Received 19 January 2023; revised 20 February 2023; accepted 12 March 2023. Date of publication 21 March 2023; date of current version 3 April 2023.

Digital Object Identifier 10.1109/OJCOMS.2023.3260165

Digital Twin-Aided Orchestration of Mobile Edge Computing With Grant-Free Access

NIKOS A. MITSIOU[1] (Student Member, IEEE),
VASILIS K. PAPANIKOLAOU[1] (Graduate Student Member, IEEE),
PANAGIOTIS D. DIAMANTOULAKIS[1] (Senior Member, IEEE),
TRUNG Q. DUONG[2] (Fellow, IEEE), AND GEORGE K. KARAGIANNIDIS[1,3] (Fellow, IEEE)

[1]Wireless Communication and Information Processing Group, Department of Electrical and Computer Engineering,
Aristotle University of Thessaloniki, Thessaloniki 54636, Greece

[2]School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, BT7 1NN Belfast, U.K.

[3]Cyber Security Systems and Applied AI Research Center, Lebanese American University, Beirut 1102 2801, Lebanon

CORRESPONDING AUTHOR: N. A. MITSIOU (e-mail: nmitsiou@ece.auth.gr)

This work was supported by the European Union's Horizon 2020 Research and Innovation Programme under Grant 957406.
The work of Vasilis K. Papanikolaou was supported by Greece and the European Union through the IKY Scholarship
Programme for Ph.D. candidates in the Greek Universities.

ABSTRACT Digital twin-aided (DT) edge computing is investigated, where users utilize grant-free random access with adaptive rate to offload their tasks to the edge server. A novel, with lower implementation complexity, probabilistic partial offloading scheme is introduced, while each device is assumed to have an infinite buffer to store its tasks. The aim of the proposed work is to minimize the average delay of the partial offloading. To that end, the average delay of waiting in the queue, the delay of offloading, and the local computation delay are extracted by using queuing theory tools. Then, the non-convex problem of minimizing the average delay of all clients is formulated, while taking into account DT imperfections. Successive convex approximation (SCA), alternating optimization (AO), and various algebraic manipulations are utilized to transform the problem into an equivalent convex problem with tractable solution. Finally, simulation results showcase the value of the proposed analysis and offer important insights for the proposed DT-aided edge network. Specifically, the proposed partial offloading scheme is shown to be more delay efficient compared to both local computing and full offloading, particularly, for greater task generation rates at the users. Also, the impact of the DT imperfections at the average delay is shown to be more notable as the number of users, or the tasks' size, increases.

INDEX TERMS Grant-free random access, 6G, mobile edge computing, digital twin.

I. INTRODUCTION

NEXT-GENERATION Internet of Things (IoT) networks are expected to rely on smart devices at which edge intelligence and computing can be fully realized, e.g., smartphones, vehicles, machines, and robots [1], [2]. Such a device-centric network poses new challenges and requirements on the design and operation of wireless communication since smart devices will not only generate or exploit data, but will actively join the network management [1], [2]. Moreover, the large number of potentially active devices complicates resource allocation, therefore contention-based protocols attracted attention recently, as a way to avoid too many resources remaining idle due to intermittent traffic [3], [4].

Toward reducing the access delay and signal overhead of traditional contention-based protocols, the 3rd generation partnership project (3GPP) in Release 16 of the 5th generation new radio (5G NR) proposed a two-step random access scheme, namely grant-free (GF) access [5]. The main idea behind GF access is that an active device does not wait for a response from the base station (BS) after transmitting its preamble (1st step), but immediately transmits its data packet (2nd step). Therefore, the handshaking process between the user and the base station is avoided, consequently reducing the associated signalling overhead. The key challenge for GF transmissions is contention, as multiple users may choose to transmit at the same channel resources at the same time.

However, future networks as they are shaped by the sixth-generation (6G) concept [6] will need to support novel use cases, for example, virtual, augmented, and extended reality (VR/AR/XR), tactile Internet, intelligent power grids and smart cities [1], [2], [7]. Those use cases will require scalable wireless sensor networks, but will also demand intensive computing and medium to high data rates [1], [7]. Moreover, intelligent functionalities will be extended to the edge nodes, due to their advanced computational capabilities, thus enabling the convergence of artificial intelligence (AI), communications, and edge computation [1]. As such, edge computing [8] architectures will provide intelligence physically closer to the end users [8]. This can significantly improve the end-to-end latency, especially for users who repeatedly offload intensive tasks to the server.

Furthermore, recently, edge computing has been combined with the emerging technology of digital twins (DTs). A DT is a comprehensive software representation of an individual physical object or system, that includes its real-life properties, conditions and behaviours [9]. By combining mobile edge computing (MEC) and DT, the MEC server status and the status of the end devices, such as the distance of the end users from the MEC server, the average task arrival at the users' buffers or the CPU frequency at the edge servers can be directly transferred to the network's orchestrator. Then, based on that knowledge, the orchestrator provides the physical network with intelligent and optimal decisions [9]. Therefore, the DT's goal is to capture the physical features of the network, while the orchestrator's aim is to develop an optimal strategy based on those features. Due to the fact the a DT continuously evolves, alongside its physical entity, it provides the orchestrator with an improved view of the underlying physical network.

A. LITERATURE REVIEW

GF transmission schemes have gained considerable attention [3], [4], [5], [10], [11], [12], [13], [14]. In [4], [10] the concept of GFRA was presented for massive machine-type communication (mMTC) and ultra-reliable low-latency communication (URLLC), while its combination with non-orthogonal multiple access (NOMA) was discussed in [11]. Furthermore, GF-NOMA was investigated in [13] and [15]. In [3], two reliability-enhancing solutions were proposed for GFRA aided URLLC applications. The first proposes re-transmissions over shared resources, whereas the second proposal incorporates GF-NOMA with overlapping transmissions being resolved through the use of advanced receivers. Moreover, in [12], URLLC with multiple grants was designed. Finally, in [14], the success probability of GFRA with massive multi-input multi-output was examined.

MEC has also been extensively studied, often by taking into account congestion occurring at the buffers of the users or at the buffers of the MEC server [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26]. However, the MEC state-of-the-art cannot be generalized for GF transmissions, since continuous-time queueing models are adopted. For instance, in [16], the M/M/c/c queuing system was exploited and a holistic QoS-aware framework for Industrial IoT systems was designed, whereas in [17] a heuristic scheduling model was designed to maximize the offloading energy and execution efficiency of an Erlang queueing MEC system. Similarly, in [27] three Erlang based queueing models were applied, one at the mobile users, one at the edge server and one at the cloud server. However, their model is based on orthogonal multiple access, which is fundamentally different to the GF-access. Moreover, the delay of a MEC serving multi-class users, based on continuous-time queueing, was studied in [18], while in [20] a closed-form water-filling computation offloading solution was proposed to investigate the average delay. Furthermore, in [21], a distributed task offloading scheme was investigated with consideration to the upper layer queueing dynamics and the lower-layer coupled wireless interference. Also, in [25], a stochastic buffer-aided relay-assisted MEC was examined.

Furthermore, in [28], a cross-layer MEC design was studied for URLLC and enhanced mobile broadband (eMBB) services with short packet transmissions. The delay of the system was analyzed, however, the duration of the data transmission slot and partial offloading were not considered, while the channel access was not GF-aided. Furthermore, in [29], the users' power consumption under partial offloading was minimized, while statistical constraints were imposed on task queue lengths by applying extreme value theory. Moreover, in [30], a deep reinforcement learning algorithm was designed to study the joint optimization of task offloading for a MEC system with application to AR. In addition, in [31], resource allocation was investigated for URLLC vehicular edge computing, while in [32] the trade-off between latency and reliability in URLLC MEC was examined. Nonetheless, queueing delay was not considered in any of [30], [31], [32].

Moreover, several studies have aimed on designing DTs combined with MEC [9], [33], [34], [35], [36], [37], [38], where the MEC-DT concept is utilised to extract optimal network orchestration strategies based on real-time data. For instance, in [33], [34], [35], [36], deep reinforcement learning (DRL)-based algorithms were proposed to train the DT of the MEC network for making offloading decisions, edge association and resource allocation, thus increasing the network's performance. In addition, [37] and [38] minimize end-to-end latency in a DT-aided MEC system, while also considering deviations between the DT and its physical counterpart. Finally, in [9] a secure and latency-aware edge computing architecture was designed. Despite their merits, none of the aforementioned works have studied the integration of DT and MEC for GF access schemes, which is a fundamental for the next generation IoT.

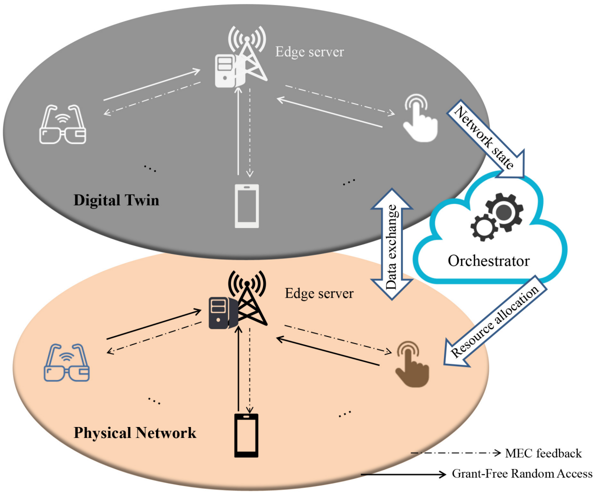


FIGURE 1. The proposed DT-aided architecture.

