) *Sensing and Data Offloading (SO)*: During a time slot, the ID senses information from its surrounding environment and transmits the raw information to the MEC server for processing. Upon successful data processing, the MEC server obtains the update status for IoT application.

TABLE I
NOTATIONS AND DEFINITIONS

Notations	Definitions
$\mathcal{I}$/ N	Set of IDs/ Number of IDs
T_s	Duration of a time slot
τ_n^S	Sensing time for n-th ID to perform a sensing task
D_n	Data size of data generated by a single sensing operation
e_n^S	Energy consumption of n-th ID for sensing a bit of data
M	Number of orthogonal wireless channels
p_n	Transmission power of n-th ID
h_n	channel fading gain of n-th ID
d	Distance between a ID and MEC
γ	Path loss exponent
N_0	background noise power
b_n	Bandwidth of n-th ID
τ_n^{UM}	Transmission latency for n-th ID to transmit the data D_n
E_n^{UM}	Energy consumption for n-th ID to transmit the data D_n
P_n^{UM}	Transmission success probability of data D_n
D'_n	Data size of updated status
τ_n^{UL}	Transmission latency for n-th ID to transmit the data D'_n
E_n^{UL}	Energy consumption for n-th ID to transmit the data D'_n
P_n^{UL}	Transmission success probability of data D'_n
η	Maximum number of transmissions in a time slot
τ_n^L	Local computation latency for processing the status update
f_n	CPU frequency of n-th ID
δ_n	Energy consumption coefficient of n-th ID
A_n	age of information of n-th ID
A_{th}	AoI violation threshold
z	Number of small time intervals in each time slot
τ_z	Duration of a small time interval

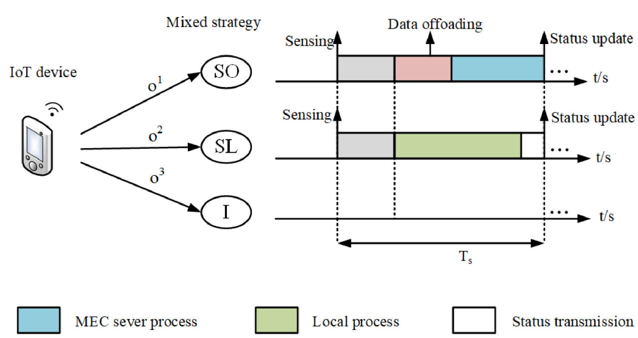

Fig. 2. Example of sensing and processing procedure.

2) *Sensing and Local Computation (SL)*: During a time slot, the ID senses information from its surrounding environment and performs local processing on the data. Once the computational tasks are processed locally, the ID transmits the obtained update status to the MEC server.
3) *Idle (I)*: During a time slot, considering the limited spectrum resources and the different AoI states of the IDs, an ID may choose to remain idle without engaging in data sensing, transmission, or processing operations.

B. Sensing Model

The IDs can sense raw data from the monitored environment at the beginning of each time slot. Let τ_n^S represent the duration required by the nth ID to accomplish a comprehensive sensing task, while D_n corresponds to the volume of data generated in a single sensing operation. Similar to [9], the energy consumption associated with the completion of a sensing process by the nth ID is given as

$$E_n^S = e_n^S D_n \tag{1}$$

where e_n^S represents the energy consumption per data bit for sensing by the nth ID. It is crucial to note that the data collected through the sensing operation is in its raw form and requires additional processing to obtain the updated status.

C. Transmission Model

We consider the system comprises M identical orthogonal wireless channel, each randomly assigned to IDs with transmission requests. When N' IDs send data with the same operation (e.g., SO or SL), the number of IDs selecting the same channel can be calculated as N'/M. Then, the signal-to-interference-plus-noise ratio (SINR) of this channel can be given as follows:

$$\Gamma = \frac{p_n h_n d_n^{-\gamma}}{N_0 + I} \tag{2}$$

where p_n is the transmission power, h_n represents the corresponding channel fading gain between the nth ID and the MEC server, and d_n denotes the distance between nth ID and the MEC server. γ represents the path-loss exponent, N_0 is the power of background noise, and $I = \sum_{i\in\chi} p_i h_i d_i^{-\gamma}$ accounts for the intercell interference. The set χ denotes the group of IDs utilizing the same wireless channel as the nth ID.

For the uplink transmission, the transmission rate of the nth ID is

$$\begin{aligned} R_n^U &= b_n \log_2(1+\Gamma) \\ &= b_n \log_2\left(1 + \frac{p_n h_n d_n^{-\gamma}}{N_0 + \sum_{i\in\chi} p_i h_i d_i^{-\gamma}}\right) \end{aligned}$$

where b_n is the bandwidth allocated to the nth ID.

In a time slot, if the nth ID chooses to perform an SO operation, it needs to transmit the collected raw data D_n to the MEC server. Consequently, the latency is calculated as

$$\tau_n^{\mathrm{UM}} = \frac{D_n}{R_n^U}. \tag{3}$$

Therefore, the energy consumption for transmitting the raw data is

$$E_n^{\mathrm{UM}} = p_n \tau_n^{\mathrm{UM}} = \frac{p_n D_n}{R_n^U}. \tag{4}$$

When the nth ID performs the SO operation, we can arrive at the following lemma for the successful transmission probability.

Lemma 1: When the nth ID offloads raw data to the MEC server, the successful transmission probability can be calculated as

$$P_S^{\mathrm{UM}} = \exp\left(-\frac{N_0 + \sum_{i\in\chi} p_i h_i d_i^{-\gamma}}{P_n d_n^{-\gamma}}\left(2^{\frac{D_n}{b_n \tau_{\mathrm{th}}^U}} - 1\right)\right) \tag{5}$$

where τ_{th}^U is the target transmission latency.

Proof: The MEC server can successfully receive transmissions from IDs only if the transmission delay is less than τ_{th}^{U}. Assuming the channel experiences Rayleigh fading, we denote the successful transmission probability as P_S^{UM}, which is derived as follows:

$$\begin{aligned}P_S^{\text{UM}} &= \mathbb{P}(\tau_n^{\text{UM}} \le \tau_{\text{th}}^{U}) \\ &= \mathbb{P}\left(\frac{D_n}{b_n \log_2\left(1+\frac{p_n h_n d_n^{-\gamma}}{N_0+\sum_{i\in\chi} p_i h_i d_i^{-\gamma}}\right)} \le \tau_{\text{th}}^{U}\right) \\ &= \mathbb{P}\left(h_n \ge \frac{N_0+\sum_{i\in\chi} p_i h_i d_i^{-\gamma}}{p_n d_n^{-\gamma}}\left(2^{\frac{D_n}{b_n \tau_{\text{th}}^{U}}}-1\right)\right) \\ &\overset{a}{=} \exp\left(-\frac{N_0+\sum_{i\in\chi} p_i h_i d_i^{-\gamma}}{p_n d_n^{-\gamma}}\left(2^{\frac{D_n}{b_n \tau_{\text{th}}^{U}}}-1\right)\right) \quad (6)\end{aligned}$$

where step (a) can be obtained utilizing the Rayleigh distribution of the channel gain, where $h_n \sim \exp(1)$, and recognizing that the random variables h_i for $i \in \chi$ are independent and exponentially distributed as $h_i \sim \exp(1)$. ■

Note that if the nth ID chooses SL, it generates an updated status D_n' after successfully processing the raw data D_n, where $D_n' < D_n$. Then, the ID needs to transmit the updated status data D_n' to the MEC server, and the transmission latency can be given by

$$\tau_n^{\text{UL}} = \frac{D_n'}{R_n^U}. \quad (7)$$

Then, the energy consumption for transmitting the update status is calculated as

$$E_n^{\text{UL}} = p_n \tau_n^{\text{UL}} = \frac{p_n D_n'}{R_n^U}. \quad (8)$$

The probability of successful transmission when the nth ID performs the SL operation is given in the following lemma.

Lemma 2: When the nth ID transmits the updated status data to the MEC server, the transmission success probability can be calculated as

$$P_S^{\text{UL}} = \exp\left(-\frac{N_0+\sum_{i\in\chi} p_i h_i d_i^{-\gamma}}{p_n d_n^{-\gamma}}\left(2^{\frac{D_n'}{b_n \tau_{\text{th}}^{U}}}-1\right)\right) \quad (9)$$

where τ_{th}^{U} is the target transmission latency.

Proof: The proof is similar to the one of Lemma 1, so we omit it here. ■

Moreover, since the transmission from the ID can may fail, we incorporate a retransmission mechanism that enables the ID to transmit up to η times. If the transmission fails η times, the IoD will discard the data and remain idle until new data is generated through sensing.

D. Computation Model

We present the computational model for status updates in the following. The computation offloading strategies employed by the IDs have an impact on the time and energy consumption required for performing the state update computation task.

1) Local Computation: If the nth ID uses its local processor to process the status update, the computation latency is

$$\tau_n^L = \frac{D_n}{f_n} \quad (10)$$

where f_n is the CPU frequency of the nth ID. Let δ_n denote the energy consumption coefficient specific to the chip used in the ID. The computational power of ID can be calculated as $P_n^L = \delta_n f_n^3$ [31], [32]. Then, the energy consumption of the nth ID for local processing can be expressed as

$$E_n^L = P_n^L \tau_n^L = \delta_n {f_n}^2 D_n. \quad (11)$$

2) Computation Offloading: If the nth ID chooses to offload its computational task, it needs to transmit data D_n to the MEC server for further processing. Then, the computation latency can be calculated as

$$\tau_n^M = \frac{D_n}{f_n^M} \quad (12)$$

where $f_n^M = (f^M/N_M)$ denotes the allocated computation resource of the nth ID. Here, f_M is the CPU frequency of the MEC server, and N_M is the number of IDs that choose to offload their raw data.

