Frequency Resource Allocation and Interference Management in Mobile Edge Computing for an Internet of Things System

Woongsoo Na, Seonmin Jang, Yoonseong Lee, Laihyuk Park, Nhu-Ngoc Dao, and Sungrae Cho

Abstract—**Internet of Things (IoT) systems are characterized by highly automated operating environments, which comprise several IoT end devices (IDs) that generate vast amounts of data with strict real-time communication and high data rate requirements. Edge computing facilities are an alternative to traditional cloud computing and support massive data processing in IoT systems while reducing the burden on data centers. In this paper, we consider an edge-based IoT system that comprises an edge server (ES), edge gateways (EGs), and IDs that communicate wirelessly. The EGs reduce the load on the ES by preprocessing data received from ID. However, it may not be possible for a few EGs to accommodate a sheer number of IDs, given the limited computing power and communication coverage of the EGs. Therefore, it is necessary for a few IDs to directly connect to the ES without the support of EGs. Thus, we propose a resource orchestration scheme between EGs and ES and/or among EGs based on a Lagrangian and the Karush–Kuhn–Tucker condition. The scheme allocates optimal resources by considering the computing capacities of EGs and ES and manages interference among the EGs to maximize the efficiency of IoT systems. The performance evaluation indicates that the proposed scheme outperforms the existing schemes in terms of aggregate throughput, latency, data reception rate, and workload fairness among EGs by 42%, 59%, 37%, and 40%, respectively.**

Index Terms—**Edge-based Internet of Things (IoT) system, edge-gateway (EG), edge-server (ES), IoT device, resource allocation.**

I. Introduction

EDGE computing presents a significant opportunity to unburden existing cloud computing infrastructure by extending cloud services to edge nodes with sufficient amounts of storage, computation, communication capabilities, and management functions at the edge of the network [1], [2]. This delivers significant benefits including reduced response times, increased spectral efficiency, security, and lower operating expenses, and is suitable for applications that produce vast amounts of time-sensitive data.

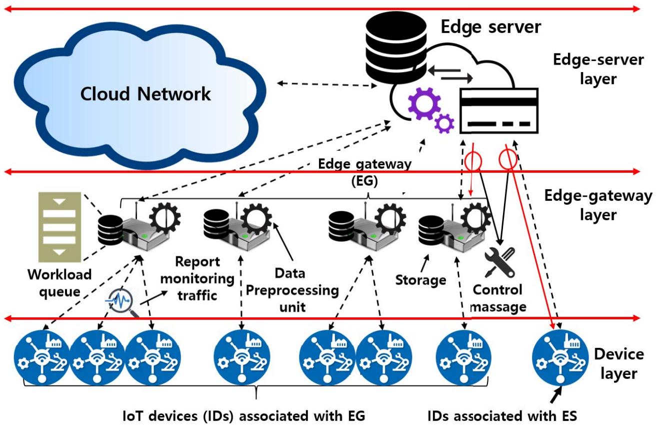


Fig. 1. Edge-based IoT system architecture based on wireless links (all links are wirelessly connected (black dotted lines).

An Internet of Things (IoT) system is characterized by a highly automated operating environment, consisting of IoT devices (IDs) with strict real-time communication requirements [3], [4]. Furthermore, the number of IDs increases annually, thereby resulting in the generation of significant amounts of data in addition to requirements for higher data rates. Given this dynamic, edge computing is a promising technology in an IoT system environment.

Unfortunately, traditional edge-based IoT systems including smart manufacturing system interconnect equipment, such as IDs, edge gateways (EGs), and edge server (ES), by using wired communication media. This implies that deployment costs become prohibitive when the amount of equipment increases, thereby affecting the scalability of the network. Thus, it is necessary to connect edge-based IDs wirelessly as a means to reduce deployment costs and increase service scalability. Conversely, wireless edge-based IoT systems exhibit the advantage of scalability such as longer distance service when compared to existing solutions without special facilities and easy deployment of new devices. Fortunately, this is achieved through conventional cellular networks or sufficiently wide bandwidth communication protocols (e.g., LTE [5], [6]).

Fig. 1 shows an example of an edge-based IoT system architecture. As shown in the figure, an edge-based IoT system

Manuscript received June 9, 2018; revised September 21, 2018 and November 16, 2018; accepted November 30, 2018. Date of publication December 6, 2018; date of current version June 19, 2019. This work was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science ICT under Grant NRF-2017R1A2B4009802. *(Corresponding author: Sungrae Cho.)*

W. Na is with the Electronics and Telecommunications Research Institute, Daejeon, South Korea (e-mail: wsna@uclab.re.kr).

S. Jang is with Network System Software Engineering, Samsung Electronics, Suwon, South Korea (e-mail: smjang@uclab.re.kr).

Y. Lee, L. Park, N.-N. Dao, and S. Cho are with the School of Computer Science and Engineering, Chung-Ang University, Seoul 156-756, South Korea (e-mail: yslee@uclab.re.kr; lhpark@uclab.re.kr; dnngoc@uclab.re.kr; srcho@cau.ac.kr).

Digital Object Identifier 10.1109/JIOT.2018.2885348

consists of one ES layer, and multiple EGs layer, and IDs (device layer). The ID simply senses the environment and transmits status information to the EG. The EGs exhibit a computing capacity and are deployed in close proximity to the IDs to receive monitoring traffic. The received monitoring traffic is preprocessed by an EG and delivered to the ES. Furthermore, IDs that are not located in an EG's region of coverage directly transmit monitoring frames to the ES. The ES corresponds to the orchestrator that manages the entire edge-based IoT system and controls certain EGs or IDs. Furthermore, it denotes the bridge that connects to external cloud servers. The structure provides IDs with prompt access and request services at extremely high-speed and significantly low transmission and computing latency.

However, given radio resource constraints, efficient resource management to enhance service capabilities is considered as an emerging challenge in edge-based applications. Specifically, the ES and EGs share radio resources in wireless edge-based IoT system environments, and thus the orchestrator (ES) should manage the interference between them. Given the aim of providing a higher data rate and service quality, the orchestrator may prefer to assign radio resources to an EG first. Although the approach achieves a better network throughput and spectral efficiency, the workload is concentrated on the EGs, and this can result in frequent interference between the EGs.

To resolve the problem, we consider the following environment in an edge-based IoT system. The IDs periodically send monitoring traffic to the associated EG or ES. EGs forward the received monitoring traffic to the ES for preprocessing. The ES stores/processes the traffic delivered from IDs and EGs and conveys control messages concerning network management back to the IDs and EGs as shown in Fig. 1. Hence, we propose a frequency resource allocation algorithm between the EGs and ES, and an interference management (IM) algorithm among the EGs. The contributions of this paper are as follows.

1) We proposed a queue backlog-based resource orchestration scheme by considering the storage/computing capacity and interference among EGs for wireless edge-based smart manufacturing system. The scheme determines the ratio of radio resource allocation between the EGs and ES through a Lagrangian and the Karush–Kuhn–Tucker (KKT) condition.
2) Different from other existing researches where a single channel is assumed, we modeled an edge based IoT network assuming a dual-channel (LTE for licensed channel and WiFi for unlicensed channel) with LTE-WLAN aggregation technique. With this assumption, our proposed scheme orchestrates the resource of licensed channel considering the available resources of the unlicensed channel.
3) We analyzed the proposed resource orchestration scheme comparing other resource allocation scheme in terms of throughput, latency, availability, and transmission fairness in the Samsung Vietnam semi-conductor manufacturing system.

The remainder of this paper is organized as follows. In Section II, we describe related work on edge-based IoT systems. Furthermore, we describe the system model for edge-based IoT systems in Section III, and in Section IV we analyze the performance of the proposed scheme with a Lagrangian function and determine the optimal ratio of resource allocation between EGs and ES. Section V evaluates the performance via simulations, and the conclusions and suggestions for future research directions are detailed in Section VI.

II. Related Work

Fog computing brings the capabilities and benefits of cloud computing to edge devices, and this is close to where data is generated. This generated significant interest for multiple industry verticals and technologies including semiconductors, connected cars, and IT hardware industries such as NVDIA, HPE, Dell, and EdgeXFoundary [7]. The aforementioned companies established various service platforms with the aim of minimizing user latency and ensuring compatibility in various industrial environments.