B. MOTIVATION AND CONTRIBUTIONS

Inspired by the above as well as the advantages of DT-aided edge computing and GF access protocols, this work aims to combine DT-aided MEC with grant-free access. To the authors' best knowledge, this is the first work that proposes DT-aided edge computing with GF access and partial offloading. Next-generation IoT's role will be to improve intelligence in physical systems, such as smart cities, transportation and power grids by monitoring their unique characteristics [1], [2]. Those data will then be processed, either locally, or by the edge in order to improve the system's intelligence. Since IoT devices produce tasks sporadically, they do not need to access the wireless medium constantly. On top of that, the large number of IoT devices makes scheduling quite complicating, as wireless resources are limited to stay idle due to intermittent traffic. Thus, GF access is a strong candidate to enable the communication between next-generation IoTs and the MEC server.

Moreover, the majority of the existing literature on MEC or DT-aided MEC, considers the partial offloading as a procedure which splits the size of a task into two parts, which in practice may not be straightforward to implement, since separating a task into smaller subtasks depends on the task's structure and functions. In our analysis, a task is either transmitted to a MEC server, or it is locally processed, which offers lower implementation complexity, and it is more appropriate for next-generation IoT. Therefore, the proposed partial scheme is a probabilistic binary offloading scheme, which adjusts the ratio of the tasks transmitted to the MEC server and the number of the tasks processed locally, according to the channel conditions, the density of the network, the available preambles, etc. Furthermore, in contrast to the state of the art on discrete-time queuing theory, where instant packet transmissions on error-free channels are assumed, in our analysis, an error-prone channel is considered, while the data transmission duration is also optimized. The contributions of the paper are summarized below:

- A DT-aided MEC system with GFRA is designed under a novel partial offloading. The partial binary offloading is a probabilistic binary offloading scheme, that can be visualized as a switch, which with probability θ sends a packet to the transmission buffer and with $1-\theta$ probability sends a packet to the local computing buffer. Closed-form delay expressions are extracted, under the assumption of infinity buffer capacity. In contrast to the state-of-the-art on discrete-time queuing theory, error-prone channel conditions are studied, while the transmission delay is also considered.
- The average delay until a task is processed, either locally or by the MEC server, is minimized. Moreover, the data transmission phase duration is also optimized. It should also be noted that the formulated optimization problem can be easily modified so that the devices' power consumption is minimized, subject to delay constraints. To tackle the non-convex delay minimization problem, an efficient algorithm is proposed, which utilizes successive convex approximation (SCA) and alternating optimization (AO).
- Numerical results illustrate the superiority of the proposed schemes over full-binary approaches. Moreover, valuable insights about the DT-aided MEC network are extracted, while the convergence of the proposed optimization algorithm is illustrated.

C. STRUCTURE

In Section II, the proposed system model is introduced. In Section III, a brief stability analysis is presented for the buffers, followed by Section IV, where the delay analysis for the buffers is demonstrated, while in Section V, the computation model of the DT is presented. In Section VI, the delay minimization problem is formulated and solved. In Section VII, numerical results are shown and different insights are discussed. Finally, in Section VIII a conclusion is drawn.

II. SYSTEM MODEL

We consider a DT-empowered MEC server-client system model, which consists of N devices and one MEC server, as shown in Figure 1. The architecture relies on two layers, the physical layer and the DT layer. The physical layer consists of all the physical components of the network, such as devices, edge servers, base stations, while also taking into account each components limitations regarding the hardware, the transmission power etc. Each device k lies in a distance d_k and employs GFRA to communicate with the server and to transmit computationally expensive tasks. The communication between the users and the MEC server is separated into two phases, a preamble phase and a data transmission phase. The preamble phase duration is much shorter than the data phase duration, while the available number of preambles is denoted as L_p. The preambles are orthogonal, therefore a collision between the users can only happen during the preamble phase. A user that is active, uniformly chooses

TABLE 1. List of symbols and notations.

Notation list					
N	No. of devices	k_k	CPU energy consumption constant	L_p	Available preambles
q_k	Access probability of k-th user	P_k	Transmit Power of k-th user	f_k	CPU cycle of k-th user
λ_k	Task rate of k-th user	L_k	Size of task (bits)	X_k	No. of required CPU cycles
θ_k	Offloading factor	DT_k	Digital Twin of k-th user	τ	Time slot duration
μ	Probability of successful transmission	l_k	Average queue length	R_k	Transmission rate
g	Queue length probability	N_0	Noise spectral density	Ω_k	Path loss of k-th user
$d_{\mathrm{cmp},k}$	Computation latency	$d_{t,k}$	Transmission delay	$d_{\mathrm{gap},k}$	Computing latency deviation
$d_{q,k}$	Full offloading queueing delay	$d_{\mathrm{cq},k}$	Local computation queueing delay	$d_{\mathrm{mec},k}$	MEC computing delay
$d_{p,k}$	Partial offloading delay k	$d_{\mathrm{o},k}$	Full offloading delay	n	DT imperfection

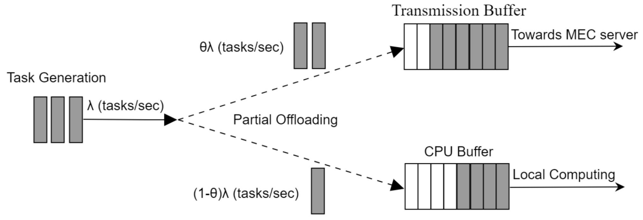


FIGURE 2. Partial offloading diagram.

one of the available preambles and transmits it to the MEC server. Then, immediately, the user enters the data transmission phase, without waiting for a response from the MEC server. The access probability, i.e., the probability that a user becomes active and gets access to the channel is defined as q_k. Also, let P_k represent the transmission power of the data transmission phase. Due to the fact that a user is not always active, the total number of active users is unknown. In Table 1, the list of symbols and notations is presented.

Moreover, a device has the ability to perform basic computations by itself and hence a full offloading is not mandatory. We assume that each device k generates tasks at the rate of λ_k tasks per second, where each task has a size of L_k bits. Also, the devices' computing capabilities are described by their processors' CPU cycles per second, f_k. Each task is described by its size and a required number of CPU cycles per bit, X_k. The splitting factor $\theta_k \in [0, 1]$ represents the percentage of the tasks which are transmitted to the MEC server and thus, $1-\theta_k$ represents the percentage of the tasks that will be computed locally. We assume that each device has a buffer in order to store its generated tasks. Since the devices utilize a partial offloading strategy, it is convenient to assume that the storage capability of the devices is separated into two buffers, one for the tasks that are waiting to be transmitted to the MEC server and one for the tasks that are in line to be computed locally, as shown in Figure 2. Both buffers are considered to have infinite capacity. Practically, tasks cannot be afforded to be lost, therefore the assumption of an infinite buffer is meaningful.

The DT layer is an exact replica of the underlying physical layer which takes into account all hardware components of the physical devices and the network topology. The DT interacts with the physical layer in order to gather data which aid to improve the digital representation of the physical world and capture physical changes in real-time. Based on the information provided by the DT, an orchestrator efficiently manages the networks' available resources. The DT of the physical layer is represented as $DT = \{(\mathcal{M}, \tilde{\mathcal{M}}), (\mathcal{N}, \tilde{\mathcal{N}})\}$, including the MEC server and the N users. Without loss of generality, it is assumed that the DT of the MEC server is perfect due to the superiority of the wired backhaul channels. Therefore, the set containing all the parameters and variables to describe the physical MEC server, $\mathcal{M}$ is exactly the same as its DT counterpart, $\tilde{\mathcal{M}}$. The DT of the k-th user is denoted as $DT_k = \{(\mathcal{N}, \tilde{\mathcal{N}})\}$. $\tilde{\mathcal{N}}$ is the set containing all the parameters and variables describing the physical users. As in [37], a deviation n of available CPU frequency is used to describe the deviation between real users and their digital counterparts, which can be either positive or negative. The deviation n will be assumed known in advance [37]. The following assumptions hold in our analysis:

- The channel is not error-free. A packet can be lost either due to a collision during the preamble phase or due to an outage event caused by the channel's conditions during the data transmission phase.
- The preamble phase duration is negligible compared to the data phase duration, due to the preambles' small size. Therefore, the data transmission phase approximately captures the whole duration of a time slot, which equals τ seconds. Due to synchronization issues, the time slot duration is equal for all users in the system and the duration of the time slot is limited by the worst user or the user with the biggest task [4].
- A packet that failed to be transmitted, will be retransmitted at the next time slot with the same probability q_k. Also, the failure or the success of a packet is considered known instantly.
- Tasks are generated with an average rate of λ_k tasks per second. The communication time is divided into time slots of duration τ, therefore, a task is equivalently modelled to be generated at the end of a time slot with probability λ_k^*, and no task is generated with probability $1-\lambda_k^*$. From [39], the following holds
$$\lambda_k = \frac{\lambda_k^*}{\tau}. \qquad (1)$$
Since $0 \leq \lambda_k^* \leq 1$, it needs to hold that $\tau \leq \frac{1}{\lambda_k}$ $\forall k$.
- Regarding GF transmissions, a task departs from the queue at the beginning of a time slot, with the assumption that only one task departs at a slot. Note, that a

late arrival model is adopted, and so, a task arrives just before the departure of a task.