E. Age of Information at MEC Server

Since the information sensed by ID changes over time, the freshness of information at the MEC server significantly impacts monitoring and control aspects. In this section, we introduce the concept of AoI at the MEC server to evaluate the freshness of the status update.

The AoI at the MEC server is defined as the elapsed time since the reception of the last valid state update until the arrival of the subsequent valid state update at the MEC server. Let $\Delta_n(t) = 1$ represent the event that a status update from the nth ID is successfully received by the MEC server at time t; otherwise, $\Delta_n(t) = 0$. For the nth ID, the AoI at the MEC server can be expressed as

$$A_n(t+1) = \begin{cases} 1, & \text{if } \Delta_n(t) = 1 \\ \min\{A_n(t)+1, A_{\max}\}, & \text{if } \Delta_n(t) = 0 \end{cases} \quad (13)$$

where $A_{\max}$ is the upper bound of AoI. For real-time applications, the AoI higher than $A_{\max}$ means that the status is too outdated to be useful. Additionally, we assume that each time slot is divided into z small time intervals with a duration of τ_z, and all operations conclude at the end of the small time interval.

An example of the evolution curve of AoI with a series of operations is shown from Fig. 3. Without loss of generality, we assume the initial AoI is A_0, each time slot is divided into six small time intervals, and the threshold of AoI is A_{th}. In the kth slot, ID performs an SO operation, where it senses the data at the beginning of the time slot and then offloads the data to the MEC server. When the transmission fails η times, such as twice in this example, the data is discarded. In the $(k+1)$th slot, the ID performs another SO operation and successfully transmits the data to the MEC server after one retransmission. After the MEC server receives the data, it takes time τ^M to process the data and update the status. During sensing,

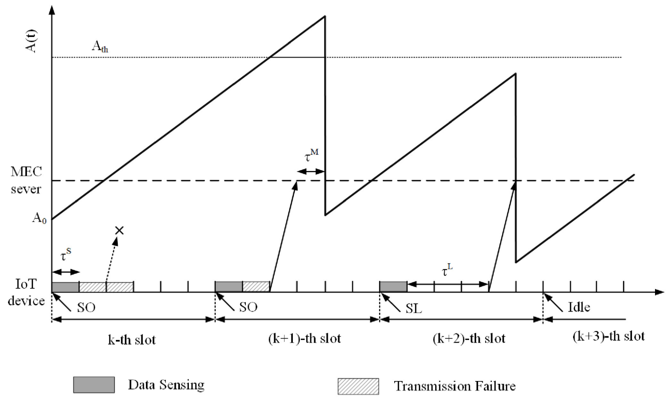

Fig. 3. Evolution curve of AoI.

transmission, and processing, the AoI increases linearly. Upon successful acceptance of a new status update by the MEC server, AoI is reset to a smaller value. In the $(k+2)$th slot, ID performs an SL operation, where it senses the data at the beginning of the time slot and process the data locally within a time duration τ^L. After the ID successfully transmits the processed data to the MEC server, the AoI at MEC server decreases. In the $(k+3)$th slot, the ID performs an Idle operation, where it becomes idle without sensing, transmitting, or processing data.

F. Cost Function and Problem Formulation

In this section, we analyze and derive the cost function for IDs, and then formulate an optimization problem to minimize the long-term average cost.

As mentioned above, the freshness of status updates received by the MEC server can serve as a performance metric for IDs, which is closely related to the AoI. As the AoI increases, the effectiveness of the updated state decreases. Therefore, IDs are penalized when the AoI is higher than a threshold A_{th}. We define the AoI violation cost as follows:

$$\mathcal{F}(A_n(t)) = \begin{cases} 0, & \text{if } A_n(t) \le A_{\text{th}} \\ \rho(A_n(t) - A_{\text{th}}), & \text{if } A_n(t) > A_{\text{th}} \end{cases} \tag{14}$$

where ρ is a vary large constant to ensure that the AoI of ID remains below the threshold.

Consequently, the AoI of the nth ID in time slot t is

$$C_n^A(t) = \int_0^{T_s} [A_n(t) + \mathcal{F}_n(A_n(t))]dt. \tag{15}$$

The average AoI of all IDs can be quantified as

$$\begin{aligned} \bar{C}_A(t) &= \frac{1}{N}\sum_{n=1}^{N} C_n^A \\ &= \frac{1}{N}\sum_{n=1}^{N}\int_0^{T_s} [A_n(t) + \mathcal{F}(A_n(t))]dt. \end{aligned} \tag{16}$$

Next, we analyze the energy consumption of IDs. In this system, we consider the nth ID adopting a mixed strategy $o_n(t) = \{o_n^1(t), o_n^2(t), o_n^3(t)\}$, which denotes the probabilities of choosing SO, SL, and I operations at time slot t, respectively. Each ID independently decides its mixed strategy at the start of each time slot. In time slot t, the energy consumption of the nth ID can be expressed as

$$C_n^E(t) = \bar{E}_n^{\text{UM}} o_n^1(t) + \left(E_n^L + \bar{E}_n^{\text{UL}}\right) o_n^2(t) \tag{17}$$

where $\bar{E}_n^{\text{UM}}$ is the expected transmission energy consumption of the ID when it chooses the SO operation, which can be expressed as

$$\begin{aligned} \bar{E}_n^{\text{UM}} = &\sum_{m=1}^{\eta} m E_n^{\text{UM}} \left(1 - P_S^{\text{UM}}\right)^{m-1} P_S^{\text{UM}} \\ &+ \eta E_n^{\text{UM}} \left(1 - P_S^{\text{UM}}\right)^{\eta} \end{aligned} \tag{18}$$

and $\bar{E}_n^{\text{UL}}$ is the expected transmission energy consumption of ID when it chooses the SL operation, which is

$$\begin{aligned} \bar{E}_n^{\text{UL}} = &\sum_{m=1}^{\eta} m E_n^{\text{UL}} \left(1 - P_S^{\text{UL}}\right)^{m-1} P_S^{\text{UL}} \\ &+ \eta E_n^{\text{UL}} \left(1 - P_S^{\text{UL}}\right)^{\eta}. \end{aligned} \tag{19}$$

Then, the average energy consumption cost of all IDs can be quantified as

$$\begin{aligned} \bar{C}_E(t) &= \frac{1}{N}\sum_{n=1}^{N} C_n^E(t) \\ &= \frac{1}{N}\sum_{n=1}^{N} [E_n^{\text{UM}} o_n^1(t) + \left(E_n^L + E_n^{\text{UL}}\right) o_n^2(t)]. \end{aligned} \tag{20}$$

In order to achieve a balance between the energy consumed by IDs and the average AoI encountered by IDs, we introduce a comprehensive measure for the decision-making process at a given time step t. This measure, denoted as the overall cost $C(t)$, can be formulated as follows:

$$C(t) = c_1 \bar{C}_A(t) + c_2 \bar{C}_E(t) \tag{21}$$

where c_1 and c_2 serve as parameters to normalize the equation.

Similar to [5], in this work, our optimization objective is to design dynamic strategies that minimize the long-term average cost, expressed as follows:

$$\begin{aligned} \mathcal{P}:\ &\min_{\boldsymbol{O}^T}\ J = \lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T} C(t)\right] \\ &\text{s.t. } \boldsymbol{O}^T = (\boldsymbol{O}_1, \boldsymbol{O}_2, \ldots \boldsymbol{O}_N) \\ &\qquad \boldsymbol{O}_n = (o_n(1), o_n(2), \ldots o_n(T)) \quad \forall n \in \mathcal{I} \end{aligned} \tag{22}$$

where $\boldsymbol{O}^T$ represents the set of strategies for IDs, and $\boldsymbol{O}_n$ denotes a sequence of strategies made by the nth ID from step 1 to step T.

III. MFG Reformulation

When the number of IDs is very large, solving the optimization problem becomes extremely challenging due to the exponential growth of interactions between IDs. In this article, we utilize the MFG to mitigate the complexity of interaction and computation. In this section, we initially present the GMFG. Then, we reformulate GMFG as an MFG with teams by exploiting the network structure of states.

A. GMFG

The MFG framework provides a robust paradigm for investigating the collective behavior of systems where the number of agents tends toward infinity. It uses the Fokker–Planck–Kolmogorov (FPK) equation and the Hamilton–Jacobi–Belman (HJB) equation to capture the dynamics and interactions of the system. The HJB equation characterizes the optimal condition for the system, while the FPK equation elucidates the system's dynamics. By modeling the interaction between individual agents within a vast network, the MFG approach reveals the emergent behavior of large-scale systems.

In this section, we present a comprehensive formulation of the MFG for classical N-player Markov games. We examine a system where N players share a finite state space $\mathcal{S}$ and select actions from a corresponding finite action space $\mathcal{A}$. The state space $\mathcal{S}$ is associated with a network represented by the graph $(\mathcal{S}, \mathcal{E})$, wherein $\mathcal{E} \subset \mathcal{S} \times \mathcal{S}$ represents the set of edges. These edges encompass all possible transition paths between states. For a given state $s \in \mathcal{S}$, the neighborhood $\mathcal{N}_{s_n^t}$ comprises nodes connected to s by a single edge, including s itself.