Studies on fog and edge computing are performed in both academia and the industrial sector. Zeng *et al.* [8] designed a task scheduling and resource management scheme with a minimized task completion time to improve user experience. A three-stage heuristic algorithm is proposed to solve the mixed integer nonlinear programming problem of minimizing the task completion time and consists of the computation time, I/O interrupt handling response time, and transmission time.

Energy-efficient resource management is another area of interest in fog and edge computing networks [9]–[11]. Zhang *et al.* [9] focused on the energy efficiency of computation offloading networks with the aim of minimizing energy consumption under task latency constraints. Energy-efficient computation offloading mechanisms jointly optimize offloading and radio resource allocation to obtain the minimal energy consumption under the latency constraints. Sun *et al.* [10] suggested an energy-aware mobility management (EMM) scheme based on Lyapunov optimization and multiarmed bandit algorithm. The EMM scheme optimizes the delay caused by both the radio handover and computation migration cost. Dinh *et al.* [11] minimized task execution latency as well as mobile device energy consumption. In contrast to other studies, the aforementioned study focuses on minimizing the energy consumption of a mobile device that is served by multiple edges and solves the minimization problem by using a linear programming relaxation and semi-definite relaxation approach.

In [12], Stackelberg game and matching theory is applied to a resource allocation problem involving fog nodes, data service operators, and data service subscribers. The authors proposed a framework that solves the resource purchasing problem and pairing problem and achieves an equilibrium or stable results. Stochastic optimization is typically utilized in resource allocation problems for fog networks [13], [14]. In [13], resource allocation in a fog setting is formulated as a stochastic optimization problem by considering queuing delay, traffic arrivals, and channel conditions as constraints. A joint mode selection and resource allocation algorithm is proposed and achieves a flexible tradeoff between the average throughput and delay. In [14], a stochastic joint radio and computational resource management scheme is proposed based on Lyapunov optimization to deal with the tradeoff between the average weighted sum power consumption and average execution delay in a multiuser MEC system. Ni *et al.* [15] proposed priced timed Petri nets (PTPN) models and designed algorithms for resource allocation in fog computing. A strategy based on PTPN guides dynamic resource allocation in fog computing, predicts the completion times of tasks, computes the credibility evaluation of resources, and dynamically allocates fog resources.

The above resource allocation techniques for edge-based networks is classified into the following two categories: 1) central-guided and 2) *ad hoc* solutions. In centralized techniques [8]–[11], [13]–[15], there is a central server or coordinator that collects status information, such as data rate, storage capacity, and computing resources of edge. Based on the collected information, the central entity allocates optimized resources to each edge device. Conversely, in distributed solutions [12], [16], [17], each edge device acquires limited resources by coexisting with neighboring edge devices, negotiating, and competing. The former is resource-optimized although the building costs are high. Conversely, the latter costs less although it is difficult to obtain optimized results. In the latter case, the communication overhead in distributed environments is extremely high given the high number of edge devices deployed in IoT scenarios in general. Thus, edge-based IoT systems use more centralized resource allocation techniques and the study also assumes centralized architecture.

Additionally, only fog nodes are considered in previous studies, and either computing resources and network frequency resources are distributed or tasks are scheduled. However, in several industries including smart manufacturing systems, the deployment of a fog node to cover all user devices is not scalable and is also costly. Therefore, a few user devices send and receive services through a connection with the existing cloud server. At this time, a resource allocation technique between the cloud server and fog nodes is necessary. In this paper, we propose an optimal resource allocation algorithm between a cloud server and edge nodes by assuming that IDs connected to edge nodes and a cloud server are mixed in a IoT system. Additionally, we propose a resource reallocation algorithm based on the workload (queue backlog size) between edge nodes where coverages overlap.

III. System Model

A. Basic Assumption

The overall system model is illustrated in Fig. 1 and key notations are described in Table I. We consider a network model that consists of an ES and N_g EGs in an IoT system. The ES assigns a workload for data preprocessing to the EGs, and the IDs wirelessly transmit monitoring frames to associated EGs or the ES. In the network, N_s^m IDs are associated with the ES, and the ith EG includes $N_{g(i)}^m$ IDs within its service coverage. We assume that N_g^m denotes the total number of IDs associated with EGs. Therefore, N_g^m is obtained as $N_g^m = \sum_i^{N_g} N_{g(i)}^m$.

TABLE I
Key Notation Descriptions

Notations	Description
C	Spectral efficiency
N_s^m	Number of IDs associated to ES
N_g	Number of EGs
N_g^m	Number of IDs associated to EG
RB_s	Allocated number of wireless resource blocks for ES
RB_g	Allocated number of wireless resource blocks for EG
RB_{tot}	Total number of wireless resource Blocks
r_{max}	Maximum data rate of licensed channel
$r_s(j)$	Data rate for licensed channel that ES allocates to its j_{th} ID
$r_g(i,j)$	Data rate for licensed channel that i_{th} EG allocates to its j_{th} ID
$r_g^{'}(i,j)$	Data rate of j_{th} ID within i_{th} EG range for unlicensed channel
w_s	Weight parameter of ES
w_g	Weight parameter of EG

In an IoT system, the EGs can be temporarily overloaded or unavailable due to external factors (e.g., interruption of electricity supply or system failure). Additionally, a few IDs are mobile, and thus they alternate between the coverage of the EGs and ES. Thus, we assume that the IDs covered by the ES report a monitoring frame to the ES during a superframe period while the IDs associated with an EG decide whether to report a monitoring frame to the ES or an EG by considering the mobility of the node. Therefore, if the mobility of an ID (m) associated with an EG exceeds a certain threshold, then the ID m transmits a monitoring frame to the ES ($v \in N_s^u$). Otherwise, $v \in N_{g(i)}^u$.

All the IDs in the licensed band possess a cellular network interface that divides the spectrum into RB_{tot} radio resource blocks. Alternately, IDs in the unlicensed band use a contention-based channel access scheme. Thus, they use a CSMA/CA scheme to transmit data frames in the time domain.

The ES receives the report frame from its jth ID within the achievable data rate $r_s(j)$. When compared with the single ES, there are several EGs. Therefore, we assume that each ES receives monitoring frames with different data rates, and thus the ith EG receives monitoring frames from the jth ID based on the achievable data rate $r_g(i, j)$.

IV. Network Analysis

In this section, we analyze the performance of the edge-based IoT system based on the different methods of resource allocation between the ES and EGs. Thus, we design an optimization problem to obtain the theoretical maximum throughput of the edge-based IoT system for our analysis.

The approach of the optimization problem involves computing two weight parameters, namely w_s and w_g for the ES and EGs, respectively. Based on w_s and w_g, the total resource blocks RB_{tot} are separately allocated to the ES as RB_s and to the EGs as RB_g. In this context, we obtain suitable w_s and w_g values that maximize the network throughput by solving the optimization problem using Lagrangian functions and KKT conditions. The parameter setting, optimization constraints, and definition of the optimization problem are explained as follows.

To fairly allocate resource blocks to the ES and EGs, we consider the number of IDs in each coverage area and the achievable data rate that each ES or EG allocates to its IDs to set w_s and w_g. Hence, w_s is defined as

$$w_s = N_s^m \times \max(r_s(j)),\ \ 1 \le j \le N_s^m \tag{1}$$

where $r_s(j)$ denotes the achievable data rate that jth ID transmits to the ES. Here, w_s is proportional to the number of IDs associated with the ES and the data rate that the ES allocates to the IDs.