- During partial offloading, a task's size is not altered, but the task is either computed locally or it is transmitted to the MEC server as a whole. The partial factor θ_k expresses the percentage of tasks that are transmitted to the MEC server. Therefore, $\theta_k\lambda_k$ tasks per second enter the transmission buffer and $(1-\theta_k)\lambda_k$ tasks per second enter the CPU buffer. Since $0 \leq \theta_k \leq 1$, the partial offloading is a probabilistic strategy, where with θ_k probability a task is sent to the MEC server, otherwise it is locally processed. It is noted that the proposed offloading offers lower implementation complexity compared to the conventional partial offloading, since separating a task into subtasks depends on the task's structure, content and functions.

The average throughput (bits/sec) of the proposed GF transmission, under Rayleigh fading conditions, with normalized bandwidth, and for one available preamble, is given by [40] as

$$\bar{R}_k = R_k \exp\left(-\frac{(2^{R_k}-1)N_0}{P_k\Omega_k}\right) q_k \prod_{i\neq k}(1-q_i), \tag{2}$$

where R_k is the fixed data rate (bits/sec) of each device and $\Omega_k = \mathbb{E}[|h|^2]$ with h denoting the channel fading coefficient that follows a Rayleigh distribution. Effectively, this term is related to the average received power, so it can be utilized to include the path loss. N_0 is the power spectral density of noise. The second term, from the left, of (2) expresses the probability of non-outage probability during the data phase, assuming a Rayleigh channel. The last term, from the left, of (2) is the probability of no-collision during the preamble phase, for the case of one preamble. The probability of no-collision, for L_p available preambles, can easily be found, [14], as

$$q_k \prod_{i\neq k}\left(1-\frac{q_i}{L_p}\right). \tag{3}$$

Also, the power consumption due to local computation is given as [8],

$$P_{\text{cmp}} = k_k f_k^3, \tag{4}$$

where k_k is a constant related to the hardware architecture.

III. DELAY ANALYSIS

A. DELAY ANALYSIS OF THE GFRA BUFFER

We consider a GFRA scheme where each user has an infinite buffer capacity and a discrete-time queueing model. The analysis of the queue can be carried out similarly to [41]. The total probability flow through any closed boundary must be zero, so for the k-th user according to Figure 3, the following needs to hold,

$$\begin{aligned} &\lambda'_k g_{0,k} = \mu_k(1-\lambda'_k)g_{1,k},\\ &\lambda'_k(1-\mu_k)g_{i,k} = \mu_k(1-\lambda'_k)g_{i+1,k}, \quad i \in Z\\ &\lambda'_k(1-\mu_k)g_{K-1,k} = \mu_k g_{K,k}, \end{aligned} \tag{5}$$

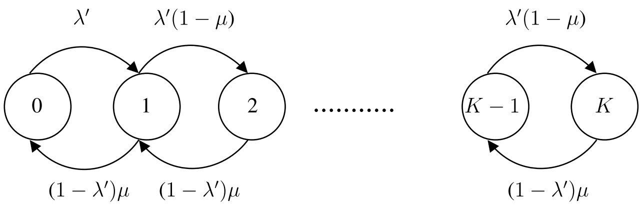


FIGURE 3. Probability flow chart of the transmission buffer.

where $g_{i,k}$ denotes the probability that the buffer of the k-th user contains i tasks. The probability that a task will arrive to the transmission queue, following (1), is given as

$$\lambda'_k = \theta_k\lambda_k\tau. \tag{6}$$

Also, μ_k is the probability of successful transmission of the k-th user and for a GFRA scheme with imperfect channel conditions, is given by (2) as,

$$\mu_k = \exp\left(-\frac{(2^{R_k}-1)N_0}{P_k\Omega_k}\right) q_k \prod_{i\neq k}\left(1-\frac{q_i}{L_p}\right). \tag{7}$$

From (5), $g_{i,k}$, $i \in Z$, can be calculated with respect to $g_{0,k}$ as,

$$\begin{aligned} g_{i,k} &= \left(\frac{\lambda'_k(1-\mu_k)}{\mu_k(1-\lambda'_k)}\right)^{i-1} \frac{\lambda'_k}{\mu_k(1-\lambda'_k)} g_{0,k}, \quad i \in Z\\ g_{K,k} &= \frac{\lambda'_k}{\mu_k}\left(\frac{\lambda'_k(1-\mu_k)}{\mu_k(1-\lambda'_k)}\right)^{K-1} g_{0,k} \end{aligned} \tag{8}$$

Substituting (8) into the total probability law, i.e.,

$$\sum_{i=0}^{K} g_{i,k} = 1$$

results in

$$\begin{aligned} g_{0,k} &= \left(\frac{\mu_k}{\mu_k-\lambda'_k} - \frac{\lambda'^2_k}{\mu_k(\mu_k-\lambda'_k)}\frac{\lambda'_k}{\mu_k}\left(\frac{\lambda'_k(1-\mu_k)}{\mu_k(1-\lambda'_k)}\right)^{K-1}\right)^{-1}\\ &= 1-\rho_k, \quad \text{for} \quad K \longrightarrow \infty, \end{aligned} \tag{9}$$

where $\rho_k = \lambda'_k/\mu_k$. Now that $g_{0,k}$ is known, every $g_{i,k}$ can be calculated using (5). Furthermore, the average queue length of the k-th user can be found as [41],

$$l_k = \sum_{i=0}^{K} i g_{i,k}$$

and for an infinite buffer, i.e., $K \longrightarrow \infty$, by [41] we have

$$l_k = \frac{\rho_k(1-\lambda'_k)}{1-\rho_k}.$$

Hence, using Little's law [41] the average response time, in seconds, is given as,

$$d_{\text{q},k} = \frac{l_k}{\lambda'_k}\tau = \frac{(1-\lambda'_k)}{\mu_k(1-\rho_k)}\tau = \frac{(1-\lambda'_k)}{(\mu_k-\lambda'_k)}\tau. \tag{10}$$

Note that the average response time is defined as the duration from the moment a packet enters the queue until its successful departure.

B. DELAY ANALYSIS OF THE LOCAL COMPUTATION BUFFER

The local computation buffer is also assumed infinite and there is no need for retransmission mechanisms at the output of the CPU buffer. The CPU buffer acts in a deterministic way, therefore it can be modelled as a Geo/D/1 queue [39]. Its average departure rate is then given as

$$\bar{\mu}_k = \frac{f_k}{X_k L_k}\tau \quad (tasks/slot). \tag{11}$$

The average input rate is given as $\bar{\lambda}_k = (1-\theta_k)\lambda_k\tau$. The Geo/D/1 queueing delay is known and given from [39], in seconds, as,

$$d_{\text{cq},k} = \frac{\bar{\rho}_k}{2(1-\bar{\rho}_k)}\left(\frac{1}{\bar{\mu}_k}-1\right)\tau, \tag{12}$$

where $\bar{\rho}_k = \frac{\bar{\lambda}_k}{\bar{\mu}_k}$. Note that the queueing delay measures the delay spent until a task reaches the end of the queue and begins to be served by the CPU.

IV. STABILITY ANALYSIS

A queueing system is said to be unstable if the queue size goes to infinity with non-zero probability, so it is important to examine the stability for queueing systems with infinite buffer capacity. In this section, we study the stability of the proposed buffer architecture with infinite buffer capacity. From [41] an infinite buffer system is global stable if and only if its mean input data rate is equal or less than its mean output data rate. For the buffer of the k-th device, which is dedicated to full offloading stability is provided when it holds that

$$\mu_k \geq \lambda'_k. \tag{13}$$

Moreover, when a packet successfully leaves the queue, all of its bits are pushed to the MEC server, which does not happen instantaneously. Its time duration has to be less than the duration of the data transmission phase of one time slot. Otherwise, the next packet in queue of any user may suffer an unnecessary collision in the next time slot, since by assumption all packets are transmitted to the beginning of the data transmission phase. Thus, the following has to hold

$$\frac{L_k}{R_k} \leq \tau. \tag{14}$$

Note that with (14), an adaptive rate is introduced to the GFRA transmission, which is contradictory to the existing literature where it is assumed that a packet is transmitted instantly. On the other hand, for the buffer dedicated to local computing to be global stable, the following is required to hold

$$\bar{\mu}_k \geq \bar{\lambda}_k. \tag{15}$$

It should be noted that a system which can perform partial offloading is more flexible and stable than a system which performs binary offloading, since utilizing partial offloading means $\theta \leq 1$, therefore each buffer experiences less congestion in comparison to the case of full offloading or local computing. Moreover, the stability constraints have to be taken into account for any optimization problem, otherwise the optimal solution of the problem might lead to an unstable solution and huge queueing delays.

V. COMPUTATION MODEL OF PHYSICAL AND DT COUNTERPARTS

The computation latency until a task of L_k bits is processed when a physical device's CPU operates with frequency of f_k is known and given as

$$d_{\text{cmp},k} = \frac{L_k X_k}{f_k} \tag{16}$$

The DT of the k-th device is expressed as $\text{DT}_k = \{(f_k, \tilde{f}_k)\}$, where $\tilde{f}_k = f_k + n_k$ is the estimated frequency at the DT and n_k is the frequency deviation between the virtual representation and its k-th physical counterpart. Assuming that the deviation of the CPU processing frequency between the physical devices and their DT can be acquired in advance [37], the computing latency gap between the real value and the DT estimation can be calculated as [37]

$$d_{\text{gap},k} = -L_k X_k \frac{n_k}{f_k(f_k+n_k)}, \tag{17}$$

therefore the total computation latency estimated at the DT is given as

$$\tilde{d}_{\text{cmp},k} = d_{\text{cmp},k} + d_{\text{gap}} = \frac{L_k X_k}{f_k + n_k} \tag{18}$$

It is also assumed, that the DT has perfect knowledge of the MEC's condition, which can be justified by the superiority of the wired backhaul channels. By following a similar approach, the queueing delay at the devices' buffers is estimated at the DT as

$$\tilde{d}_{\text{cq},k} = \frac{\tilde{\rho}_k}{2(1-\tilde{\rho}_k)}\left(\frac{1}{\tilde{\mu}_k}-1\right)\tau, \tag{19}$$

where $\tilde{\mu}_k = \frac{f_k+n_k}{X_k L_k}\tau$ and $\tilde{\rho}_k = \frac{\bar{\lambda}_k}{\tilde{\mu}_k}$.