We assume that all agents are identical, indistinguishable and interchangeable. In the case of a large number of agents, we can consider the limit of the players' states $s^t = (s_1^t, s_2^t, s_3^t, \ldots, s_N^t)$ as a distribution of the population state. This distribution can be expressed as

$$\mu^t(s) = \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}_{\{s_n^t = s\}} \tag{23}$$

where $\mathbb{1}$ is an indicator function that returns 1 if the specified condition is true and 0 otherwise.

When the nth agent with state $s_n^t \in \mathcal{S}$ takes an action $a_n^t \in \mathcal{A}$ at time step t, it receives a reward defined as

$$r_n(s^t, a^t) = r\big(s_n^t, \mu^t(s), a_n^t\big), \quad n \in \mathcal{I} \tag{24}$$

where $r(s_n^t, \mu^t(s), a_n^t)$ denotes the reward function. The state then transitions to a neighboring state $s_n^{t+1} \in \mathcal{N}_{s_n^t}$, based on a transition probability defined as

$$s_n^{t+1} \sim P_n(s^t, a^t) = P\big(\cdot | s_n^t, \mu^t(s), a_n^t\big), \quad n \in \mathcal{I}. \tag{25}$$

Based on (24) and (25), the policy π_n^t is influenced by both the distribution of population states $\mu^t(s)$ and the state s_n^t. Consequently, the action a_n^t can be expressed as

$$a_n^t \sim \pi_n(s^t) = \pi(s_n^t, \mu^t(s)) \in \mathcal{P}(\mathcal{A}), \quad n \in \mathcal{I}. \tag{26}$$

Given the homogeneity of the agents, we can focus on a representative agent, allowing us to consider the following optimization problem:

$$\begin{aligned} \max \quad & V(s, \boldsymbol{\pi}, \boldsymbol{\mu}) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r(s^t, \mu^t, a^t) | s^0 = s \right] \\ \text{s.t.} \quad & s^{t+1} \sim P\big(s^t, \mu^t, a^t\big), \quad a^t \sim \pi^t\big(s^t, \mu^t\big) \end{aligned} \tag{27}$$

where $\boldsymbol{\pi} := \{\pi^t\}t = 0^T$ represents the sequence of policies, $\boldsymbol{\mu} := \{\mu^t\}t = 0^T$ denotes the flow of distributions, $\gamma \in (0, 1)$ is a discount factor, and s^0 is the initial state. At each time step t, the representative agent, following policy π^t, chooses an action a^t and consequently gets a reward determined by the function $r(s^t, \mu^t, a^t)$. The agent's state then evolves through $P(\cdot | s_n^t, \mu^t(s), a_n^t)$.

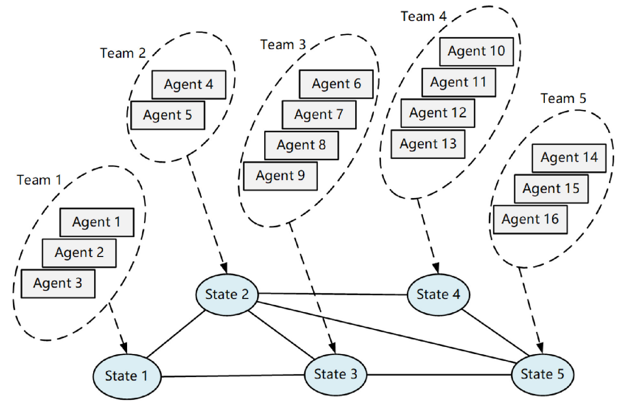

Fig. 4. Example of MFG with teams.

In the GMFG, N agents participate by observing the population state distribution. Each agent has an observation space of size $|\mathcal{S}|$ and action space of size $|\mathcal{A}|$. Consequently, the Nash equilibrium, representing the solution of the game, can be derived as follows.

Definition 1: Let $\boldsymbol{\mu}^*$ be the optimal distribution flow and $\boldsymbol{\pi}^*$ be the optimal policy sequence. The Nash equilibrium can be defined as follows:

$$V\big(s, \boldsymbol{\pi}^*, \boldsymbol{\mu}^*\big) \geq V\big(s, \boldsymbol{\pi}, \boldsymbol{\mu}^*\big) \tag{28}$$

$$V\big(s, \boldsymbol{\pi}^*, \boldsymbol{\mu}^*\big) \geq V\big(s, \boldsymbol{\pi}^*, \boldsymbol{\mu}\big). \tag{29}$$

According to this definition, that value V with the optimal policy sequence $\boldsymbol{\pi}^*$ cannot be increased by changing to another policy sequence $\boldsymbol{\pi}$ when the distribution flow and the initial state s is fixed. Furthermore, when the policy sequence and the initial state s is fixed, the value V can only be maximized if the distribution flow is $\boldsymbol{\mu}^*$.

B. MFG With Teams

In this section, we present how to reduce computational complexity by reformulating the GMFG as a networked MDP involving teams of agents. Fig. 4 illustrates an example of an MFG with teams.

The primary objective is to capitalize on the inherent homogeneity among agents within the MFG framework and strategically regroup these N agents based on their respective states. This reorganization can transform the original game, involving N agents, into a networked MDP characterized by $|\mathcal{S}|$ teams of agents, each indexed according to its state.

For a team with state s, it consists of $N \cdot \mu(s)$ agents. The nth agent with state s takes action a_n based on the policy $\pi(s_n^t, \mu^t(s)) \in \mathcal{P}(\mathcal{A})$. Therefore, we can use $m(s) \in \mathcal{P}^{N \cdot \mu(s)}(\mathcal{A})$ to represent the empirical action distribution of the action set $\mathcal{A}_s' = \{a_1, a_2, \ldots, a_{N \cdot \mu(s)}\}$ for the team with state s.

At each time step $t = 0, 1, \ldots, T$, the team with state s takes action set $h^t(s)$ according to a team policy $\Pi_s(m^t(s) | \mu^t(s))$.

Then, the team with state s will receive a team reward $r_s(\mu^t(s), m^t(s))$, which can be expressed as

$$r_s(\mu^t(s), m^t(s)) = \sum_{a \in \mathcal{A}} r(s, \mu^t(s), a) m(s)(a), \quad s \in \mathcal{S} \quad (30)$$

where $m(s)(a)$ represents the proportion of agents in state s that take action a.

The team policy $\Pi_s(m^t(s)|\mu^t(s))$ denotes the probability that a appears $N \cdot \mu^t(s) m^t(s)(a)$ times in $\mathcal{A}'_s$, where $a \in \mathcal{A}$ and $a_i \sim^{i.i.d} \pi(s_n^t, \mu^t(s))$. This probability can be found using a polynomial distribution, given by

$$\Pi_s(m^t(s)|\mu^t(s)) = \frac{(N \cdot \mu^t(s))!}{\prod_{a \in \mathcal{A}} \varphi!} \prod_{a \in \mathcal{A}} (\pi(s, \mu(s))(a))^{\varphi} \quad (31)$$

where $\varphi = N \cdot \mu^t(s) m^t(s)(a)$.

Now, we focus on all the teams. Given the distribution of population states $\mu^t \in \mathcal{P}(\mathcal{S})$, we can express the set of observed action distributions for all states as

$$m^t \sim \Pi(m^t|\mu^t) = \prod_{s \in S} \Pi_s(m^t(s)|\mu^t(s)). \quad (32)$$

After each step, the population state distribution μ^t will transition to μ^{t+1}. From the perspective of the teams, the transition probability described in (25) can be interpreted as a Markov process, characterized by the induced transition probability denoted as P^N. Therefore, we have

$$\mu^{t+1} \sim P^N(\cdot|\mu^t, m^t) \in \mathcal{P}(\mathcal{S}) \quad (33)$$

which corresponds to the FPK equation in MFG.

According to (30), the mean-field reward over all teams can be expressed as

$$r^t(\mu^t, m^t) = \sum_{s \in \mathcal{S}} r_s(s, \mu^t(s), m^t(s)). \quad (34)$$

Consider the population distribution flow $\boldsymbol{\mu} := \{\mu^t\}_{t=0}^T$, the policy sequence $\boldsymbol{\Pi} := \{\Pi(m^t|\mu^t)\}_0^T$, and a discount factor $\gamma \in (0, 1)$. The maximum expected cumulative award for all teams is expressed as follows:

$$\begin{aligned} V(\boldsymbol{\mu}) &= \max_{\boldsymbol{\Pi}} \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r^t(\mu^t, m^t)|\mu^0 = \mu\right] \\ &= \max_{\boldsymbol{\Pi}} \mathbb{E}\left[\sum_{t=0}^{T} \sum_{s \in \mathcal{S}} \gamma^t r_s(s, \mu^t, m^t)|\mu^0 = \mu\right] \quad (35) \end{aligned}$$

which corresponds to the HJB equation in MFG [33].

In contrast to the GMFG, where there are N agents in the game, the MFG with teams focuses on a simplified scenario with a single representative agent. In this setup, the representative agent has an observation space of size $|\mathcal{S}|$ and an action space of size $|\mathcal{S}| \cdot |\mathcal{A}|$. It can simply the game, enabling more efficient computation and analysis while still capturing the fundamental dynamics and interactions of the original game.