Similarly, w_g is also proportional to the average number of IDs associated with EGs and the data rate that the ES allocates to the IDs. Additionally, w_g considers data rates of unlicensed channels, such as 2.4 or 5 GHz, to increase the data rate via a link aggregation scheme [18]. The data rate of an unlicensed channel is based on [19]. Thus, w_g is defined as

$$\begin{aligned} w_g = \frac{\sum_{i=1}^{N_g} N_{g(i)}^m}{N_g} \times \max\big(r_g(i,j)\big) - \min\Big(r'_g(i,j)\Big),\\ 1 \le i \le N_g;\ \ 1 \le j \le N_{g(i)}^m \ \ \forall i,j. \end{aligned} \tag{2}$$

The total number of resource blocks RB_{tot} is automatically divided between an ES and EGs based on w_s and w_g. We note that RB_s and RB_g represent the numbers of resource blocks allocated to the ES and EGs, respectively. Hence, RB_s and RB_g are defined as

$$RB_s = \frac{w_s}{w_s + w_g} \times RB_{\text{tot}} \text{ and } RB_g = \frac{w_g}{w_s + w_g} \times RB_{\text{tot}}. \tag{3}$$

We use RB_s and RB_g to calculate each ID's received data rate for a licensed channel with their spectral efficiency C_j. Therefore, the received data rates of the jth ID from the ES and the ith EG are defined as

$$r_s(j) = C_j \times RB_s \text{ and } r_g(i,j) = C_j \times RB_g \tag{4}$$

respectively, [20]. Note that, it is difficult to find the current spectral efficiency. To estimate the current spectral efficiency, we used the scheme in [21] and [22].

The summation of RB_s and RB_g should be less than or equal to the total number of resource blocks

$$RB_s + RB_g \le RB_{\text{tot}}. \tag{5}$$

To maximize the network throughput, we assume that resource block loss does not occur due to external factors such as the channel environment. Therefore, we transform (5) into (6) as follows:

$$RB_s + RB_g = RB_{\text{tot}}. \tag{6}$$

We consider the achievable data rate as the total capacity for an ES or EG, and thus we define the ranges of achievable data rates for the ES and an EG as

$$0 \le r_s(j) \le r_{\max} \text{ and } 0 \le r_g(i,j) \le r_{\max} \tag{7}$$

respectively.

Hence, we re-express (7) in terms of w_s and w_g by using (1) and (2). Furthermore, we obtain the following:

$$0 \le w_s \le N_u^m \times r_{\max} \tag{8}$$

$$0 \le w_g \le \frac{\sum_{i=1}^{N_g} N_{g(i)}^m}{N_g} \times r_{\max} - \min\Big(r'_g(i,j)\Big). \tag{9}$$

We set our utility function as the total of the received data rates for all the IDs. We increase each ID's experience to maximize the total network throughput

$$U\big(w_s, w_g\big) = \sum_{j=1}^{N_s^m} r_s(j) + \sum_{i=1}^{N_g}\sum_{j=1}^{N_{g(i)}^m} r_g(i,j) + \sum_{i=1}^{N_g}\sum_{j=1}^{N_{g(i)}^m} r'_g(i,j). \tag{10}$$

We use (6), (8), and (9) to formulate the optimization problem as

$$\max \quad \sum_{j=1}^{N_s^m} r_s(j) + \sum_{i=1}^{N_g}\sum_{j=1}^{N_{g(i)}^m} r_g(i,j) + \sum_{i=1}^{N_g}\sum_{j=1}^{N_{g(i)}^m} r'_g(i,j) \tag{11}$$

$$\text{subject to} \quad RB_s + RB_g = RB_{\text{tot}} \tag{12}$$

$$0 \le w_s \le N_s^m \times r_{\max} \tag{13}$$

$$0 \le w_g \le \frac{\sum_{i=1}^{N_g} N_{g(i)}^m}{N_g} \times r_{\max} - \min\Big(r'_g(i,j)\Big). \tag{14}$$

A. Lagrangian Form of Objective Function

To solve our optimization problem, a Lagrangian function is adopted by using (8) and (9). We do not consider (6) because (6) is automatically considered by using (3). Therefore, our Lagrangian function is defined as

$$\begin{aligned} &\text{Lag}\big(\mu_1, \mu_2, w_s, w_g\big) \\ &\quad = U\big(w_s, w_g\big) + \mu_1\big(w_s - N_s^m \times r_{\max}\big) \\ &\qquad + \mu_2\left(w_g - \frac{\sum_{i=1}^{N_g} N_{g(i)}^m}{N_g} \times r_{\max} + \min\Big(r'_g(i)\Big)\right). \end{aligned} \tag{15}$$

It should be noted that (8) and (9) are defined as inequalities, and it is difficult to solve the Lagrangian function given the inequality constraints. Consequently, KKT conditions are adopted to solve the Lagrangian function. Equations (16)–(22) express the KKT conditions of the Lagrangian function

$$1)\ w_s \le N_u^m \times r_{\max} \tag{16}$$

$$2)\ w_g \le \frac{\sum_{i=1}^{N_g} N_{g(i)}^m}{N_g} \times r_{\max} - \min\Big(r'_{g(i)}\Big) \tag{17}$$

$$\begin{aligned} &3)\ \frac{\partial}{\partial w_s}\text{Lag}\big(\mu_1, \mu_2, w_s, w_g\big) \\ &\quad = RB_g \times \frac{1}{w_s + w_g}\left(\sum_{j=1}^{N_s^m} C_j - \sum_{j=1}^{N_g^m} C_j\right) + \mu_1 = 0 \end{aligned} \tag{18}$$

4) $\frac{\partial}{\partial w_g}\mathrm{Lag}\big(\mu_1, \mu_2, w_s, w_g\big)$

$$= RB_s \times \frac{1}{w_s + w_g}\left(-\sum_{j=1}^{N_s^m} C_j + \alpha \sum_{j=1}^{N_g^m} C_j\right) - N_g N_g^m + \mu_2,$$

$$= 0, \text{ where } 1 \le \alpha \le N_g \tag{19}$$

5) $\mu_1\big(w_s - N_s^m \times r_{\max}\big) = 0$ (20)

6) $\mu_2\left(w_g - \frac{\sum_{i=1}^{N_g} N_{g(i)}^m}{N_g} \times r_{\max} + \min\Big(r'_g(i,j)\Big)\right) = 0$ (21)

7) $\mu_1, \mu_2 \ge 0.$ (22)

B. Models Based on the KKT Conditions

We use the KKT conditions, and the edge-based IoT system topology is divided into several cases based on the values of the KKT condition parameters μ_1 and μ_2. Each case model is explained as follows.

Case 1 (Only the ES Is Utilized $\mu_1 > 0$, $\mu_2 = 0$): In this case, we consider all IDs in the IoT system as highly mobile, and thus they cannot be associated with the EG (or the EGs are not deployed). Therefore, the EGs do not need to use the resource blocks, and only the ES monopolizes all the resource blocks

$$w_s^* = N_s^u \times r_{\max} \qquad w_g^* = 0.$$

Case 2 (EGs Are Densely Deployed $\mu_1 = 0$, $\mu_2 > 0$): In this case, several EGs are deployed in the IoT system and all the IDs are fixed or exhibit low mobility. Thus, all of the IDs are associated to EGs, and the EGs monopolize all the resource blocks

$$w_s^* = 0,\ \ w_g^* = \frac{\sum_{i=1}^{N_g} N_{g(i)}^m}{N_g} \times r_{\max} - \min\Big(r'_g(i,j)\Big).$$

Case 3 (EGs Are Deployed $\mu_1 = 0$, $\mu_2 = 0$): In this case, EGs are sparsely deployed, and the IDs selectively transmit monitoring traffic to the ES or EGs. Thus, the ES and EGs share the resource blocks at each time slot t.

In the model, we obtain the optimal ratio between w_s and w_g from (19) as follows:

$$w_s = 1 - \big(w_g + 1\big)^2 + \frac{1}{N_g N_g^m} RB_{\text{tot}}\left(\alpha \sum_{j=1}^{N_g^m} C_j - \sum_{j=1}^{N_s^m} C_j\right) \tag{23}$$

where $1 \le \alpha \le N_g$.