VI. DELAY MINIMIZATION

A. PROBLEM FORMULATION

We aim to minimize the average delay of every device while taking into account their power and stability requirements. Therefore, the access probability q_k, the offloading factor θ_k and the data transmission duration will be optimized according to the total number of devices in the system, the average channel statistics, and the average task generation rate of each device. We note that the proposed problem also provides a lower bound of the delay when the activation probability q_k is fixed or unknown. Furthermore, the formulated analysis can also be adjusted for a power consumption minimization problem subject to delay constraints.

The average delay between the k-th user and the MEC server is given as

$$d_{\mathrm{o},k} = d_{\mathrm{q},k} + d_{\mathrm{t},k} + d_{\mathrm{mec},k}, \tag{20}$$

where $d_{\mathrm{t},k}$ expresses the delay caused when a task successfully leaves the queue until all of its bits are pushed to the MEC server and is given as,

$$d_{\mathrm{t},k} = \frac{L_k}{R_k}. \tag{21}$$

The computation time of the MEC server and the delay caused during the downlink communication between the MEC and its clients is denoted as $d_{\mathrm{mec},k}$ and it is omitted since the MEC server has superior capabilities compared to the devices it serves. Similarly, the average local computation delay of every device is given as,

$$d_{\mathrm{l},k} = \tilde{d}_{\mathrm{cmp},k} + \tilde{d}_{\mathrm{cq},k}, \tag{22}$$

Thus, the overall average delay until a task of the k-th device is completed, either at the MEC server or locally, is given as,

$$d_{\mathrm{p},k} = (1-\theta_k)d_{\mathrm{l},k} + \theta_k d_{\mathrm{o},k}. \tag{23}$$

Therefore, the proposed delay minimization problem with power constraints can be formulated as follows,

$$\begin{aligned} \min_{\mathbf{R},\mathbf{q},\tilde{\mathbf{f}},\boldsymbol{\tau},\boldsymbol{\theta},\boldsymbol{P}} \quad & \sum_{k=1}^{N}\Big[(1-\theta_k)d_{\mathrm{l,k}} + \theta_k d_{\mathrm{k}}\Big] \\ \text{s.t} \quad & \mathrm{C}_1 : P_k + k_k\tilde{f}_k^3 \leq P_{\max,k}, \quad \forall k \in \mathrm{N} \\ & \mathrm{C}_2 : \frac{L_k}{R_k} \leq \tau \\ & \mathrm{C}_3 : \mu_k \geq \lambda'_k \\ & \mathrm{C}_4 : \bar{\mu}_k \geq \bar{\lambda}_k \end{aligned} \tag{24}$$

where $\tilde{f}_k = f_k + n_k$. Constraint C_1 limits the transmission power during the uplink communication using GFRA and the power consumed during the local computing, so that every device is power efficient. Constraints C_2-C_4 ensure that the optimal solution of the proposed problem is also stable.

B. CONVEX TRANSFORMATION

The problem is non-convex due to its non-convex objective function and constraints $\mathrm{C}_2 - \mathrm{C}_4$ containing the product of various optimization variables. In order to formulate it as a convex problem we first transform it into its epigraph form as follows,

$$\begin{aligned} \min_{\mathbf{R},\mathbf{q},\tilde{\mathbf{f}},\boldsymbol{\tau},\boldsymbol{\tau_k},\boldsymbol{\theta},\boldsymbol{P}} \quad & \sum_{k=1}^{N}\tau_k \\ \text{s.t} \quad & \mathrm{C}_1 : P_k + k_k\tilde{f}_k^3 \leq P_{\max,k}, \quad \forall k \in \mathrm{N} \\ & \mathrm{C}_2 : \frac{L_k}{R_k} \leq \tau \\ & \mathrm{C}_3 : \mu_k \geq \lambda'_k \\ & \mathrm{C}_4 : \bar{\mu}_k \geq \bar{\lambda}_k \\ & \mathrm{C}_5 : (1-\theta_k)d_{\mathrm{l},k} \leq \tau_k \\ & \mathrm{C}_6 : \theta_k d_{\mathrm{o},k} \leq \tau_k. \end{aligned} \tag{25}$$

The problem is still non-convex. By substituting the relations for $\bar{\mu}_k$, λ'_k, $\bar{\lambda}_k$, $d_{\mathrm{o},k}$ and $d_{\mathrm{l,k}}$, problem (25) is equivalently written as,

$$\begin{aligned} \min_{\mathbf{R},\mathbf{q},\tilde{\mathbf{f}},\boldsymbol{\tau},\boldsymbol{\tau_k},\boldsymbol{\theta},\boldsymbol{P}} \quad & \sum_{k=1}^{N}\tau_k \\ \text{s.t} \quad & \mathrm{C}_1 : P_k + k_k\tilde{f}_k^3 \leq P_{\max,k}, \quad \forall k \in \mathrm{N} \\ & \mathrm{C}_2 : \frac{L_k}{R_k} \leq \tau \\ & \mathrm{C}_3 : \mu_k \geq \lambda_k\theta_k\tau \\ & \mathrm{C}_4 : \frac{\tilde{f}_k}{X_kL_k} \geq (1-\theta_k)\lambda_k \\ & \mathrm{C}_5 : (1-\theta_k)\frac{X_kL_k}{\tilde{f}_k} + \\ & \frac{(1-\theta_k)^2\lambda_k}{2\Big(\frac{\tilde{f}_k}{X_kL_k} - (1-\theta_k)\lambda_k\Big)}\left(\frac{X_kL_k}{\tilde{f}_k} - \tau\right) \leq \tau_k \\ & \mathrm{C}_6 : \theta_k\frac{L_k}{R_k} + \theta_k\frac{(1-\lambda_k\theta_k\tau)}{(\mu_k - \lambda_k\theta_k\tau)}\tau \leq \tau_k \end{aligned} \tag{26}$$

Problem (26) is non-convex, due to μ_k containing the product of several optimization variables, as well because of the constraints C_2-C_6. To formulate the problem as convex we will introduce the following auxiliary variable

$$\tilde{\mu}_k \geq \mu_k. \tag{27}$$

One way to deal with the product of optimization variables in C_2-C_6 is to take the logarithm of both sides of those constraints. However, due to the summation in the left side of constraints C_5 and C_6, those constraints will be difficult to handle. To that end, the variables $\tau_k^{(i)}$, $i \in \{1, 2, 3, 4\}$ will be introduced, for which it holds that

$$\tau_k^{(1)} + \tau_k^{(2)} \leq \tau_k \quad \text{and} \quad \tau_k^{(3)} + \tau_k^{(4)} \leq \tau_k. \tag{28}$$

Utilizing the above formulations, problem (26) is written as,

$$\begin{aligned} \min_{\mathbf{R},\mathbf{q},\tilde{\mathbf{f}},\boldsymbol{\tau},\boldsymbol{\tau_k},\boldsymbol{\theta},\tilde{\boldsymbol{\mu}},\boldsymbol{P}} \quad & \sum_{k=1}^{N}\tau_k \\ \text{s.t} \quad & \mathrm{C}_1 : P_k + k_k\tilde{f}_k^3 \leq P_{\max,k}, \quad \forall k \in \mathrm{N} \\ & \mathrm{C}_2 : \frac{L_k}{R_k} \leq \tau \\ & \mathrm{C}_3 : \tilde{\mu}_k \geq \lambda_k\theta_k\tau \\ & \mathrm{C}_4 : \frac{\tilde{f}_k}{X_kL_k} \geq (1-\theta_k)\lambda_k \\ & \mathrm{C}_5 : (1-\theta_k)\frac{X_kL_k}{\tilde{f}_k} \leq \tau_k^{(1)} \\ & \mathrm{C}_6 : \frac{(1-\theta_k)^2\lambda_k}{2\Big(\frac{\tilde{f}_k}{X_kL_k} - (1-\theta_k)\lambda_k\Big)}\left(\frac{X_kL_k}{\tilde{f}_k} - \tau\right) \leq \tau_k^{(2)} \\ & \mathrm{C}_7 : \theta_k\frac{L_k}{R_k} \leq \tau_k^{(3)} \\ & \mathrm{C}_8 : \theta_k\frac{(1-\lambda_k\theta_k\tau)}{(\tilde{\mu}_k - \lambda_k\theta_k\tau)}\tau \leq \tau_k^{(4)} \end{aligned}$$

$$\begin{aligned}
&\mathrm{C}_9: \tau_k^{(1)} + \tau_k^{(2)} = \tau_k \\
&\mathrm{C}_{10}: \tau_k^{(3)} + \tau_k^{(4)} = \tau_k \\
&\mathrm{C}_{11}: \exp\left(-\frac{(2^{R_k}-1)N_0}{P_k\Omega_k}\right) q_k \prod_{i\neq k}\left(1-\frac{q_i}{L_p}\right) \geq \tilde{\mu}_k.
\end{aligned} \tag{29}$$