IV. Proposed MFGAC Algorithm

In this section, we present a reformation of the problem of minimizing long-term average costs in the IoT system as an MFG problem. Subsequently, we introduce a novel actor–critic algorithm based on MFG, integrating RL and MFG principles, to address the optimization problem.

A. Problem Reformulation Based on the Mean Field

In this section, we reformulate the initial optimization problem as an MFG with teams, which can be subsequently transformed into a single-agent RL problem. To accomplish this, we have the following definitions.

1) *Mean-Field Observation:* $\mu^t \in \mathcal{P}(\mathcal{S})$. Let s^t represent the AoI state of IDs at the beginning of the tth time slot. The state s^t can take values in the set $\mathcal{S} = \{0, 1, 2, \ldots, A_{\max}\}$. Then, the mean-field observation $\mu^t(s)$ is the distribution of states of all IDs and can be calculated as follows:

$$\mu^t(s) = \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}_{\{s_n^t = s\}} \quad (36)$$

where $\mathbb{1}$ is an indicator function which returns 1 if the given condition is true and 0 otherwise.

2) *Mean-Field Action:* $m^t(s) \in \mathcal{P}^{N \cdot \mu(s)}(\mathcal{A})$. Each ID takes an action at the beginning of each time slot. Let the action a^t represent the operation performed by IDs in the tth time slot, which is denoted as $a^t \in \mathcal{A} = \{\text{SO}, \text{SL}, \text{I}\}$. Then, the mean-field action can be expressed as

$$m^t(s)(a) = \frac{\sum_{n=1}^{N} \mathbb{1}_{\{s_n^t = s, a_n^t = a\}}}{\sum_{n=1}^{N} \mathbb{1}_{\{s_n^t = s\}}} \quad (37)$$

which represents the proportion of IDs with state s that perform operation a.

3) *Mean-Field Reward:* $r(\mu^t, m^t)$. The mean-field reward $r(\mu^t, h^t)$ represents the reward received by all the IDs at the tth time step. In the optimization problem, the objective is to maximize the reward, which is equivalent to minimizing the long-term average cost. Then, the mean-field reward can be given by

$$r(\mu^t, m^t) = -\sum_{s \in \mathcal{S}} \mu^t(s) l^t(s, m) \quad (38)$$

where $l^t(s, m)$ is the cost of all IDs with state s at the tth time slot.

By reformulating the original cost function into mean-field form, we can obtain the $l^t(s, m)$. According to the previous analysis, the average system cost of IDs at time slot t can be given as

$$\begin{aligned} \bar{C}(t) &= \mathbb{E}_{s \sim \mu}\left[\int_0^{T_s} c_1[A(t) + \mathcal{F}_n(A_n(t))]dt \right. \\ &\quad \left. + c_2\left[\bar{E}_n^{\text{UM}} o_n^1(t) + \left(E_n^L + \bar{E}_n^{\text{UL}}\right) o_n^2(t)\right]\right] \\ &= \sum_{s \in \mathcal{S}} \mu^t(s) \left(\int_0^{T_s} c_1[A + \mathcal{F}_n(A)]dt\right. \end{aligned}$$

$$+ c_2\big[E_n^{\mathrm{UM}} m^t(s)(SO) + \big(E_n^L + E_n^{\mathrm{UL}}\big) m^t(s)(SL)\big]\Big)$$
$$= \sum_{s\in\mathcal{S}} \mu^t(s) l^t(s,m) \tag{39}$$

where

$$l^t(s,m) = \int_0^{T_s} c_1\varphi[A + \mathcal{F}_n(A)]dt + c_2\big[E_n^{\mathrm{UM}} m^t(s)(SO) + \big(E_n^L + E_n^{\mathrm{UL}}\big) m^t(s)(SL)\big]. \tag{40}$$

At each time slot t, the teams select their actions m^t according to the policy $\Pi(m^t|\mu^t)$ at each time slot t. The population then receives a reward $r(\mu^t, m^t)$, and the state distribution μ^t evolves through $P^N(\cdot|\mu^t, m^t)$. The objective of this mean-field system is to find a sequence of policies $\mathbf{\Pi} := \{\Pi(m^t|\mu^t)\}_0^T$ that maximizes the expected discounted accumulated mean-field reward. It is given as

$$V(\mu) = \max_{\mathbf{\Pi}} \ \mathbb{E}\Bigg[\sum_{t=0}^{T} \gamma^t r\big(\mu^t, m^t\big)|\mu^0 = \mu\Bigg]$$
$$\text{s.t.} \ \ \mu^{t+1} \sim P^N\big(\cdot|\mu^t, m^t\big), \quad m^t \sim \Pi(m^t|\mu^t) \tag{41}$$

where $\gamma \in (0,1)$ represents a discount factor, and μ^0 represents the initial state distribution.

B. Value Function and Q Function

According to (41), the value function of the population can be calculated as

$$V_\Pi(\mu) = \mathbb{E}\Bigg[\sum_{t=0}^{\infty} \gamma^t r(\mu^t, h^t)|\mu^0 = \mu\Bigg]. \tag{42}$$

To evaluate the effect of taking a mean-field action h with state distribution μ, we make use of the state–action-value function. The Q value follows the Bellman equation in RL. The Bellman optimality equation for Q value can be expressed as

$$Q^*(\mu, m) \ = r(\mu, m) \ + \gamma\mathbb{E}[V_{\Pi^*}(\mu')] \tag{43}$$

where $r(\mu, m)$ is the reward for taking mean-field action m at mean-field observation μ, and μ' is the mean-field observation of the next time step.

Then, the value function can be defined as the expected of the Q value, which can be written as

$$V_\Pi(\mu) = \mathbb{E}_h[Q(\mu, m)] = \sum_m \Pi(m|\mu)Q(\mu, m) \tag{44}$$

where the Q function can be updated according to

$$Q^{t+1}(\mu, m) \ = (1-\eta)Q^t(\mu, m) \ + \eta[r^t + \gamma \max_{m'} Q^t(\mu', m')] \tag{45}$$

where η is the learning rate.

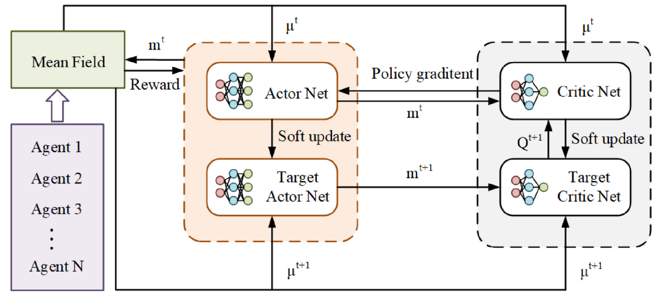

Fig. 5. Framework of the proposed MFGAC algorithm.

C. MFGAC Algorithm Design

To address the RL problem in the MFG setting, we propose the MFGAC algorithm, which is illustrated in Fig. 5.

Within the framework of the MFG algorithm, all individual agents are replaced by a population that takes mean-field actions and receives instantaneous rewards. The states of all agents are transformed into a mean-field according to (36). Similar to the DDPG algorithm [34], our algorithm consists of four networks: 1) the actor network; 2) the critic network; 3) the target actor network; and 4) the target critic network.

To obtain the optimal mean-field action h^t from the mean-field observation μ^t, we introduce the actor network $f(\mu|\theta)$, which is a neural network parameterized by θ. The actor network aims to find the action sequence $\boldsymbol{m} := \{m^t\}_{t=1}^T$ that minimizes the long-term average cost. The actor network parameters can be updated using the sampled policy gradient, which is calculated as

$$\nabla_{\theta^f} J = \frac{1}{M}\sum_j \nabla_h Q\big(\mu^j, m|\omega\big)|_{m=f(\mu^j)} \nabla_\theta f(\mu^j|\theta) \tag{46}$$

where M is the batch size.

Furthermore, the mixed strategies of agents can be got from the actor network. The actor network consider a strategy based on the softmax operation

$$\Pi_s(m(s)(a)|\mu(s)) = \frac{\exp[\beta f_{s,a}(\mu|\theta)]}{\sum_{a\in\mathcal{A}} \exp[\beta f_{s,a}(\mu|\theta)]} \tag{47}$$

where β represents the temperature parameter governing the softmax operation [26].