With $\alpha = N_g$ in (23), the difference in data rates for the ES and N_g determines w_s and w_g. Subsequently, we set the two determined weight parameters as w_s^* and w_g^*, respectively. After determining w_s^* and w_g^*, the ES/EGs use resource blocks in the corresponding ratio. In all cases, we obtain

$$RB_s^* = \frac{w_s^*}{w_s^* + w_g^*} \qquad RB_g^* = \frac{w_g^*}{w_s^* + w_g^*} \tag{24}$$

and we also get

$$r_s^* = C \times RB_s^* \qquad r_g^* = C \times RB_g^*. \tag{25}$$

Algorithm 1 Optimized Resource Allocation Scheme for ES and EGs (Wireless Resource Blocks)

```
KKT parameter update; // $\mu_1, \mu_2$
loop
if there are IDs that are associated to the EGs then
    Set $\mu_2 = 0$
    if EGs are deployed in IoT system then
      Set $\mu_1 = 0$ // allocate resources to the ES and EGs
    else
      Set $\mu_1 = 1$ // allocate resources to the ES
    end if
else
    Set $\mu_1 = 0$ and $\mu_2 = 1$ // allocate resources to EGs
end if
end loop

Primal Parameters Update; // $w_s^*, w_g^*, RB_s^*, RB_g^*$
loop
if $\mu_1 = 0$ and $\mu_2 > 0$ then
    Set $w_s^* = 0$, $w_g^* = \frac{\sum_{i=1}^{N_g} N_{g(i)}^m}{N_g} \times r_{max} - \min(r'_g(i,j))$
elseif $\mu_1 > 0$ and $\mu_2 = 0$ then
    if no interference coordination scheme then
        Set $RB_s^* = RB_g^* = RB_{tot}$
    else
        Set $w_s^* = N_s^m \times r_{max}$,    $w_g^* = 0$
    end if
elseif $\mu_1 = 0$ and $\mu_2 = 0$ then
    Set $w_s = 1 - (w_g + 1)^2$
        $+\frac{1}{N_g N_g^m} RB_{tot}(\alpha \sum_{j=1}^{N_g^m} C_j - \sum_{j=1}^{N_s^m} C_j)$
end if

if $RB_s^* = RB_g^* \neq RB_{tot}$ then
    Set $RB_s^* = \frac{w_s^*}{w_s^* + w_g^*} \times RB_{tot}$,
        $RB_g^* = \frac{w_g^*}{w_s^* + w_g^*} \times RB_{tot}$
end if
end loop
```

In the above situation, the coverage areas of the EGs can overlap with each other. Therefore, the report frames transmitted by the IDs can conflict with each other. Hence, a resource allocation algorithm is required between the EGs where coverage areas overlap. The resource allocation algorithm between the EGs should prioritize the EGs that exhibit high workloads, and the number of associated IDs should also be considered. The following sections present the interference model between the IDs and introduce the resource allocation algorithm between the EGs.

C. Resource Allocation Between EGs

1) Interference Model: To apply the proposed algorithm, we begin by setting the interference model between separate EGs. We adopt the A3 event measurement to verify the interference among EGs and adopt a radio resource control (RRC) measurement report for the IDs. An A3 event [which is

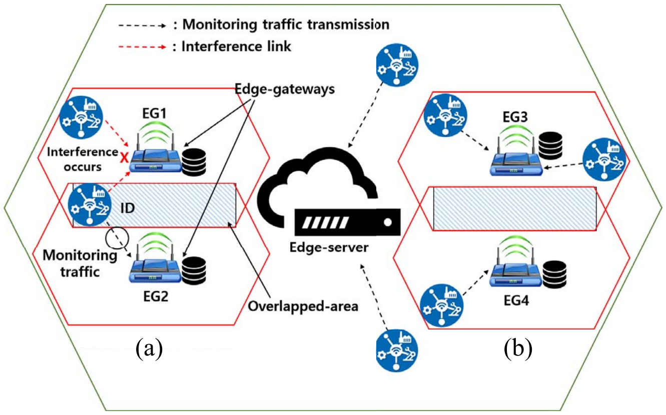


Fig. 2. Interference model between EGs. (a) Interference between EG_1 and EG_2. (b) No interference between EG_3 and EG_4.

based on the reference signal received power (RSRP)] occurs when (26) is satisfied at an EG [23]

$$p_j^i \leq p_k^i + \gamma_i, j \in \mathcal{M}_i, k \notin \mathcal{M}_i \tag{26}$$

where p_j^i and $\mathcal{M}_i$ denote the RSRP of the ith EG for the jth ID and the set of IDs associated with the ith EG, respectively.

In Fig. 2(a), the coverage areas of EG_1 and EG_2 overlap with each other. The ID is deployed in the overlapped region and is originally associated to EG_2. Therefore, an interference occurs with EG_1 if the ID sends a monitoring frame to EG_2.

In this case, the ID checks the RSRP from EG_1, satisfies (26), and accordingly sends the RRC measurement report to EG_2. Subsequently, the ES obtains the control signals including interference information from all the EGs. Hence, the ES constructs the EG interference map as a conflict graph. Conversely, if IDs are absent in the of overlap area covering EG_3 and EG_4 in Fig. 2(b), then interference is not considered.

2) Conflict Graph: We consider a network formed by a set $\mathcal{G}$ of EGs, denoted by $g_i \in \mathcal{G}$. To schedule the EGs, a conflict graph is constructed such that the set of vertices corresponds to $\mathcal{G}$ (the EGs) and two vertices are connected by an edge if the corresponding EGs suffer from mutual interference, which satisfies (26).

The conflict graph is described through its adjacency matrix in which elements $\mathcal{E}_{(j,k)}$ between $g_j \in \mathcal{G}$ and $g_k \in \mathcal{G}$ are defined as follows:

$$\mathcal{E}_{(j,k)} = \begin{cases} 1, & \text{if } g_j \text{ interferes with } g_k \text{ where} \\ & g_j \in \mathcal{G},\ g_k \in \mathcal{G}, \text{ and } j \neq k \\ 0, & \text{otherwise.} \end{cases} \tag{27}$$

Additionally, the set of neighbor nodes of each node is defined as follows:

$$\mathcal{N}(i) \triangleq \left\{ g_a | \mathcal{E}_{(i,a)} = 1 \text{ where } g_a \in \mathcal{G} \right\} \quad \forall g_i \in \mathcal{G}. \tag{28}$$

3) Weight Model: As shown in Fig. 3, each EG includes a queue $Q_i[t]$ for $g_i \in \mathcal{G}$. We set the weight for our algorithm as the queue wherein length evolves based on

$$Q_i[t+1] = \max[0, Q_i[t] - \mu_i[t]] + \lambda_i[t] \tag{29}$$

where $Q_i[t]$, $\mu_i[t]$, and $\lambda_i[t]$ denote the queue backlog size at the EG g_i, the number of bits leaving the queue of the EG g_i, and the number of bits added to the queue of the EG g_i, respectively. For example, g_1 exhibits a higher priority when compared with g_2 in Fig. 3 since the queue backlog size of g_1 exceeds that of g_2.

4) Storage/Computing Capacity: In our edge-based IoT system architecture, we assume that each EG performs preprocessing on data transmitted by an ID and delivers it to