Problem (29) is still non-convex. From constraint C_2 and relations (1), (12) it is concluded that the time slot duration τ is lower bounded by the value of $\max\{\frac{L_k}{R_k}\}$ and it is upper bounded by the lowest value of $\min\{\frac{1}{\lambda_k}\}$ and $\min\{\frac{X_k L_k}{\tilde{f}_k}\}$. Therefore the following has to hold,

$$\max_k\left\{\frac{L_k}{R_k}\right\} \leq \min_k\left\{\frac{1}{\lambda_k}, \frac{X_k L_k}{\tilde{f}_k}\right\}. \tag{30}$$

Otherwise, the problem is either infeasible, due to (1) and (12), or collisions occur between two packets sent to different time slots due to C_2. To make the problem simpler to solve, the concept of AO will be utilized, by fixing the value of τ when optimizing the rest of the variables. The optimal value of τ is given as

$$\tau^* = \max_k\left\{\frac{L_k}{R_k^*}\right\}. \tag{31}$$

To deal with the non-convex constraints, the logarithm of both sides of constraints C_5-C_8 and C_{11} will be taken. Consequently, the problem is formulated as follows,

$$\begin{aligned}
\min_{\mathbf{R},\mathbf{q},\tilde{\mathbf{f}},\tau_k,\boldsymbol{\theta},\boldsymbol{P}} \quad & \sum_{k=1}^{N}\tau_k \\
\text{s.t} \quad & (29)\ \mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3, \mathrm{C}_4, \mathrm{C}_9, \mathrm{C}_{10} \\
& \mathrm{C}_5: \log\left(\frac{\tilde{f}_k}{X_k L_k}\right) + \log\left(\tau_k^{(1)}\right) + \log(1-\theta_{k,0}) + \frac{(\theta_k-\theta_{k,0})}{\theta_{k,0}} \leq 0 \\
& \mathrm{C}_6: \log\left(\frac{X_k L_k}{\tilde{f}_{k,0}} - \tau\right) - \frac{\tilde{f}_k - \tilde{f}_{k,0}}{\tilde{f}_{k,0}X_k L_k - \tau\tilde{f}_{k,0}} \\
& + 2\log(1-\theta_{k,0}) - 2\frac{\theta_k-\theta_{k,0}}{1-\theta_{k,0}} \\
& - \log\left(\frac{\tilde{f}_k}{X_k L_k} - (1-\theta_k)\lambda_k\right) + \log\left(\frac{\lambda_k}{2}\right) - \log\left(\tau_k^{(2)}\right) \leq 0 \\
& \mathrm{C}_7: -\log\left(\frac{R_k}{L_k}\right) - \log\left(\tau_k^{(3)}\right) + \log(\theta_{k,0}) + \frac{\theta_k-\theta_{k,0}}{\theta_{k,0}} \leq 0 \\
& \mathrm{C}_8: -\log(\tilde{\mu}_k - \lambda_k\theta_k\tau) + \log(\tau) - \log\left(\tau_k^{(4)}\right)\log(\theta_{k,0}) \\
& + \frac{\theta_k-\theta_{k,0}}{\theta_{k,0}} + \log(1-\lambda_k\theta_{k,0}\tau) - \lambda_k\tau\frac{\theta_k-\theta_{k,0}}{1-\lambda_k\tau\theta_{k,0}} \leq 0 \\
& \mathrm{C}_{11}: \frac{(2^{R_k}-1)N_0}{P_k\Omega_k} - \log(q_k) - \sum_{i\neq k}\log\left(1-\frac{q_i}{L_p}\right) \\
& - \log(\tilde{\mu}_{k,0}) - \frac{\tilde{\mu}_k-\tilde{\mu}_{k,0}}{\tilde{\mu}_{k,0}} \leq 0
\end{aligned} \tag{32}$$

Algorithm 1 Solution to (32)

```
1: Choose the maximum number of iterations $n_{\max}$, an initial point $x_0$, initial points $\theta_{k,0}$, $r_{k,0}$ for the SCA prodecure and tolerance $\epsilon$
2: while ($n \leq n_{\max}$ and $||x^k - x^{k-1}||_\infty \geq \epsilon$) do
3:     solve problem (32) and obtain optimal $x^*$
4:     $x_0 \leftarrow x^*$, $\theta_{k,0} \leftarrow \theta^*$, $\tilde{\mu}_{k,0} \leftarrow \tilde{\mu}_k^*$, $\tilde{f}_{k,0} \leftarrow f^*$
5:     if $\max\{\frac{L_k}{R_k^*}\} \leq \min\{\frac{1}{\lambda_k}, \frac{X_k L_k}{\tilde{f}_k^*}\}$},   $\forall k$ then
6:         $\tau^* = \max\{\frac{L_k}{R_k^*}\}$
7:     else
8:         Problem is infeasible
9:     end if
10: end while
```

where in order to cope with the non-convex negative logarithmic terms of constraints C_2, C_3, $\mathrm{C}_6 - \mathrm{C}_8$ and C_{11}, that occurred due to taking the logarithm of both sides, SCA was exploited. Specifically, each negative logarithmic term was approximated by its first order Taylor approximation. In [42] three conditions are mentioned that are required to hold when approximating a non-convex constraint. Assume that $\gamma(x)$ is the non-convex term and $\tilde{\gamma}(x, x_k)$ is its convex approximation. Then the following need to hold,

$$(i) \quad \gamma(x) \leq \tilde{\gamma}(x, x_k) \tag{33}$$

$$(ii) \quad \gamma(x_k) = \tilde{\gamma}(x_k, x_k) \tag{34}$$

$$(iii) \quad \frac{\partial\gamma(x_k)}{\partial x} = \frac{\partial\tilde{\gamma}(x_k, x_k)}{\partial x}. \tag{35}$$

It can be easily verified that for the log(·) function and its Taylor approximation all three conditions hold. Eventually, problem (32) is convex. Furthermore, it should be noted that the formulated average delay minimization problem can be easily written as a power consumption minimization problem subject to average delay constraints. Specifically, by fixing the values of τ_k in constraints C_9 and C_{10} and by substituting the objective of the problem with constraint C_1, the formulation describes a power minimization problem subject to average delay constraints.

C. PROPOSED ALGORITHM

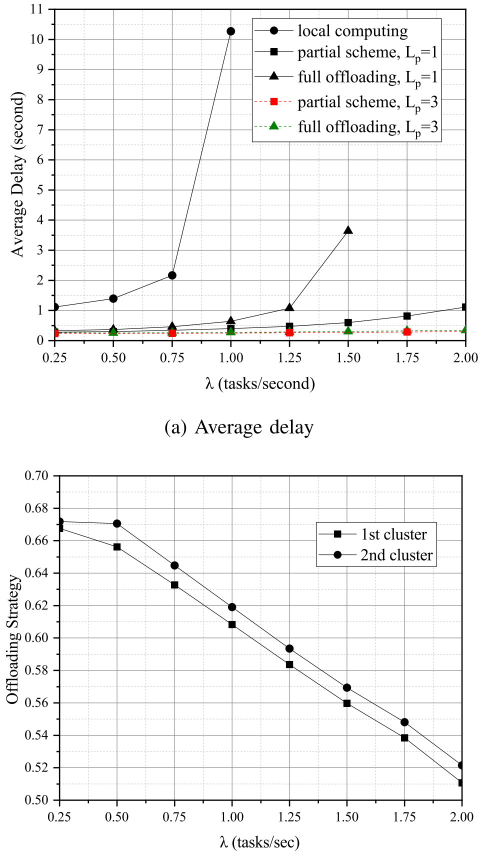
The problem can be solved by any general purpose convex optimization method, following Algorithm 1. A common approach to solve non-linear constrained convex problems is the interior-point method with complexity of roughly $\mathrm{O}(\mathrm{N}^3)$ [43], where N is the number of variables. For convex problems, the interior-point method converges to a global optimal point with great accuracy. Moreover, in line 4 of Algorithm 1 the initial point of the SCA procedure is updated by the optimal solution obtained from the interior-point method. Therefore, at each iteration, the Taylor approximation is more accurate, since the approximation is closer to the optimal point of (32). Note that the complexity of Algorithm 1 in conjunction with the interior-point method is $\mathrm{O}(n_{max}\mathrm{N}^3)$, where n_{max} is the maximum number of iterations allowed.

VII. NUMERICAL RESULTS

In this section, simulation results are presented for a GFRA MEC network. Unless otherwise stated, the simulation parameters are given in Table 2. To highlight the effect of average received power in the proposed method, the total number of users N is separated into two clusters, that

TABLE 2. Simulation parameters.

Parameter	Value
N	6
N_0	-174dBm/Hz
L_k	1Mbit
L_p	1
λ_k	0.5 (tasks/sec)
X_k	500 CPU cycles/bit
$f_{max,k}$	1GHz
$P_{max,k}$	0.125W
B_k	1MHz
k_k	10^{-27}
ϵ	10^{-4}
n_{max}	20
d_1, d_2	50m, 100m
α	2.5
n_k	$0\%, 1\%, 2\%, 5\%$



(a) Average delay

(b) Offloading strategy

FIGURE 4. Resource allocation vs the task generation rate.

are denoted by $i \in \{1, 2\}$. The users of the same cluster are considered to have equal average received power Ω_i. Without loss of generality, the path loss model is given as $\Omega_i = \frac{1}{(1+d_i)^\alpha}$. In order to extract insights about the network's performance, the optimal resource allocation strategy presented in this section has been evaluated by using Algorithm 1.