The critic network $Q(\mu, m|\omega)$ is a neural network parameterized by ω. It is utilized to estimate the Q value based on the mean-field observation and action. Furthermore, the critic network's parameter can be updated by minimizing its loss. Analogous to the DQN algorithm [35], the critic network's loss function is a TD-error, expressed as

$$L = \frac{1}{M}\sum_j (y_j - Q(\mu^j, m^j|\omega)) \tag{48}$$

where y_j is the target Q value, which can be obtained from the Bellman optimization equation. To ensure stability during the critic network's learning process, we introduce a target critic network Q' with parameter ω'. The target Q value can be calculated as

$$y_j = r^j + \gamma Q(\mu^{j+1}, f'(\mu^{j+1}|\theta')|\omega') \tag{49}$$

Algorithm 1 MFGAC Algorithm

Initialize: Randomly initialize actor network $f(\mu|\theta)$ and critic network $Q(\mu, m|\omega)$ with parameters θ and ω. Initialize target actor network f' and target critic network Q' with weights θ' and ω'. Initialize replay buffer R.
for episode $= 1, 2, \ldots, L$ **do**
  Receive an initial state distribution μ^0;
  Initialize the noise process $\mathcal{N}$ for action exploration;
  for step $t = 1, 2, \ldots, T$ **do**
    Choose a mean field action $m^t = f(\mu^t|\theta) + \mathcal{N}$ based on the noise process and the actor network;
    The population takes the mean field action m^t and gets a mean field reward $r^t(\mu^t, m^t)$;
    Update to environment, and the state distribution μ^t evolves to μ^{t+1};
    Store transition $(\mu^t, m^t, r^t, \mu^{t+1})$ in the replay buffer R;
    Randomly sample a batch of M transitions $(\mu^j, m^j, r^j, \mu^{j+1})$ from R;
    Calculate the target Q value:
      $y_j = r^j + \gamma Q(\mu^{j+1}, f'(\mu^{j+1}|\theta')|\omega')$;
    Update the critic network by minimizing the loss:
      $L = \frac{1}{M}\sum_j (y_j - Q(\mu^j, m^j|\omega))$
    Update the actor network by using the sampled policy gradient:
      $\nabla_{\theta^f} J = \frac{1}{M}\sum_j \nabla_h Q(\mu^j, m|\omega)|_{m=f(\mu^j)} \nabla_\theta f(\mu^j|\theta)$;
    Soft update the target actor network parameter:
      $\theta' \leftarrow \alpha\theta + (1-\alpha)\theta'$;
    Soft update the target critic network parameter:
      $\omega' \leftarrow \alpha\omega + (1-\alpha)\omega'$;
  end for
end for

where f' is the target actor network, which is a neural network with parameter θ'.

The pseudocode of MFGAC is presented in Algorithm 1. Initially, the replay buffer and all neural networks are initialized. The total number of training episodes for the neural network is denoted by L. At the beginning of each episode, the system receives an initial state distribution μ^0 and initializes the noise process $\mathcal{N}$ for action exploration.

During each episode, the mean-field interacts with the environment for T steps. At the beginning of each step, the population takes a mean-field action m^t according to the noise process, the actor network and the current mean-field observation μ^t. Subsequently, the population gets a mean-field reward according to (38), and the state distribution evolves to μ^{t+1}. We utilize a replay buffer R to store transitions $(\mu^t, m^t, r^t, \mu^{t+1})$, and during the training process, a random batch of M transitions is sampled from this buffer. The critic network is then updated by minimizing the loss function defined in (48). After updating the critic network, the actor network is then updated using the sampled policy gradient, which is computed as (46).

At the end of each step, a soft update method [34] is applied to update the target actor network parameter θ' and the target critic network parameter ω', as follows:

$$\theta' \leftarrow \alpha\theta + (1-\alpha)\theta', \quad \omega' \leftarrow \alpha\omega + (1-\alpha)\omega' \tag{50}$$

where $\alpha \in (0, 1)$ is a parameter that controls the slow update for the target network.

TABLE II
PARAMETER SETTING IN THE SIMULATION

Parameters	**Values**
The number of IDs N	$[50, 500]$
CPU frequency of the MEC sever f^M	2 GHz
Data size of sensed information D	$[3.6, 4.6, 5.6]$ kbit
Noise power N_0	-110 dBm
Data size of updated status D'	1.3 kbit
Path loss exponent γ	4
Transmission power of ID p_n	0.1W
Distance between IDs and sever d_n	150 m
Bandwidth of each channel b_n	$[0.2, 0.9]$ MHz
Maximum number of transmissions in a time slot η	2
Duration of a small time interval τ_z	10 ms
Energy consumption for sensing per bit of data e_n^S	4.3×10^{-7}
CPU frequency of n-th ID f_n	0.2 MHz
Energy consumption coefficient of ID δ_n	2.7×10^{-17}
Duration of a time slot T_s	$[60, 70, 80]$ ms

D. Complexity Analysis

Low computational complexity is critical to effectively deploying IoT systems. In MFGAC, all IDs are simplified as a single representative agent. This agent needs to observe the state distribution of all IDs and perform a mean-field action $m^t(s)(a)$ at the beginning of each time slot. The size of observation space and the action space for the representative agent are $|\mathcal{S}|$ and $|\mathcal{S}| \cdot |\mathcal{A}|$, respectively. Therefore, the computational complexity of MFGAC can be presented as $\mathcal{O}(|\mathcal{S}| \cdot |\mathcal{A}|)$, which is independent of the number of IDs.

V. PERFORMANCE EVALUATIONS

We present the evaluation of the proposed method in this section. The relevant system parameters are referred to [6], [9], and [21] and are listed in Table II. For the neural network training process, we employ fully connected deep neural networks (DNNs) for both the actor network and the critic network. Each network has two hidden layers, each comprising 128 nodes. In addition, we set the number of time steps per episode to 100, the batch size M to 64, the maximum number of training episodes L to 350, the learning rate η to 0.001, the replay buffer size to 10 000, the soft update parameter α to 0.005, and the discount factor γ to 0.98.

We compare our proposed scheme with DDPG and MAPPO [36] to validate its convergence and effectiveness. Then, we measure the performance of our algorithms by comparing them to four established baselines: 1) all edge computation offload (AECO); 2) all local computation (ALC); 3) MFG-based strategy with SO and I (MFG-OI); and 4) MFG-based strategy with SL and I (MFG-LI). After that, we assess the impact of several factors, including the AoI threshold, the number of wireless channel, the bandwidth of the channel, and the data size of sensed information. Finally, we analyze

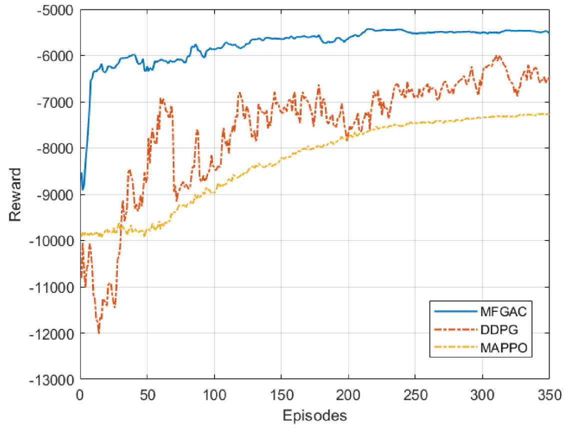

Fig. 6. Convergence comparison.

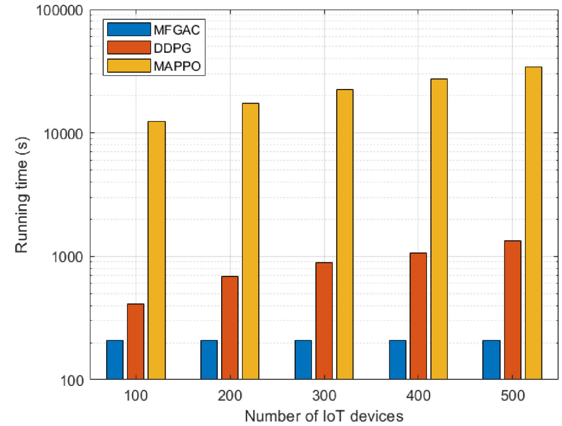

Fig. 7. Running time versus number of IDs.

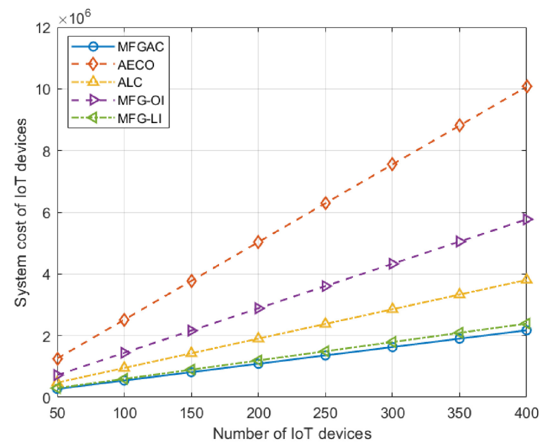

Fig. 8. System cost of IDs versus number of IDs.

the tradeoff between the energy consumption cost and the AoI cost in the IoT system.

1) *AECO:* Within the framework of the AECO, IDs sense environmental information at the beginning of each time slot and then offload the sensed data to the MEC server for processing.
2) *ALC:* Within the framework of the ALC, IDs sense environmental information at the beginning of each time slot and process it locally. Then, they transmit the update status to the MEC server.
3) *MFG-OI:* Within the framework of the MFG-OI, IDs make MFG-based RL strategies, and the sensed information is computed at the MEC server.
4) *MFG-LI:* Within the framework of the MFG-LI, IDs make MFG-based RL strategies, and the sensed information is locally computed.

Fig. 6 illustrates the reward achieved by each algorithm as the number of episodes increases. Notably, our MFGAC demonstrates convergence when the number of episodes reaches approximately 250. In contrast, DDPG does not achieve stability and fails to converge to the optimal solution. In addition, the MAPPO is stable but converges slowly. It is obvious that MFGAC shows a faster convergence rate and obtains a higher reward compared to the other two algorithms.