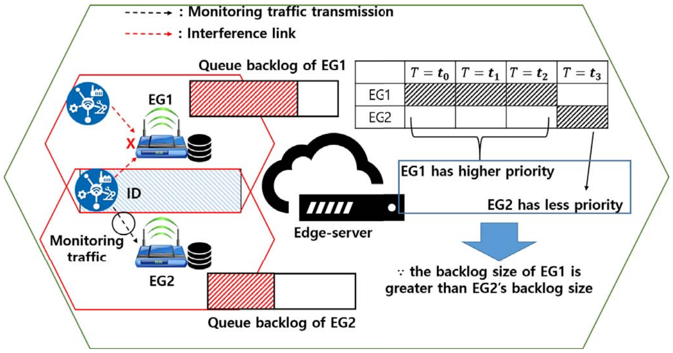
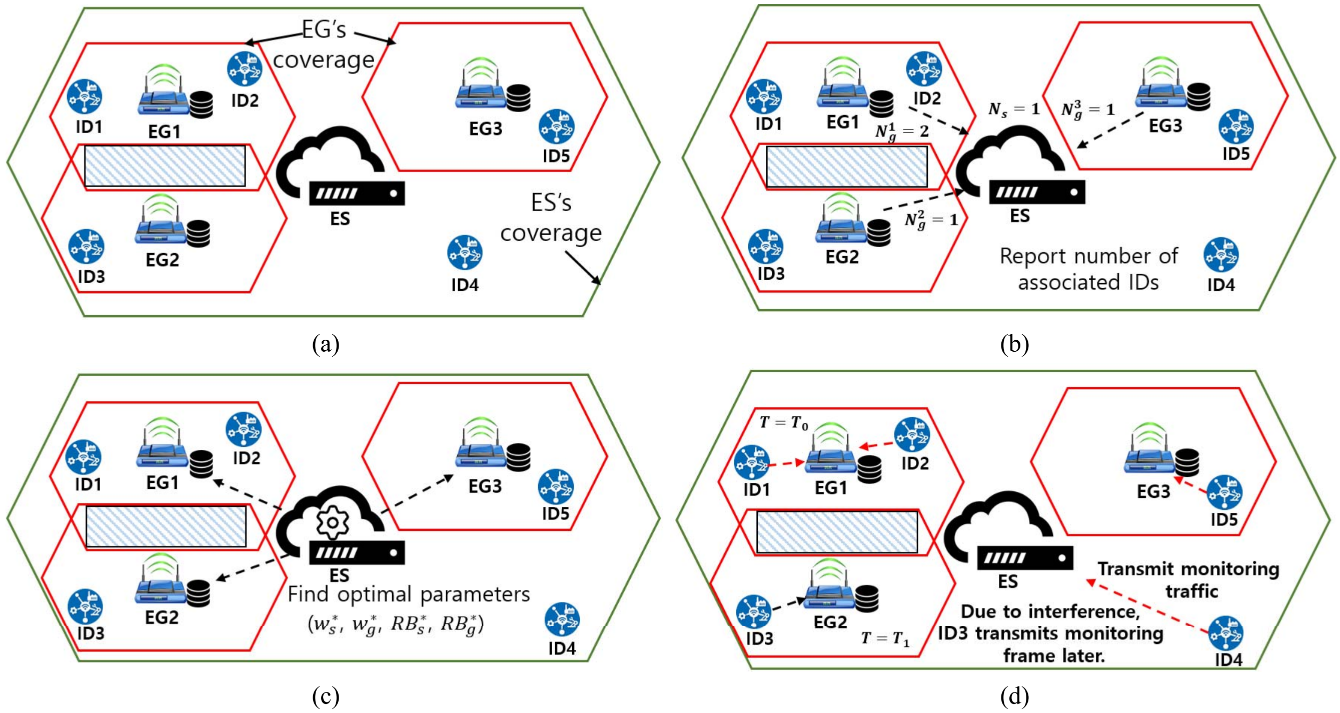


Fig. 3. Example of resource allocation algorithm based on backlog size of EG (workload).

Algorithm 2 Scheduling Algorithm Between EGs

```
Require: $\mathcal{T}, \mathcal{O}$ // Set of timeslots and overlapped groups
Ensure: $\mathcal{I}^*(t), \forall t \in \mathcal{T}$ // Optimal scheduling result
Set temporary variables $i, j, k, k' = 1$;
while $i \leq N_g$ do
    $L_i = 0$; // Initialize phase
end while
Set $i = 1$;
while $i \leq |\mathcal{T}|$ do
    while $j \leq |\mathcal{O}|$ do
        while $k \leq |\mathcal{O}_j|$ do
            Find $g_k$ with max $Q_k[i]$ AND $S_k = 0$;
            Set $\mathcal{S}_k = 1$; // Set the visiting EG identifier
            Set $\mathcal{I}_k = 1$; // Scheduling $g_k$
            while $k' \leq |\mathcal{O}_j|$ AND $k' \neq k$ do
                if $!(\mathcal{I}_k(i) + \mathcal{I}_{k'}(i) + \mathcal{E}_{k,k'} \leq 2, \forall k')$ then
                    Set $\mathcal{I}_k(i) = 0$; // Roll back.
                end if
                $k' \Leftarrow k' + 1$;
            end while
            $k' = 1$;
            $k \Leftarrow k + 1$;
        end while
        $j \Leftarrow j + 1$;
    end while
    Set $\mathcal{S}_k = 1, \forall k$; // Initialize visiting EG identifier
    Update $Q_k[i+1] = \max[0, Q_k[i] - \mu_k[i] * \mathcal{I}_k(i)] + \lambda_k[i], \forall k$; // Update next queue backlog size based on scheduling results
    if $Q_k[i+1] \geq \mathbf{S}_{max}$ then
        Drop workload $Q_k[i+1] - \mathbf{S}_{max}$; //Overflow
    end if
    $i \Leftarrow i + 1$;
end while
```


EG's coverage
ID1
EG1
ID2
EG3
ID5
ES
ES's coverage
ID3
EG2
ID4
$N_s = 1$
$N_g^1 = 2$
$N_g^3 = 1$
$N_g^2 = 1$
Report number of associated IDs
Find optimal parameters
$(w_s^*,\ w_g^*,\ RB_s^*,\ RB_g^*)$
$T = T_0$
$T = T_1$
Transmit monitoring traffic
Due to interference, ID3 transmits monitoring frame later.

(a)

(b)

(c)

(d)

Fig. 4. Exemplary of the proposed scheme process.

the ES with the computing unit and storage.[1] Generally, the computing power and storage capacity of the EG are limited, and thus there is a problem wherein buffer overflow and latency increase if the IDs are concentrated in a few EGs. Therefore, with respect to workload orchestration between EGs, the following constraints should be satisfied:

$$Q_i[t] \le \mathbf{S}_{\max} \quad \forall i, t \tag{30}$$

where $\mathbf{S}_{\max}$ denotes the maximum backlog size of EG. According to (29), (30) can be rewrite as

$$\max[0, Q_i[t] - \mu_i[t]] + \lambda_i[t] \le \mathbf{S}_{\max} \quad \forall i, t. \tag{31}$$

5) Scheduling Algorithm: With respect to scheduling at time $t+1$, the ES compares the backlog sizes of g_j and g_k at time t that are denoted as $Q_j[t]$ and $Q_k[t]$, respectively, when g_j and g_k exhibit an interference relationship. Subsequently, an EG with a higher queue backlog size (i.e., with a higher workload) exhibits a higher priority and receives more resources when compared with other EGs. To mathematically design our scheduling method, the objective involves determining a set of EGs (i.e., nodes of the conflict graph defined above) that subsequently maximize the sum of weights over all possible independent sets. This yields the maximum weight independent set (MWIS) problem as follows:

$$\max: \ \mathcal{F}(\mathcal{I}) \triangleq \sum\nolimits_{\forall g_i \in \mathcal{G}} w_i \mathcal{I}_i, \tag{32}$$

$$\text{s.t.} \ \ \mathcal{I}_j + \mathcal{I}_k + \mathcal{E}_{j,k} \le 2 \quad \forall g_j \in \mathcal{G}, \forall g_k \in \mathcal{G} \tag{33}$$

$$\mathcal{I}_i \in \{0, 1\} \quad \forall g_i \in \mathcal{G} \tag{34}$$

$$\max[0, Q_i[t] - \mathcal{I}_i \mu_i[t]] + \lambda_i[t] \le \mathbf{S}_{\max} \quad \forall i, t \tag{35}$$

where $\mathcal{I}_i$ is defined as

$$\mathcal{I}_i = \begin{cases} 1, & \text{if } g_i \text{ is scheduled where } g_i \in \mathcal{G} \\ 0, & \text{otherwise.} \end{cases} \tag{36}$$

The aforementioned formulation ensures that conflicting EGs are not simultaneously scheduled: if $\mathcal{E}_{j,k} = 0$ (edges are absent between g_j and g_k), then $\mathcal{I}_j + \mathcal{I}_k \le 2$, i.e., both indicator functions are equal to 1. In contrast, if $\mathcal{E}_{j,k} = 1$, then $\mathcal{I}_j + \mathcal{I}_k \le 1$, i.e., a maximum of one of the two indicator functions corresponds to 1.