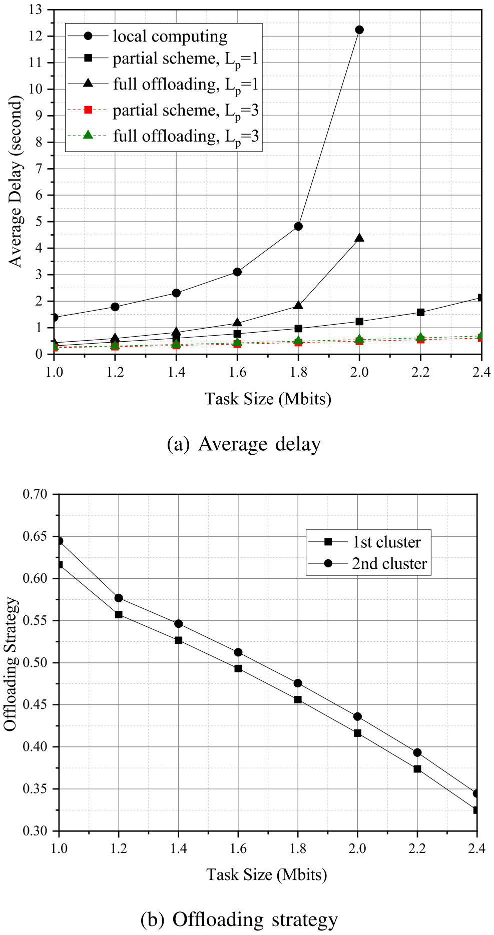
In Figure 4, the offloading strategy and the average delay are plotted against the average task generation of the devices. In Figure 4a, we observe that for the parameters chosen the local computing delay is by far worse than both the full offloading and the proposed partial scheme. For low average task generation rates, the proposed partial scheme is slightly more efficient than the full offloading. However, as the task generation rate increases the performance gap between the proposed scheme and the full offloading increases as well. Both local computing and full offloading experience congestion that results in greater delays compared to the proposed strategy. It is interesting to note that the delay of the proposed scheme can be 5 times less than the full offloading delay and 20 times less than the local computing delay. The partial scheme will eventually also experience congestion, but for greater values of task generations rate, thus it is more delay efficient.

The delay efficiency of the proposed offloading scheme is attributed to the fact that it adjusts the percentage of tasks transmitted to the MEC server or the percentage of tasks that are computed locally. In Figure 4b the offloading strategy is shown for the same values of λ. Due to the fact that full offloading causes smaller delays in comparison to local computing, for low task generation rates the majority of the tasks are sent to the MEC server. When the task generation rate increases, full offloading, due to collisions or outage, cannot provide the required queue stability, therefore a percentage of the tasks are eventually computed locally. That causes θ to decrease, thus aiding the partial scheme to retain stability at its buffers, as can be verified from constraints $C_3 - C_4$ of (24).

In Figure 5, the impact of the tasks' size is investigated. In Figure 5a the average delay is illustrated. It is observed that both local computing and full offloading experience congestion for smaller task sizes in comparison to the proposed scheme, which results in increased delays. The delay of the partial scheme can be seen to be 4 times lower than full offloading and 10 times lower than local computing, while both binary strategies experience congestion. The proposed scheme offers increased flexibility, since from Figure 5a and Figure 4a it is concluded that the partial scheme supports both greater task generation rates and large task sizes.

In Figure 5b, the offloading strategy is presented. Note that the task size L does not affect the partial strategy θ directly, as it can be verified from the theoretic analysis. Nonetheless, the task size has a great impact on the time slot duration, since for collisions to be avoided between successive time slots, a task has to be transmitted to the MEC server in a duration less than a time slot. As such, Figure 5b shows that the task size has a great impact on the offloading strategy, since θ rapidly diminishes for greater values of L. This attributed to the fact that transmitting bigger tasks requires greater data rates R. However, greater data rates cause the outage probability to increase, as can be seen from (2), which undermines the stability of the transmission buffer.

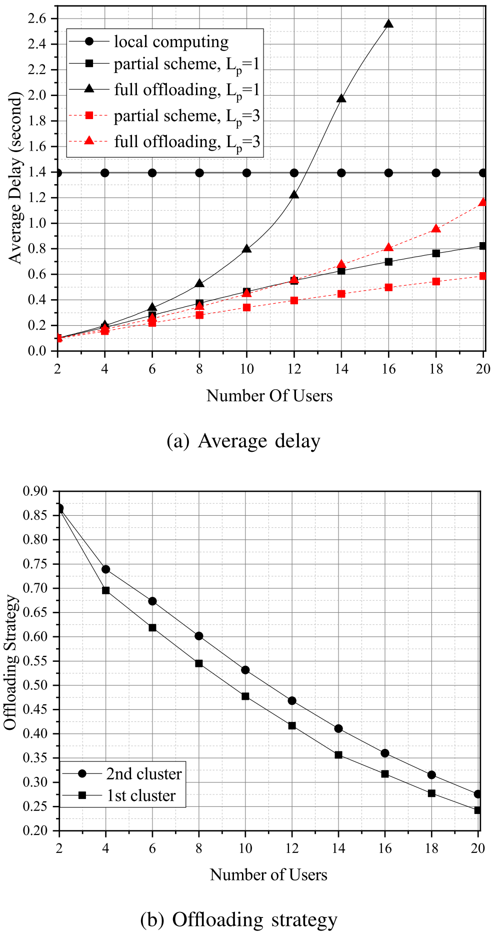
In Figure 6, the number of devices in the network and its impact on the optimal offloading strategy are studied. As expected, local computing delay is constant for all number of users. On the other hand, full offloading experiences congestion for a relatively small number of users, which causes



(a) Average delay

(b) Offloading strategy

FIGURE 5. Resource allocation vs the task size.



(a) Average delay

(b) Offloading strategy

FIGURE 6. Resource allocation vs the total number of users.

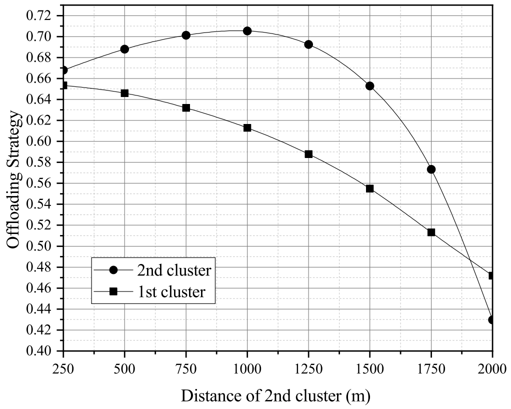
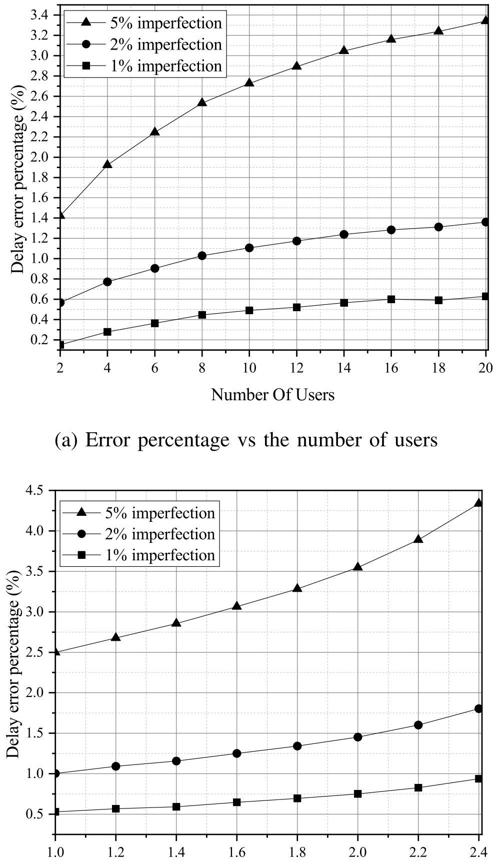


FIGURE 7. The offloading strategy vs the distance of 2nd cluster.

full offloading to be less delay efficient than local computing. This is due to the fact more clients are likely to transmit data to the MEC server, therefore the probability of collisions dramatically increases. The partial scheme is robust, since it can adjust its partial strategy. As a consequence, the average delay slowly increases with conjunction to the number of users and for 20 users, the delay is about half the delay occurred by utilizing local computing. Nonetheless, the delay of the partial scheme gradually approaches the delay of local computing, which can be verified by Figure 6b, where the offloading factor θ is shown to rapidly decrease as the number of users increases.

In Figure 7, the impact of the distance from the MEC server is examined. In this setup we have assumed that the 1st cluster of users lie in 50m distance from the MEC and the distance of the 2nd cluster is altered. It is observed that the 2nd cluster of users choose to offload a greater amount of tasks to the MEC server compared to the 1st cluster of users which lie closer to the MEC. At first glance that may seem contradictory. However, because of the greater path loss of the 2nd cluster, its users have to consume more power when offloading their tasks to the MEC server, compared to the users of the 1st cluster. If we take into account the probabilistic offloading scheme, with which some tasks will be offloaded, and other will be locally processed, it is concluded that less power is available to the users of the second cluster, for local computing, which limits the tasks that are locally processed, and therefore, the offloading factor of the 2nd cluster is greater compared to the 1st cluster. Moreover, we note that the users of the 2nd cluster, due to higher outage probability, access the channel more frequently, causing increased interference to the users of the 1st cluster during the preamble phase.