Fig. 7 shows the running time of each algorithm as the number of IDs increases. The results demonstrate that the running time of our proposed MFGAC is approximately 210 s and remains relatively constant as the number of IDs increases. In contrast, the running times of DDPG and MAPPO are significantly higher and increase exponentially with the number of IDs. This discrepancy arises from the lower computational complexity of MFGAC compared to the other algorithms. Unlike DDPG and MAPPO, the algorithmic complexity of MFGAC is independent of the number of IDs. According to the performance results, it can be concluded that the proposed scheme for MFG-based sensing and computation is not only effective but also highly time efficient.

Fig. 8 illustrates the system cost of different methods as the number of IDs varies. First, in all methods, the system cost of IDs increases as the number of IDs increases. This is because as the number of IDs increases, the available spectrum resources become more constrained, resulting in a decrease in the successful transmission probability of IDs. Consequently, the average AoI of the IDs increases as their successful transmission probability decreases, leading to an overall rise in the system cost of the IDs. Furthermore, our proposed MFGAC outperforms others, showcasing a reduction in system cost compared to AECO by 78.41%, ALC by 42.91%, MFG-OI by 62.33%, and MFG-LI by 9.04%. Fig. 9 shows the average system cost of IDs with different methods as the λ increases. Note that N/M indicates the radio of the number of IDs to the number of channels. It can be seen that the average cost increases with N/M in all methods. This is due to the fact that as N/M increases, spectrum resources become scarce, leading to a decrease in the successful transmission probability of IDs. Moreover, compared to the others, MFGAC achieves the lowest average system cost as λ increases. Therefore, our MFGAC perform a more rational joint sensing and computation strategy.

Fig. 10 illustrates the average AoI at each step, considering varying AoI thresholds (A_{th}) as the number of steps progresses. The entire process is segmented into four stages, wherein distinct initial state distributions are observed at steps 0, 40, 60, and 80, correspondingly. As depicted in Fig. 10, the average AoI exhibits an upward trend as the A_{th} increases. When a

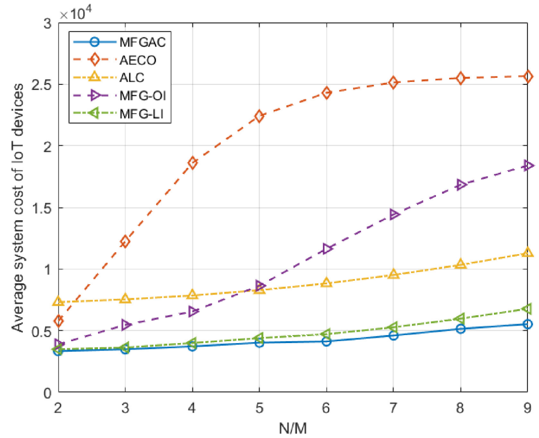

Fig. 9. System cost of IDs versus N/M.

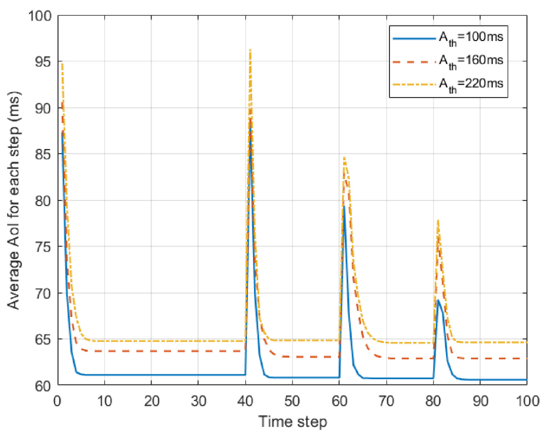

Fig. 10. Average AoI versus time step.

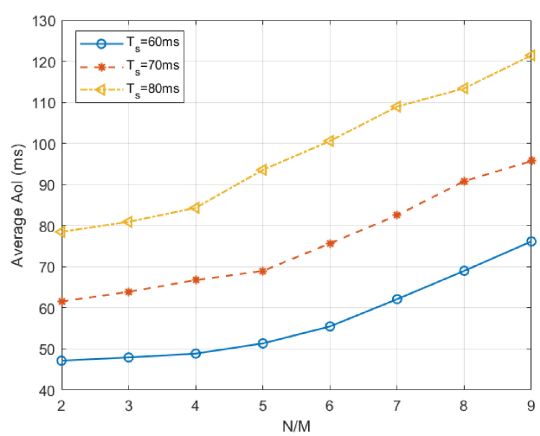

Fig. 11. Average AoI versus N/M.

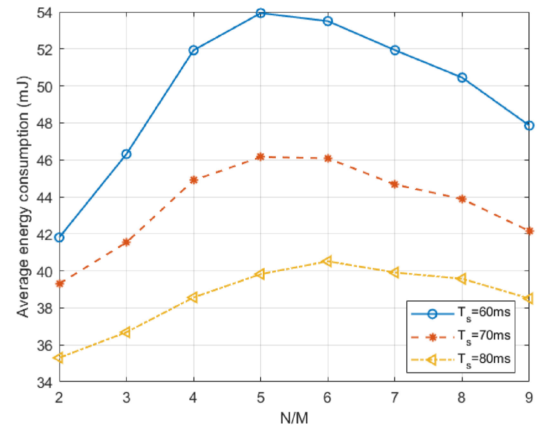

Fig. 12. Average energy consumption versus N/M.

new initial state distribution arrives, the state distribution of the population can evolve by taking the mean-field action obtained through MFGAC, which leads to a rapid decrease in the average AoI.

Fig. 11 demonstrates the influence of N/M on the average AoI for IDs at various T_s. The results indicate a positive correlation between the average AoI and the increase in N/M. It can be attributed to the decreased number of available wireless channels as N/M increases, resulting in a reduced transmission success probability for IDs when transmitting data to the MEC server. Consequently, IDs with lower AoI are more inclined to engage in the I operation, resulting in an overall rise in the average AoI. Furthermore, the average AoI of IDs shows an upward trend with the augmentation of T_s. This is because a longer sensing period T_s necessitates a prolonged waiting duration for each ID before transmitting raw data or updating its status to the MEC server.

Fig. 12 shows the average energy consumption of IDs across different values of N/M and the sensing period T_s. The observed trend reveals an initial increase and subsequent decrease in average energy consumption as N/M grows. For example, when T_s is 60 ms, the average energy consumption first increases then begin to decrease when N/M more than 5. This behavior can be attributed to the energy consumption associated with local computation and subsequent transmission of task updates to the MEC server, which is comparatively higher than the energy consumed when offloading data for processing. When N/M is small, the proportion of IDs employing the *SL* action rises with an increase in N/M, leading to a higher average energy consumption. However, as N/M becomes large, the reduced transmission success probability leads to a larger number of IDs taking *I* operation, thus resulting in a lower average energy consumption of IDs.

Fig. 13 shows the impact of channel bandwidth on the average AoI with different values of the size of sensed information. The results demonstrate a notable reduction in average AoI as the channel bandwidth increases. This is because, according to (5) and (9), the transmission success rate of IDs increases as the channel bandwidth increases, which allows the MEC server to obtain the updated status of IDs in a timely manner. In addition, it can be seen that the average AoI increases as the data size of sensed information increases. This is because as the ID senses more information from the environment, it takes more time for ID to transmit the raw data to the MEC server.

Fig. 14 depicts the influence of channel bandwidth on the average energy consumption, considering different data sizes of sensed information. The results demonstrate a clear inverse relationship between channel bandwidth and average energy consumption. Notably, as the channel bandwidth increases,

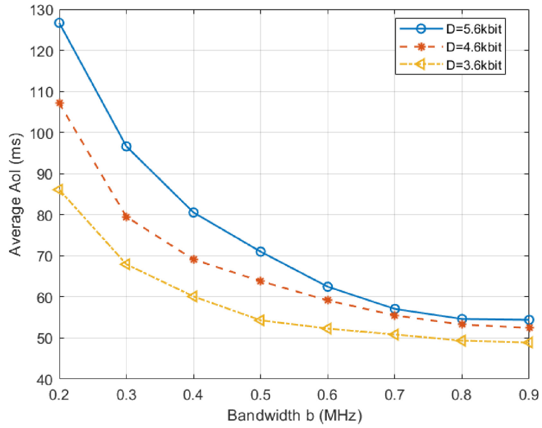

Fig. 13. Average AoI versus bandwidth b.

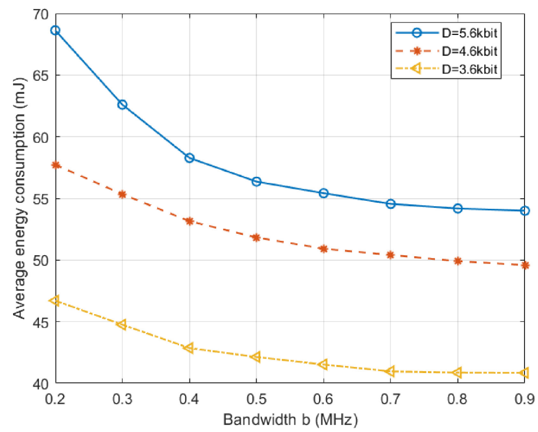

Fig. 14. Average energy consumption versus bandwidth b.