After solving the MWIS problem, a set of active EGs is obtained, and the actual rates (including all interference caused by the active transmitters on the link receivers) are used to update the queues in the EGs. To solve the MWIS problem, various heuristics and approximation algorithms are proposed since MWIS is a well-known NP-hard problem. One of these methods corresponds to the greedy approach that is adopted henceforth.

Algorithm 2 shows the pseudo-code of the scheduling algorithm for FGs. The scheduling algorithm operates on a constant time slot length $|\mathcal{T}|$. It also sets the algorithm input as a set of timeslots $\mathcal{T}$ and overlapped groups $\mathcal{O}$. Our heuristic algorithm gives priority to the EG with the highest backlog size of the queue (lines 9–22). The proposed algorithm solves the starvation problem of EG by granting priority to EGs with high queue backlog size. After the EG with the highest weight is scheduled, the EG that can be scheduled at the same time as the scheduled FG (i.e., the EG that does not cause interference) is scheduled to the maximum possible extent in a timeslot (lines 13–17). After scheduling the corresponding time slot, the queue backlog size of the scheduled FG is updated, and the scheduling for the next timeslot is performed (lines 19–21).

[1] The architecture is found in a smart manufacturing system or disaster IoT environment [24], [25].

TABLE II
SIMULATION PARAMETERS

Parameters	Value
Noise	3.4233×10^{-8} mW
Operated frequency	2 GHz
Bandwidth	20 MHz
ES transmission power	40 dBm
EGs transmission power	20 dBm
Number of total IDs	10 to 1000
Number of EGs	10 to 100
Number of IDs associated to ES	20
Number of total resource blocks	111
RSRP offset	1 dB

If the workload exceeds the storage capacity of the EG, the overflow will occur (lines 25–27).

Fig. 4 shows an example of the proposed scheme process. The example assumes an environment with 1 ES, 3 EGs, and 5 IDs wherein all IDs except ID4 are associated with an EG [Fig. 4(a)]. Prior to scheduling, ES receives the number of IDs (N_g^m) associated to itself from each EG [Fig. 4(b)]. Based on the received information, the ES computes the optimal parameters ($w_s^*, w_g^*, RB_s^*,$ and RB_g^*) and distributes the number of allocated resource blocks to the EG [Fig. 4(c)]. Finally, the ES also performs the scheduling through Algorithm 2 for the overlapped area (the area corresponding to FG 1 and FG 2 in the figure) and communicates it based on the scheduling result [Fig. 4(d)].

In the next section, we evaluate the performance of the edge-based IoT system by using the scenario of the models categorized above. Specifically, we analyze how the interference between overlapping coverage regions reduces the overall network performance by comparing several cases.

V. PERFORMANCE EVALUATION

In this section, we conduct simulations to verify the performances of the network models. The models are implemented in the OPNET simulator. In the simulation, we consider an environment in which an edge device operating in cellular networks is deployed. Additionally, we assumed an edge-based Samsung Electronics' Vietnam Semiconductor smart manufacturing system for simulation. In a smart manufacturing system, multiple Raspberry Pi-grade EG (approximately 1.0 GHz CPU and 4 GB storage) are deployed and are connected via Ethernet with ES (approximately 4.5 GHz CPU and 1 TB storage). Each EG approximately accommodates 10 IDs. Each ID randomly generates monitoring packets, and the arrival period ranges from 1 ms to 100 ms, and the data size ranges from 100 bytes to 100 kB. More detailed simulation parameters are listed in Table II.

To increase the efficiency of the simulations, the operating frequency is set as 2 GHz and bandwidth is configured as 20 MHz. The EGs are randomly deployed in the IoT system, and all IDs are randomly deployed in the coverage area of each associated EG. Conversely, a few parameter values are fixed such as the transmission powers at 40 dBm and 20 dBm for the ES and EGs, respectively.

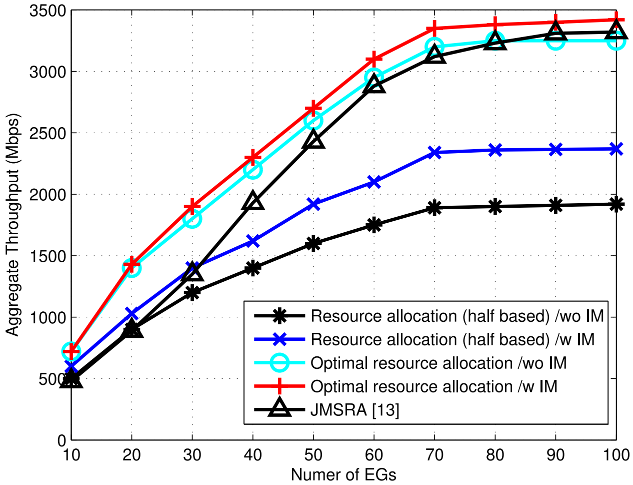


Fig. 5. Total aggregate throughput versus the number of EGs.

We calculate the UE's signal to interference noise ratio (SINR) based on the received signal strength with the maximum interference from other BSs. Based on the value of the SINR (in dB), we can set the channel quality indicator and a suitable coding rate for the channel.

For comparison, we use the half-based scheme that allocates frequency resources equally to the ES and EGs and our proposed optimal resource allocation scheme. We compare an environment where the proposed IM technique for EGs is used with an environment where it is not used. Furthermore, we compare the performance of the proposed technique with existing resource allocation scheme [13]. Additionally, we compare an environment where the proposed IM technique for EGs is used with an environment where it is not used.

For the performance evaluation, we use the following metrics.

1) *Aggregate Throughput:* total data traffic in bits transferred successfully from IDs to EGs/ES divided by time.
2) *Buffer Overflow Ratio of EGs:* The number of frames lost due to buffer overflow divided by the total number of frames received for all EGs.
3) *Data Latency:* Average latency of data transmitted from IDs to the ES.
4) *Amount of Successfully Received Data (Availability):* The number of data frames successfully transmitted from IDs to the ES.
5) *Fairness Index of EG's Backlog Size:* The square of the average of x_i divided by the average of x_i^2 where N denotes the number of EGs and x_i denotes the backlog size for the EG i

$$J(x_1, x_2, \ldots, x_N) = \frac{\left(\sum_{i=1}^{N} x_i\right)^2}{N \sum_{i=1}^{n} x_i^2}.$$

Fig. 5 illustrates the aggregate throughput where the data is generated from all IDs and is transferred via the EG or directly to the ES. As shown in the figure, the aggregate throughput increases when the number of EGs increases. This is because

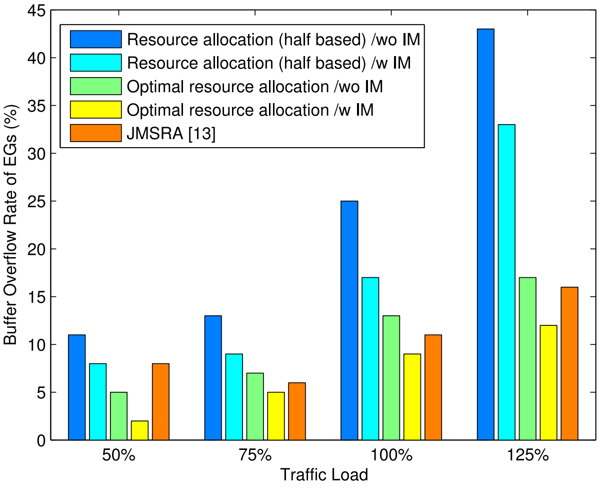


Fig. 6. Buffer overflow rate of EGs versus traffic load.

frequency resources are shared among the EGs since their coverage area is low. Thus, it is optimal to allocate the maximum possible number of resource blocks to the EG in terms of the overall network throughput. However, when the number of EGs exceeds 70, the throughput is saturated due to signal attenuation and because it exceeds the computing capacity of the EGs. In [13], the throughput is low when the number of EGs is low although the slope rapidly increases when the number of EGs increases. In our experiment, it is assumed that the IDs are distributed deployed, and thus IDs cannot be connected to EGs in environments with fewer EGs. In [13], the situation where the IDs are connected to the ES is not considered, and thus it is observed as a low throughput in environments where fewer EGs are deployed. Additionally, the proposed resource optimization scheme exhibits a performance improvement of approximately 42% compared with the conventional resource allocation method for the EGs and ES. This is because the ES determines the optimal w_g^* and w_s^* in the proposed algorithm through the number of associated IDs with EGs and ES. Therefore, the ES allocates more resource blocks to EGs in environments in which their number is high. Fig. 6 shows the buffer overflow ratio of EGs versus traffic load. As shown in the figure, the proposed optimal resource orchestration scheme has less packet loss due to buffer overflow than other existing schemes. This is because, the proposed scheme computes the backlog size of each EG and allocates as many resources as it can accommodate.