(a) Error percentage vs the number of users

(b) Error percentage vs L

FIGURE 8. Deviation impact to the DT delay.

Furthermore, the offload factor of the 2nd cluster has a non-monotonic behaviour. In general, offloading the tasks to the MEC server is faster that local computing. As the distance increases the users of the 2nd cluster offload their data to the MEC for efficiency and the offload factor increases. However, after some distance, offloading a task to the server is less efficient than locally processing it, therefore the offload factor rapidly diminishes to the point where the users of the 2nd cluster offload less tasks than the users of the 1st cluster.

Regarding the impact of the available preambles on the networks’ performance, from Figure 3-5 it is evident that as the number of preambles increases, the delay diminishes rapidly, due to the fact that the possible collisions are reduced. For an equal number of preambles L_p, the proposed partial framework is more delay efficient as can be seen from the figures depicting the average delay, for $L_p = 1$. From Figure 6, it is also verified that increasing the number of preambles increases the system’s connectivity, since both full and partial offloading, for $L_p = 3$, experience much less congestion compared to the case of $L_p = 1$.

In Figure 8, the impact of the imperfection between the DT and its underlying physical counterparts is examined. In this setting we have assumed that a constant deviation between the devices’ CPU frequency and their DT counterpart exists, as in Section V. On top of that, due to the unreliability of the wireless channels, the DT is also assumed to have poor knowledge on the hardware configuration of the devices, therefore $\hat{P}_{\max,k} = P_{\max,k} + n_k$ and $\hat{\lambda}_k = \lambda_k + n_k$. The imperfection will be expressed as a percentage of the real value of the parameters [37], so for $n_k = 1\%$, $\hat{\lambda}_k = \lambda_k \pm \frac{1}{100}\lambda_k$, etc. From Figure 8a it is observed that a slight uncertainty between the DT and the physical system causes greater errors to the optimal resource allocation as the number of users increases. Therefore, imperfections tend to be more damaging as the size of the physical space and the DT increase. Moreover, in Figure 8b the percentage of error caused with various task sizes is plotted. The error increases as the task size increases. Consequently, inaccuracies between the DT and the physical network cannot be ignored when the state of the network approaches congestion. These imperfections might cause significant errors under computationally demanding network states, for instance, in cases of excessive traffic packet generation.

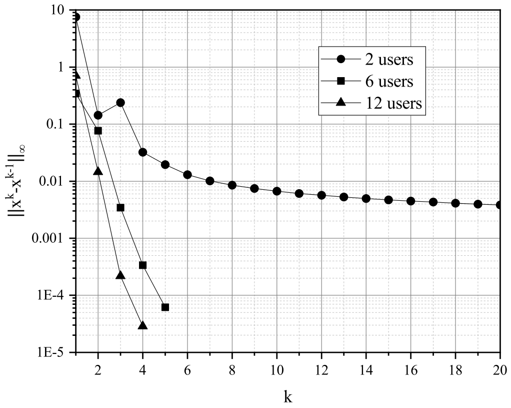


FIGURE 9. Convergence of Algorithm 1.

In Figure 9, the convergence of Algorithm 1 is shown. More users in the network indicate a larger number of constraints and optimization variables. By choosing a random point, which satisfies the stability constraints, and for the case of 2 users, it is shown that the algorithm slowly converges with accuracy of approximately $5 \cdot 10^{-2}$ within 20 iterations. However, for the case of 6 users the algorithm needs 5 iterations to converge with accuracy lower than 10^{-4}. Moreover, for 12 users, only 4 iterations are needed. The fact that the warm start approach is used between the iterations greatly accelerates the proposed algorithm. Therefore, a good strategy for choosing the initial point is to run the algorithm for a small and easy problem, for example for the 2 users case, and then, the optimal point found is utilized as the initial point for other cases.

VIII. CONCLUSION

In this paper, we studied the average delay for a DT-aided MEC system with GF random access by using queueing theory tools. A novel partial offloading scheme was proposed in which a task is probabilistically computed locally or offloaded to the MEC server. The duration of the data transmission phase, an arbitrary number of preambles and the average outage probability were taken into account, while an adaptive data transmission rate was utilized. Then, considering imperfections between the DT and its physical counterpart, closed-form relations were extracted for the average delay of each device and an optimization aiming to minimize the average delay of all users was formulated by utilizing SCA and AO. Finally, simulation results were presented which give insights about the network's resource allocation strategy under different scenarios. The impact of different network's parameters, such as the number of users and the task generation rate were examined and it was shown that the proposed scheme can efficiently adjust its partial strategy to avoid congestion. Possible future extensions of this work could aim to study semi-GF access for MEC or to further investigate the dynamic characteristics of the DT-MEC architecture in the case of stochastic and unknown imperfections between the DT and the physical world.

REFERENCES

[1] D. C. Nguyen et al., "6G Internet of Things: A comprehensive survey," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 359–383, Jan. 2022.

[2] A. Bre̍kine et al., *Building a Roadmap for the Next Generation Internet of Things. Research, Innovation and Implementation 2021–2027 (Scoping Paper)*, M. Brynskov, F. M. Facca, and G. Hrasko, Eds., Horizon Eur. (HEU), Sep. 2019. [Online]. Available: https://www.ngiot.eu/building-a-roadmap-for-thenext-generation-internet-of-things-scoping-paper

[3] N. H. Mahmood, R. Abreu, R. Böhnke, M. Schubert, G. Berardinelli, and T. H. Jacobsen, "Uplink grant-free access solutions for URLLC services in 5G new radio," in *Proc. 16th Int. Symp. Wireless Commun. Syst. (ISWCS)*, 2019, pp. 607–612.

[4] J. Choi, J. Ding, N.-P. Le, and Z. Ding, "Grant-free random access in machine-type communication: Approaches and challenges," *IEEE Wireless Commun.*, vol. 29, no. 1, pp. 151–158, Feb. 2022.

[5] J. Kim, G. Lee, S. Kim, T. Taleb, S. Choi, and S. Bahk, "Two-step random access for 5G system: Latest trends and challenges," *IEEE Netw.*, vol. 35, no. 1, pp. 273–279, Jan./Feb. 2021.

[6] A. Shahraki, M. Abbasi, M. J. Piran, and A. Taherkordi, "A comprehensive survey on 6G networks:Applications, core services, enabling technologies, and future challenges," 2021, *arXiv:2101.12475*.

[7] Z. Zhang et al., "6G wireless networks: Vision, requirements, architecture, and key technologies," *IEEE Veh. Technol. Mag.*, vol. 14, no. 3, pp. 28–41, Sep. 2019.

[8] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[9] Z. Zhou et al., "Secure and latency-aware digital twin assisted resource scheduling for 5G edge computing-empowered distribution grids," *IEEE Trans. Ind. Informat.*, vol. 18, no. 7, pp. 4933–4943, Jul. 2022.

[10] H. Zhou, Y. Deng, L. Feltrin, and A. Höglund, "Analyzing novel grant-based and grant-free access schemes for small data transmission," *IEEE Trans. Commun.*, vol. 70, no. 4, pp. 2805–2819, Apr. 2022, doi: 10.1109/TCOMM.2022.3150787.

[11] Z. Ding, R. Schober, P. Fan, and H. V. Poor, "Simple semi-grant-free transmission strategies assisted by non-orthogonal multiple access," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4464–4478, Jun. 2019.

[12] Y. Liu, Y. Deng, M. Elkashlan, A. Nallanathan, and G. K. Karagiannidis, "Optimization of grant-free NOMA with multiple configured-grants for mURLLC," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 4, pp. 1222–1236, Apr. 2022.

[13] J. Liu, G. Wu, X. Zhang, S. Fang, and S. Li, "Modeling, analysis, and optimization of grant-free NOMA in massive MTC via stochastic geometry," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4389–4402, Mar. 2021.

[14] J. Ding, D. Qu, H. Jiang, and T. Jiang, "Success probability of grant-free random access with massive MIMO," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 506–516, Feb. 2019.

[15] M. Ke, Z. Gao, Y. Wu, X. Gao, and K.-K. Wong, "Massive access in cell-free massive MIMO-based Internet of Things: Cloud computing and edge computing paradigms," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 3, pp. 756–772, Mar. 2021.

[16] S. Bebortta, D. Senapati, C. R. Panigrahi, and B. Pati, "Adaptive performance modeling framework for QoS-aware offloading in MEC-based IIoT systems," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 10162–10171, Jun. 2022.

[17] S. Guan and A. Boukerche, "A MEC-based distributed offloading model for ubiquitous and time-constraint offloading," in *Proc. IEEE/ACM 23rd Int. Symp. Distrib. Simul. Real Time Appl. (DS-RT)*, 2019, pp. 1–8.

[18] S. Guo, D. Wu, H. Zhang, and D. Yuan, "Queueing network model and average delay analysis for mobile edge computing," in *Proc. Int. Conf. Comput. Netw. Commun. (ICNC)*, 2018, pp. 172–176.

[19] L. Chen, S. Zhou, and J. Xu, "Computation peer offloading for energy-constrained mobile edge computing in small-cell networks," *IEEE/ACM Trans. Netw.*, vol. 26, no. 4, pp. 1619–1632, Aug. 2018.

[20] X. Meng, W. Wang, Y. Wang, V. K. N. Lau, and Z. Zhang, "Closed-form delay-optimal computation offloading in mobile edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4653–4667, Oct. 2019.