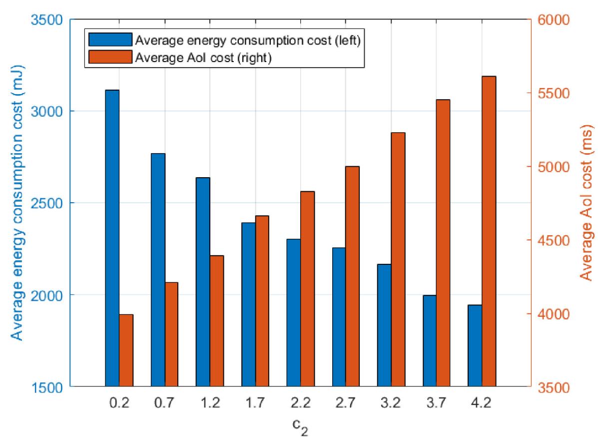

Fig. 15. Tradeoff between the energy consumption cost and the AoI cost.

the average energy consumption decreases. Additionally, it is evident that a larger data size of sensed information corresponds to a decreased average energy consumption for IDs. This phenomenon can be attributed to the reduced transmission delay of IDs as a result of increased channel bandwidth, leading to a lower overall energy consumption. Moreover, a higher amount of sensed data means that IDs need more energy for realizing sensing, computation, and transmitting operations.

In Fig. 15, we illustrate the tradeoff between the average energy consumption cost and the average AoI, while keeping the parameter $c_1 = 1$ and varying parameter c_2 from 0.2 to 4.2. The results demonstrate an inverse relationship between these costs. As c_2 increases, the energy consumption of IDs decreases, while the average AoI of IDs increases. This behavior arises from the prioritization of AoI minimization when c_2 is small, leading to increased energy consumption for computation and data transmission by IDs. Conversely, when c_2 is large, the priority shifts toward minimizing energy consumption, leading to lower energy consumption for IDs.

VI. Conclusion

In this article, we investigate the joint sensing and computation optimization problem for MEC-assisted IoT system, aiming to minimize the long-term average system cost. To address the challenges associated with interaction and computation complexity caused by a large number of IDs, we introduce the GMFG and then reformulate it as an MFG with teams of agents. We then introduce an innovative solution, MFGAC, specifically designed to tackle the optimization problem. Performance evaluations indicate that our proposed MFGAC has good stability, faster convergence during the learning process, and achieves lower system cost compared to other benchmark schemes. Furthermore, the simulation analysis illustrates the tradeoff between the average AoI and the average energy consumption cost. In future research, we aim to explore the applicability of RL-based MFG in different types of IoT systems, such as smart cities, industrial IoT, and healthcare, and it would be meaningful to consider more complex energy models that introduce dynamic power control.

References

[1] 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, Nov. 2018.

[2] L. Da Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.

[3] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Netw.*, vol. 34, no. 3, pp. 134–142, May/Jun. 2019.

[4] 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, Oct. 2020.

[5] C. Xu, Y. Xie, X. Wang, H. H. Yang, D. Niyato, and T. Q. S. Quek, "Optimal status update for caching enabled IoT networks: A dueling deep R-network approach," *IEEE Trans. Wireless Commun.*, vol. 20, no. 12, pp. 8438–8454, Dec. 2021.

[6] J. Kim, M. Kim, M.-S. Kim, and J. Lee, "Ensuring data freshness in wireless monitoring networks: Age-of-information-sensitive coverage and energy efficiency perspectives," *IEEE Internet Things J.*, vol. 10, no. 14, pp. 12811–12825, Jul. 2023.

[7] B. Zhou and W. Saad, "Age of information in ultra-dense IoT systems: Performance and mean-field game analysis," *IEEE Trans. Mobile Comput.*, vol. 23, no. 5, pp. 4533–4547, May 2023.

[8] H. Zhang, Y. Kang, L. Song, Z. Han, and H. V. Poor, "Age of information minimization for grant-free non-orthogonal massive access using mean-field games," *IEEE Trans. Commun.*, vol. 69, no. 11, pp. 7806–7820, Nov. 2021.

[9] Y. Chen, Z. Chang, G. Min, S. Mao, and T. Hämäläinen, "Joint optimization of sensing and computation for status update in mobile edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 22, no. 11, pp. 8230–8243, Nov. 2023.

[10] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. IEEE INFOCOM*, Orlando, FL, USA, 2012, pp. 2731–2735.

[11] D. S. Lakew, A.-T. Tran, N.-N. Dao, and S. Cho, "Intelligent offloading and resource allocation in heterogeneous aerial access IoT networks," *IEEE Internet Things J.*, vol. 10, no. 7, pp. 5704–5718, Apr. 2022.

[12] X. Li, S. Bi, Y. Zheng, and H. Wang, "Energy-efficient online data sensing and processing in wireless powered edge computing systems," *IEEE Trans. Commun.*, vol. 70, no. 8, pp. 5612–5628, Aug. 2022.

[13] R. Xu, Z. Chang, Z. Zhao, and G. Min, "Contract-based incentive mechanism for blockchain-enabled federated learning in vehicle edge computing," in *Proc. IEEE Global Commun. Conf.*, Kuala Lumpur, Malaysia, 2023, pp. 1812–1817.

[14] M. Song, H. H. Yang, H. Shan, J. Lee, and T. Q. S. Quek, "Age of information in wireless networks: Spatiotemporal analysis and locally adaptive power control," *IEEE Trans. Mobile Comput.*, vol. 22, no. 6, pp. 3123–3136, Jun. 2023.

[15] S. Feng and J. Yang, "Age of information minimization for an energy harvesting source with updating erasures: Without and with feedback," *IEEE Trans. Commun.*, vol. 69, no. 8, pp. 5091–5105, Aug. 2021.

[16] J. Hribar, M. Costa, N. Kaminski, and L. A. DaSilva, "Using correlated information to extend device lifetime," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2439–2448, Apr. 2019.

[17] Q. Hu, Y. Cai, G. Yu, Z. Qin, M. Zhao, and G. Y. Li, "Joint offloading and trajectory design for UAV-enabled mobile edge computing systems," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1879–1892, Apr. 2019.

[18] H. Guo, X. Zhou, J. Wang, J. Liu, and A. Benslimane, "Intelligent task offloading and resource allocation in digital twin based aerial computing networks," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 10, pp. 3095–3110, Aug. 2023.

[19] H. Guo, Y. Wang, J. Liu, and C. Liu, "Multi-UAV cooperative task offloading and resource allocation in 5G advanced and beyond," *IEEE Trans. Wireless Commun.*, vol. 23, no. 1, pp. 347–359, May 2024.

[20] N. Huang, C. Dou, Y. Wu, L. Qian, B. Lin, and H. Zhou, "Unmanned-aerial-vehicle-aided integrated sensing and computation with mobile-edge computing," *IEEE Internet Things J.*, vol. 10, no. 19, pp. 16830–16844, Oct. 2023.

[21] S. Yun, D. Kim, C. Park, and J. Lee, "Joint sensing and computation decision for age of information-sensitive wireless networks: A deep reinforcement learning approach," in *Proc. IEEE Global Commun. Conf.*, Kuala Lumpur, Malaysia, 2023, pp. 338–343.

[22] O. S. Oubbati, M. Atiquzzaman, A. Lakas, A. Baz, H. Alhakami, and W. Alhakami, "Multi-UAV-enabled AoI-aware WPCN: A multi-agent reinforcement learning strategy," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Vancouver, BC, Canada, 2021, pp. 1–6.

[23] F. Wu, H. Zhang, J. Wu, L. Song, Z. Han, and H. V. Poor, "AoI minimization for UAV-to-device underlay communication by multi-agent deep reinforcement learning," in *Proc. IEEE Global Commun. Conf.*, Taipei, Taiwan, 2020, pp. 1–6.

[24] J.-M. Lasry and P.-L. Lions, "Mean field games," *JPN J. Math.*, vol. 2, no. 1, pp. 229–260, Mar. 2007.

[25] X. Guo, A. Hu, R. Xu, and J. Zhang, "Learning mean-field games," in *Proc. 33rd Conf. Neural Inf. Process. Syst.*, 2019, p. 1–11.

[26] H. Gu, X. Guo, X. Wei, and R. Xu, "Mean-field multi-agent reinforcement learning: A decentralized network approach," 2021, *arXiv:2108.02731*.

[27] Y. Kang, H. Wang, B. Kim, J. Xie, X.-P. Zhang, and Z. Han, "Time efficient offloading optimization in automotive multi-access edge computing networks using mean-field games," *IEEE Trans. Veh. Technol.*, vol. 72, no. 5, pp. 6460–6473, May 2023.

[28] H. Zhang et al., "Mean-field-aided multiagent reinforcement learning for resource allocation in vehicular networks," *IEEE Internet Things J.*, vol. 10, no. 3, pp. 2667–2679, Feb. 2023.

[29] R. Li, Q. Ma, J. Gong, Z. Zhou, and X. Chen, "Age of processing: Age-driven status sampling and processing offloading for edge-computing-enabled real-time IoT applications," *IEEE Internet Things J.*, vol. 8, no. 19, pp. 14471–14484, Oct. 2021.

[30] R. Lowe, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, "Multi-agent actor–critic for mixed cooperative-competitive environments," in *Proc. 31st Conf. Neural Inf. Process. Syst.*, 2017, pp. 1–12.

[31] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a UAV-mounted cloudlet: Optimization of bit allocation and path planning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2049–2063, Mar. 2017.

[32] Z. Yu, Y. Gong, S. Gong, and Y. Guo, "Joint task offloading and resource allocation in UAV-enabled mobile edge computing," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3147–3159, Apr. 2020.

[33] T. Li et al., "A mean field game-theoretic cross-layer optimization for multi-hop swarm UAV communications," *J. Commun. Netw.*, vol. 24, no. 1, pp. 68–82, Feb. 2021.

[34] T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*.

[35] V. Mnih et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[36] C. Yu et al., "The surprising effectiveness of PPO in cooperative multi-agent games," in *Proc. 36th Conf. Neural Inf. Process. Syst.*, 2022, pp. 24611–24624.

Runchen Xu (Graduate Student Member, IEEE) received the B.S. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2022, where he is currently pursuing the M.S. degree with the School of Computer Science and Engineering.

His research interests include wireless communications and networking, Internet of Things, blockchain, machine learning, resource allocation, game theory, and mobile computing.

Zheng Chang (Senior Member, IEEE) received the B.Eng. degree from Jilin University, Changchun, China, in 2007, the M.Sc. (Tech.) degree from Helsinki University of Technology (currently Aalto University), Espoo, Finland, in 2009, and the Ph.D. degree from the University of Jyväskylä, Jyväskylä, Finland, in 2013.

Since 2008, he has held various research positions with Helsinki University of Technology, the University of Jyväskylä, and Magister Solutions Ltd., Jyväskylä. He was a Visiting Researcher with Tsinghua University, Beijing, China, from June to August 2013, and the University of Houston, Houston, TX, USA, from April to May 2015. He has published over 190 papers in journals and conferences. His research interests include IoT, cloud/edge computing, security and privacy, vehicular networks, and green communications.

Dr. Chang has been awarded by the Ulla Tuominen Foundation, the Nokia Foundation, and the Riitta and Jorma J. Takanen Foundation for his research excellence. He has been awarded as the 2018 IEEE Communications Society Best Young Researcher for Europe, Middle East, and Africa Region and the 2021 IEEE Communications Society MMTC Outstanding Young Researcher. He received the Best Paper Awards from IEEE ICC in 2023, and IEEE TCGCC and APCC in 2017. He serves as an Editor for IEEE WIRELESS COMMUNICATIONS LETTERS, *Wireless Networks* (Springer), and *China Communications*, and a Guest Editor for IEEE NETWORK, IEEE WIRELESS COMMUNICATIONS, *IEEE Communications Magazine*, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, *Physical Communications*, *EURASIP Journal on Wireless Communications and Networking*, and *Wireless Communications and Mobile Computing*. He was the Exemplary Reviewer of IEEE WIRELESS COMMUNICATION LETTERS in 2018. He has participated in organizing workshop and special session in Globecom'19, WCNC'18–'24, SPAWC'19, and ISWCS'18. He also serves as the Symposium Co-Chair for IEEE ICC'20 and Globecom'23, the Publicity Co-Chair for IEEE Infocom'22, the Workshop Co-Chair for ICCC'22, the TPC Co-Chair for IEEE iThing'22, and a TPC Member for many IEEE major conferences, such as INFOCOM, ICC, and Globecom.

Zhu Han (Fellow, IEEE) received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland at College Park, College Park, MD, USA, in 1999 and 2003, respectively.

From 2000 to 2002, he was a Research and Development Engineer with JDSU, Germantown, MD, USA. From 2003 to 2006, he was a Research Associate with the University of Maryland at College Park. From 2006 to 2008, he was an Assistant Professor with Boise State University, Boise, ID, USA. He is currently a John and Rebecca Moores Professor with the Electrical and Computer Engineering Department and the Computer Science Department, University of Houston, Houston, TX, USA. His main research targets on the novel game-theory-related concepts critical to enabling efficient and distributive use of wireless networks with limited resources. His other research interests include wireless resource allocation and management, wireless communications and networking, quantum computing, data science, smart grid, carbon neutralization, and security and privacy.

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, the IEEE Leonard G. Abraham Prize in the field of Communications Systems (Best Paper Award in IEEE JSAC) in 2016, the IEEE Vehicular Technology Society 2022 Best Land Transportation Paper Award, and several best paper awards in IEEE conferences. He was an IEEE Communications Society Distinguished Lecturer from 2015 to 2018 and an ACM Distinguished Speaker from 2022 to 2025, and has been an AAAS Fellow since 2019 and an ACM Fellow since 2024. He has been a 1% Highly Cited Researcher since 2017 according to Web of Science. He is also the winner of the 2021 IEEE Kiyo Tomiyasu Award (an IEEE Field Award), for outstanding early to mid-career contributions to technologies holding the promise of innovative applications, with the following citation: "for contributions to game theory and distributed management of autonomous communication networks."

Georges Kaddoum (Senior Member, IEEE) received the bachelor's degree in electrical engineering from the École Nationale Supérieure de Techniques Avancées (ENSTA Bretagne), Brest, France, the M.S. degree in telecommunications and signal processing (circuits, systems, and signal processing) from the Université de Bretagne Occidentale and Télécom Bretagne (ENSTB), Brest, in 2005, and the Ph.D. degree (Hons.) in signal processing and telecommunications from the National Institute of Applied Sciences (INSA), University of Toulouse, Toulouse, France, in 2009.

Since 2010, he has been a Scientific Consultant in the field of space and wireless telecommunications for several U.S. and Canadian companies. He is currently a Professor and a Tier 2 Canada Research Chair with the École de technologie supérieure (ÉTS), Université du Québec, Montreal, QC, Canada, and a Faculty Fellow with the Cyber Security Systems and Applied AI Research Center, Lebanese American University, Beirut, Lebanon. He has published over 300 journal articles, conference papers, two chapters in books, and has eight pending patents. His recent research activities cover wireless communication networks, tactical communications, resource allocations, and security.

Prof. Kaddoum was awarded the ÉTS Research Chair in physical-layer security for wireless networks in 2014. He received the Best Papers Award from the 2014 IEEE International Conference on Wireless and Mobile Computing, Networking, Communications (WIMOB), with three co-authors, and from the 2017 IEEE International Symposium on Personal Indoor and Mobile Radio Communications, with four co-authors. Moreover, he received the IEEE Transactions on Communications Exemplary Reviewer Award in 2015, 2017, and 2019. In addition, he received the Research Excellence Award from the Université du Québec in 2018. In 2019, he received the Research Excellence Award from ÉTS in recognition of his outstanding research outcomes. He is serving as an Area Editor for IEEE TRANSACTIONS ON MACHINE LEARNING IN COMMUNICATIONS AND NETWORKING and an Associate Editor for IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE TRANSACTIONS ON COMMUNICATIONS, and IEEE COMMUNICATIONS LETTERS.

Sahil Garg (Member, IEEE) is currently an AI/ML Architect with Ultra Communications, Montreal, QC, Canada, and also an Adjunct Associate Professor with the École de technologie supérieure (ÉTS), Montreal. He is also an Adjunct Faculty with the Centre for Research Impact and Outcome, Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, India. Prior to this, he worked as a Research Professional with Resilient Machine Learning Institute, Montreal; a Postdoctoral Research Fellow with ÉTS; and a MITACS Researcher with Ericsson, Montreal. He has over 90 publications in highly ranked journals and conferences, including 60+ top-tier journal papers and 30+ reputed conference articles. He has contributed much research in the areas of machine learning, big data analytics, security and privacy, Internet of Things, and cloud computing.

Dr. Garg has been awarded the 2022 IEEE HITC Early Career Researcher Award, the 2021 IEEE Systems Journal Best Paper Award, the 2020 IEEE TCSC Award for Excellence in Scalable Computing (Early Career Researcher), and the IEEE ICC Best Paper Award in 2018 at Kansas City, MO, USA. He is currently a Senior Editor of *Human-Centric Computing and Information Sciences*. He is also an Associate Editor of IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, *Applied Soft Computing* (Elsevier), and *International Journal on Communication Systems* (Wiley). He also served as an Associate Editor for other reputed journals, such as *Future Generation Computer Systems* (Elsevier), *IEEE Network Magazine*, and IEEE SYSTEMS JOURNAL. In addition, he also serves as the Workshops and Symposia Officer for the IEEE ComSoc Emerging Technology Initiative on Aerial Communications.

Joel J. P. C. Rodrigues (Fellow, IEEE) is with Amazonas State University, Manaus, Brazil. He has authored or co-authored over 1000 papers in refereed international journals and conferences, three books, two patents, and one ITU-T recommendation.

Dr. Rodrigues had been awarded several Outstanding Leadership and Outstanding Service Awards by IEEE Communications Society and several best papers awards. He is also the Leader of the Next Generation Networks and Applications Research Group (CNPq), the Director for Conference Development—IEEE ComSoc Board of Governors, an IEEE Distinguished Lecturer, the Technical Activities Committee Chair of the IEEE ComSoc Latin America Region Board, the President of the Scientific Council at ParkUrbis—Covilhã Science and Technology Park, the Past Chair of the IEEE ComSoc Technical Committee on eHealth and the IEEE ComSoc Technical Committee on Communications Software, a Steering Committee Member of the IEEE Life Sciences Technical Community and the Publications Co-Chair, and a Member Representative of the IEEE Communications Society on the IEEE Biometrics Council. He is the Editor-in-Chief of the *International Journal of E-Health and Medical Communications* and an editorial board member of several high-reputed journals. He has been the General Chair and the TPC Chair of many international conferences, including IEEE ICC, IEEE GLOBECOM, IEEE HEALTHCOM, and IEEE LatinCom. He is a member of the Internet Society and a Senior Member of ACM.