When the proposed IM scheme is applied to minimize the interference between EGs, the performance improvement is approximately 6% in the optimal resource allocation scheme and 20% in the equivalent resource allocation scheme. In the former case, a high portion of resources is allocated to the EG, and a limited amount of resources is allocated to the EG in the latter case. Thus, the average queue exhibits a high backlog size in the latter case, and this causes frequent interference due to the high amount of data that the EG transmits. Therefore, it is confirmed that the proposed IM method significantly reduces the interference between the EGs in the latter case.

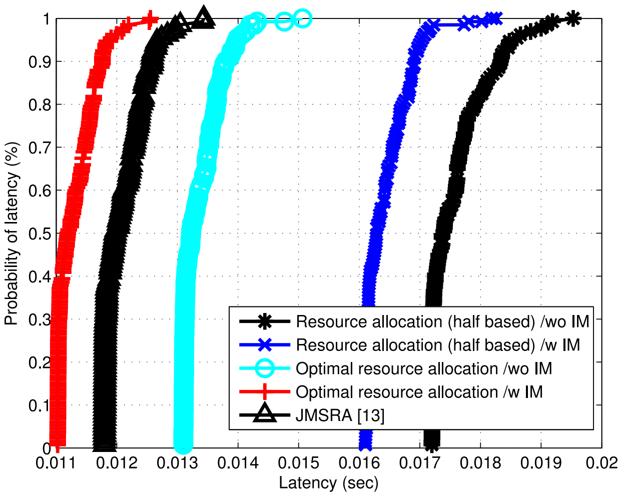


Fig. 7. Cumulative distribution function of latency.

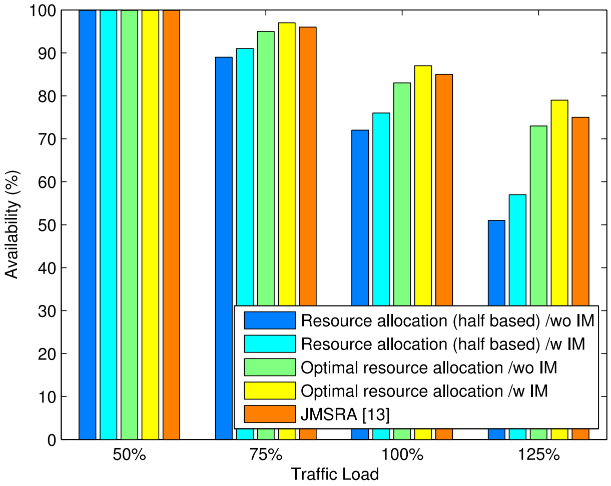


Fig. 8. Total received packets of the ES versus traffic load.

Fig. 7 illustrates the cumulative distribution function of the latency. In the simulation, when the ID sends monitoring traffic to the EG, a scenario is assumed in which the data preprocessing is performed and sent to the ES. Furthermore, we assume that the preprocessing time is approximately 10 ms. As shown in the figure, although the monitoring traffic is transmitted to the ES through the EG, the latency of approximately 10 ms is reduced by applying the proposed resource optimization algorithm when compared with that of the existing resource allocation scheme [13]. The latency value is lower than the latency requirement of 100 ms for edge computing applications in industrial environments for EdgeXFoundary, and this indicates that the proposed scheme is suitable for IoT systems.

Fig. 8 shows the amount of successfully received data at the ES divided by the amount of data transmitted from the IDs versus the traffic load. The x-axis represents the normalized traffic load. For example, a traffic load of 100% indicates that the incoming traffic requires the total rate at each ID in the IoT system to be identical to the average capacity of a single link. As shown in the figure, the proposed resource allocation scheme with IM successfully transmits approximately 87% of data traffic to the ES when the traffic load is 100%. Loss occurs because an overflow is caused by a lack of computing capacity

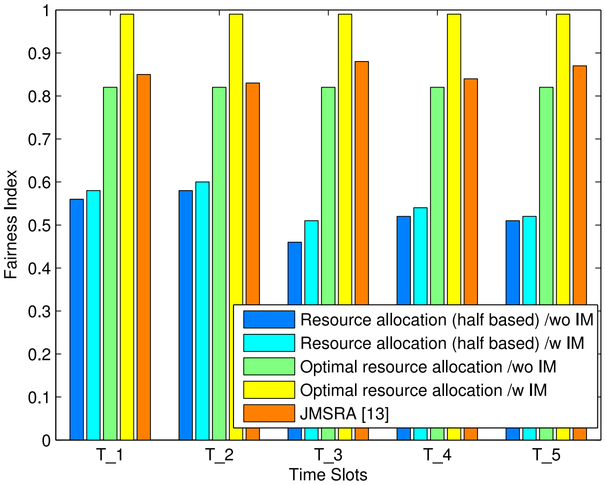


Fig. 9. Jain's fairness index for each scheme.

(queue size) when a certain EG exhibits several associated IDs. If we assume an environment with an extremely high computing capacity, then we expect that the problem will be resolved. In [13], a resource allocation exists to increase the throughput of the EGs, and thus resources are potentially not allocated to the ES. Thus, IDs connected to the ES do not transmit data, and it exhibits lower results when compared with the proposed technique.

Fig. 9 illustrates the fairness index of the queue backlog sizes of the EGs. The fairness index exhibits a value from 0 to 1. Furthermore, the closer the value is to 1, the more similar the queue backlog size is for all EGs. In the simulation results, it is observed that the proposed resource allocation scheme with IM achieves a value that is extremely close to 1, and this indicates that more frequency resources are allocated to EGs with several associated IDs and processing is faster. The workload of each EG is not considered in the technique without the resource optimization algorithm, and thus it is observed that the worst-case fairness value approximately corresponds to 0.6. In [13], resource allocation is performed by considering the dynamic traffic arrivals of each EG, and thus resources are concentrated in a few EGs. The tendency potentially corresponds to an excellent choice to increase overall network throughput although it can cause starvation problem for a few IDs/EGs. Additionally, the EG exhibits limited computing power, and thus it can lead to results that are not optimal (for e.g., buffer overflow and exceeding the service deadline) in terms of latency and availability if the workload is centered on specific EGs. The resource imbalance between the EGs is caused by the failure of an EG in the worst case, and this can lead to the failure of the entire IoT system. For example, the aforementioned problem can involve serious consequences in a scenario where UAVs or mobile BSs are installed in a disaster IoT environment to determine an emergency patient.

VI. Conclusion

In this paper, we introduced a wireless edge-based IoT system and proposed an optimal frequency resource allocation scheme for the ES and EGs. Based on the number of IDs associated with the EGs and the ES, we proposed a Lagrangian form-based objective function that determines optimal resource allocation ratios and solves the objective function of KKT conditions. Additionally, we proposed a technique for reallocating resources among EGs to determine the interference between EGs. The EG resource reallocation algorithm modeled the interference between EGs as an MWIS problem and solved it via a heuristic algorithm based on the workloads of EGs. Furthermore, we verified the performances of the models after applying the optimal resource allocation scheme and IM technique. The results of the experiments indicated that the technique of sharing resources between the ES and EGs using the proposed IM scheme exhibited an optimal performance in terms of the aggregate throughput, delay, availability, and operating fairness.

References

[1] P. Hu, H. Ning, T. Qiu, Y. Zhang, and X. Luo, "Fog computing based face identification and resolution scheme in Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 1910–1920, Aug. 2016.

[2] M. Peng, S. Yan, K. Zhang, and C. Wang, "Fog-computing-based radio access networks: Issues and challenges," *IEEE Netw.*, vol. 30, no. 4, pp. 46–53, Jul./Aug. 2016.

[3] D. Georgakopoulos, P. P. Jayaraman, M. Fazia, M. Villari, and R. Ranjan, "Internet of Things and edge cloud computing roadmap for manufacturing," *IEEE Cloud Comput.*, vol. 3, no. 4, pp. 66–73, Jul./Aug. 2016.

[4] H. S. Kang *et al.*, "Smart manufacturing: Past research, present findings, and future directions," *Int. J. Precision Eng. Manuf. Green Technol.*, vol. 3, no. 1, pp. 111–128, Jan. 2016.

[5] O. Stanze and A. Weber, "Heterogeneous networks with LTE-advanced technologies," *Bell ns Techn. J.*, vol. 18, no. 1, pp. 41–58, Jun. 2013.

[6] B. Holfeld *et al.*, "Wireless communication for factory automation: An opportunity for LTE and 5G systems," *IEEE Commun. Mag.*, vol. 54, no. 6, pp. 36–43, Jun. 2016.

[7] T. Taleb *et al.*, "On multi-access edge computing: A survey of the emerging 5G network edge architecture & orchestration," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1657–1681, May 2017.

[8] D. Zeng, L. Gu, S. Guo, Z. Cheng, and S. Yu, "Joint optimization of task scheduling and image placement in fog computing supported software-defined embedded system," *IEEE Trans. Comput.*, vol. 65, no. 12, pp. 3702–3712, Dec. 2016.

[9] K. Zhang *et al.*, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896–5907, Aug. 2016.

[10] Y. Sun, S. Zhou, and J. Xu, "EMM: Energy-aware mobility management for mobile edge computing in ultra dense networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 11, pp. 2637–2646, Nov. 2017.

[11] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. S. Quek, "Offloading in mobile edge computing: Task allocation and computational frequency scaling," *IEEE Trans. Commun.*, vol. 65, no. 8, pp. 3571–3584, Aug. 2017.

[12] H. Zhang *et al.*, "Computing resource allocation in three-tier IoT fog networks: A joint optimization approach combining Stackelberg game and matching," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1204–1215, Oct. 2017.

[13] Y. Mo, M. Peng, H. Xiang, Y. Sun, and X. Ji, "Resource allocation in cloud radio access networks with device-to-device communications," *IEEE Access*, vol. 5, pp. 1250–1262, Feb. 2017.

[14] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 5994–6009, Sep. 2017.

[15] L. Ni, J. Zhang, C. Jiang, C. Yan, and K. Yu, "Resource allocation strategy in fog computing based on priced timed Petri nets," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1216–1228, Oct. 2017.

[16] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.

[17] Q. Song, X. Wang, T. Qiu, and Z. Ning, "An interference coordination-based distributed resource allocation scheme in heterogeneous cellular networks," *IEEE Access*, vol. 5, pp. 2152–2162, Jan. 2017.

[18] C. Chen, R. Ratasuk, and A. Ghosh, "Downlink performance analysis of LTE and WiFi coexistence in unlicensed bands with a simple listen-before-talk scheme," in *Proc. IEEE Veh. Technol. Conf. (VTC Spring)*, 2015, pp. 1–5.

[19] A. Mishra, S. Banerjee, and W. Arbaugh, "Weighted coloring based channel assignment for WLANs," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 9, no. 3, pp. 19–31, 2005.

[20] Z. Zhou, K. Ota, M. Dong, and C. Xu, "Energy-efficient matching for resource allocation in D2D enabled cellular networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 5256–5268, Jun. 2017.

[21] S. M. Alamouti and S. Kallel, "Adaptive trellis coded multiple-phased shift keying for Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 42, no. 6, pp. 2305–2314, Jul. 1994.

[22] A. A. Haija and C. Tellambura, "Small-macro cell cooperation for HetNet uplink transmission: Spectral efficiency and reliability analyses," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 1, pp. 118–135, Jan. 2017.

[23] K. I. Pedersen, Y. Wang, S. Strzyz, and F. Frederiksen, "Enhanced inter-cell interference coordination in co-channel multi-layer LTE-advanced networks," *IEEE Wireless Commun.*, vol. 20, no. 3, pp. 120–127, Jun. 2013.

[24] W. Na *et al.*, "Directional link scheduling for real-time data processing in smart manufacturing system," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3661–3671, Oct. 2018, doi: 10.1109/JIOT.2018.2865756.

[25] T. Higashino, H. Yamaguchi, A. Hiromori, A. Uchiyama, and K. Yasumoto, "Edge computing and IoT based research for building safe smart cities resistant to disasters," in *Proc. IEEE Distrib. Comput. Syst. (ICDCS)*, 2017, pp. 1729–1737.

Woongsoo Na received the B.S., M.S., and Ph.D. degrees in computer science and engineering from Chung-Ang University, Seoul, South Korea, in 2010, 2012, and 2017, respectively.

He was an Adjunct Professor with the School of Information Technology, Sungshin University, Seoul, from 2017 to 2018. He is currently a Senior Research Engineer with the Electronics and Telecommunications Research Institute, Daejeon, South Korea. His current research interests include mobile chargers, directional mmWave systems, wireless mobile networks, 5G, and beyond 5G.

Seonmin Jang received the B.S. degree in electric, electronic engineering and the M.S. degree in computer science from Chung-Ang University, Seoul, South Korea, in 2015 and 2017, respectively.

He is currently a Network System Software Engineering with Samsung Electronics, Suwon, South Korea. His current research interests include LTE, LAA, network softwarization, and 5G.

Yoonseong Lee received the B.S. and M.S. degrees in computer science from Chung-Ang University, Seoul, South Korea, in 2013 and 2015, respectively, where he is currently pursuing the Ph.D. degree in computer science.

His current research interests include wireless communication, routing protocols, and blockchain application.

Laihyuk Park received the B.S., M.S., and Ph.D. degrees in computer science and engineering from Chung-Ang University, Seoul, South Korea, in 2008, 2010, and 2017, respectively.

He is currently an Assistant Professor with the Da Vinci College of General Education, Chung-Ang University. From 2011 to 2016, he was a Research Engineer with Innowireless, Bundang-gu, South Korea. His current research interests include demand response, smart grid, electric vehicles, Internet of Things, fog computing, edge computing, and wireless networks.

Nhu-Ngoc Dao received the B.S. degree in electronics and telecommunications from the Posts and Telecommunications Institute of Technology, Ho Chi Minh City, Vietnam, in 2009 and the M.S. degree in computer science from Chung-Ang University, Seoul, South Korea, in 2016, where he is currently pursuing the Ph.D. degree in computer science.

His current research interests include network security, network softwarization, fog/edge computing, and Internet of Things.

Sungrae Cho received the B.S. and M.S. degrees in electronics engineering from Korea University, Seoul, South Korea, in 1992 and 1994, respectively, and the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2002.

He was an Assistant Professor with the Department of Computer Sciences, Georgia Southern University, Statesboro, GA, USA, from 2003 to 2006, and a Senior Member of Technical Staff with the Samsung Advanced Institute of Technology, Kiheung, South Korea, in 2003. He is currently a Full Professor with the School of Computer Science and Engineering, Chung-Ang University, Seoul. From 2012 to 2013, he held a visiting professorship with the National Institute of Standards and Technology, Gaithersburg, MD, USA. He was a Research Staff Member with the Electronics and Telecommunications Research Institute, Daejeon, South Korea, from 1994 to 1996. His current research interests include wireless networking, ubiquitous computing, performance evaluation, and queuing theory.

Dr. Cho has been an Editor of *Ad Hoc Networks* (Elsevier) since 2012 and has served numerous international conferences as an Organizing Committee member such as IEEE SECON, ICOIN, ICTC, ICUFN, TridentCom, and IEEE MASS.