[21] J. Zhou, D. Tian, Z. Sheng, X. Duan, and X. Shen, "Distributed task offloading optimization with queueing dynamics in multiagent mobile-edge computing networks," *IEEE Internet Things J.*, vol. 8, no. 15, pp. 12311–12328, Aug. 2021.

[22] G. Zhang, W. Zhang, Y. Cao, D. Li, and L. Wang, "Energy-delay tradeoff for dynamic offloading in mobile-edge computing system with energy harvesting devices," *IEEE Trans. Ind. Informat.*, vol. 14, no. 10, pp. 4642–4655, Oct. 2018.

[23] P. D. Diamantoulakis, P. S. Bouzinis, P. G. Sarigiannidis, Z. Ding, and G. K. Karagiannidis, "Optimal design and orchestration of mobile edge computing with energy awareness," *IEEE Trans. Sustain. Comput.*, vol. 7, no. 2, pp. 456–470, Apr.–Jun. 2022.

[24] L. Yang, H. Zhang, M. Li, J. Guo, and H. Ji, "Mobile edge computing empowered energy efficient task offloading in 5G," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6398–6409, Jul. 2018.

[25] J. Hajipour, "Stochastic buffer-aided relay-assisted MEC," *IEEE Commun. Lett.*, vol. 24, no. 4, pp. 931–934, Apr. 2020.

[26] J. Cao, W. Feng, N. Ge, and J. Lu, "Delay characterization of mobile-edge computing for 6G time-sensitive services," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3758–3773, Mar. 2021.

[27] L. Liu, Z. Chang, X. Guo, S. Mao, and T. Ristaniemi, "Multiobjective optimization for computation offloading in fog computing," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 283–294, Feb. 2018.

[28] C. She, Y. Duan, G. Zhao, T. Q. S. Quek, Y. Li, and B. Vucetic, "Cross-layer design for mission-critical IoT in mobile edge computing systems," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9360–9374, Dec. 2019.

[29] C.-F. Liu, M. Bennis, M. Debbah, and H. V. Poor, "Dynamic task offloading and resource allocation for ultra-reliable low-latency edge computing," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4132–4150, Jun. 2019.

[30] X. Chen and G. Liu, "Joint optimization of task offloading and resource allocation via deep reinforcement learning for augmented reality in mobile edge network," in *Proc. IEEE Int. Conf. Edge Comput. (EDGE)*, 2020, pp. 76–82.

[31] M. Hao, D. Ye, S. Wang, B. Tan, and R. Yu, "URLLC resource slicing and scheduling in 5G vehicular edge computing," in *Proc. IEEE 93rd Veh. Technol. Conf. (VTC-Spring)*, 2021, pp. 1–5.

[32] J. Liu and Q. Zhang, "Offloading schemes in mobile edge computing for ultra-reliable low latency communications," *IEEE Access*, vol. 6, pp. 12825–12837, 2018.

[33] K. Zhang, J. Cao, and Y. Zhang, "Adaptive digital twin and multiagent deep reinforcement learning for vehicular edge computing and networks," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1405–1413, Feb. 2022.

[34] Y. Dai, K. Zhang, S. Maharjan, and Y. Zhang, "Deep reinforcement learning for stochastic computation offloading in digital twin networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 4968–4977, Jul. 2021.

[35] Y. Lu, S. Maharjan, and Y. Zhang, "Adaptive edge association for wireless digital twin networks in 6G," *IEEE Internet Things J.*, vol. 8, no. 22, pp. 16219–16230, Nov. 2021.

[36] R. Dong, C. She, W. Hardjawana, Y. Li, and B. Vucetic, "Deep learning for hybrid 5G services in mobile edge computing systems: Learn from a digital twin," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4692–4707, Oct. 2019.

[37] W. Sun, H. Zhang, R. Wang, and Y. Zhang, "Reducing offloading latency for digital twin edge networks in 6G," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12240–12251, Oct. 2020.

[38] T. Do-Duy, D. Van Huynh, O. A. Dobre, B. Canberk, and T. Q. Duong, "Digital twin-aided intelligent offloading with edge selection in mobile edge computing," *IEEE Wireless Commun. Lett.*, vol. 11, no. 4, pp. 806–810, Apr. 2022.

[39] A. Gravey, J.-R. Louvion, and P. Boyer, "[On the Geo/D/1 and Geo/D/1/n queues]," *Perform. Eval.*, vol. 11, no. 2, pp. 117–125, 1990.

[40] Z. Hadzi-Velkov, S. Pejoski, N. Zlatanov, and R. Schober, "Proportional fairness in ALOHA networks with RF energy harvesting," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 277–280, Feb. 2019.

[41] T. Wan and A. U. Sheikh, "Performance and stability analysis of buffered slotted ALOHA protocols using tagged user approach," *IEEE Trans. Veh. Technol.*, vol. 49, no. 2, pp. 582–593, Mar. 2000.

[42] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Oper. Res.*, vol. 26, no. 4, pp. 681–683, 1978.

[43] K. M. Anstreicher, "Linear programming in O([n3/ln n]L) operations," *SIAM J. Optim.*, vol. 9, no. 4, pp. 803–812, 1999.

NIKOS A. MITSIOU (Student Member, IEEE) was born in Achinos, Phthiotis, Greece, in 1998. He received the Diploma Degree (5 years) in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 2021, where he is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering. His research interests include optimization theory, machine learning, and game theory and applications in wireless networks. He is a member of the Wireless and Communications and Information Processing Group.

VASILIS K. PAPANIKOLAOU (Graduate Student Member, IEEE) was born in Kavala, Greece, in 1995. He received the Diploma Degree (5 years) in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 2018, where is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering. He was a Visiting Research Assistant with Lancaster University, U.K., and with Khalifa University, Abu Dhabi, UAE. His research interests include optical wireless communications, non-orthogonal multiple access, optimization theory, and game theory. He received the IEEE Student Travel Grant Award for IEEE WCNC 2018. He was an Exemplary Reviewer of the IEEE WIRELESS COMMUNICATIONS LETTERS in 2019 and the IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY in 2021 (top 3% of reviewers). He is a member of the Wireless and Communications and Information Processing Group.

PANAGIOTIS D. DIAMANTOULAKIS (Senior Member, IEEE) received the Diploma (five years) and Ph.D. degrees from the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2012 and 2017, respectively. Since 2017, he has been a Postdoctoral Fellow with the Wireless Communications and Information Processing Group, AUTH, and since 2021, he has been a Visiting Assistant Professor with the Key Laboratory of Information Coding and Transmission, Southwest Jiaotong University, Chengdu, China. His research interests include optimization theory and applications in wireless networks and smart grids, game theory, goal-oriented communications, and optical wireless communications. He is also an Editor of IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY, *Physical Communications* (Elsevier), and *Frontiers in Communications and Networks*. Since 2018, he has been an Editor of IEEE WIRELESS COMMUNICATIONS LETTERS.

TRUNG Q. DUONG (Fellow, IEEE) received the B.Eng. degree in electrical and electronics engineering from Bach Khoa Sai Gon, Vietnam, in 2002, the M.Sc. degree in computer science from Kyung Hee University, South Korea, in 2005, and the Ph.D. degree in telecommunications systems from the Blekinge Institute of Technology, Sweden, in 2012.

In 2013, he joined Queen's University Belfast, U.K., as an Academic Staff, where he is currently the Chair Professor of Telecommunications. He also holds the prestigious Research Chair of Royal Academy of Engineering. His current research interests include quantum communications, wireless communications, signal processing, machine learning, and realtime optimization. He received the Best Paper Award at the IEEE VTC-Spring 2013, IEEE ICC 2014, IEEE GLOBECOM 2016, 2019, 2022 IEEE DSP 2017, and IWCMC 2019. He is the recipient of prestigious Royal Academy of Engineering Research Fellowship from 2015 to 2020 and has won a prestigious Newton Prize in 2017. He has served as an Editor/Guest Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE COMMUNICATIONS LETTERS, IEEE WIRELESS COMMUNICATIONS LETTERS, IEEE WIRELESS COMMUNICATIONS, *IEEE Communications Magazines*, and IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS. He is currently serving as an Executive Editor for IEEE COMMUNICATIONS LETTERS.

GEORGE K. KARAGIANNIDIS (Fellow, IEEE) is currently a Professor with the Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, Greece, and the Head of Wireless Communications and Information Processing Group. He is also a Faculty Fellow with the Cyber Security Systems and Applied AI Research Center, Lebanese American University. His research interests are in the areas of wireless communications systems and networks, signal processing, optical wireless communications, wireless power transfer and applications, and communications and signal processing for biomedical engineering. Recently, he received the Three Prestigious Awards: The 2021 IEEE ComSoc RCC Technical Recognition Award, the 2018 IEEE ComSoc SPCE Technical Recognition Award, and the 2022 Humboldt Research Award from Alexander von Humboldt Foundation. He is one of the highly-cited authors across all areas of Electrical Engineering, recognized from Clarivate Analytics as Web-of-Science Highly-Cited Researcher in the eight consecutive years from 2015 to 2022. He was in the past editor in several IEEE journals and from 2012 to 2015, he was the Editor-in Chief of IEEE COMMUNICATIONS LETTERS. From September 2018 to June 2022, he served as an Associate Editor-in Chief of IEEE OPEN JOURNAL OF COMMUNICATIONS SOCIETY. He is currently in the Steering Committee of IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKS.