IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 7, 2020, accepted March 7, 2020, date of publication March 11, 2020, date of current version March 26, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2980196

Optimal Cooperative Offloading Scheme for Energy Efficient Multi-Access Edge Computation

JOSEPH HENRY ANAJEMBA[1], (Member, IEEE), TANG YUE[1], CELESTINE IWENDI[2], (Senior Member, IEEE), MAMDOUH ALENEZI[3], AND MOHIT MITTAL[4], (Member, IEEE)

[1]Department of Communication Engineering, Hohai University, Nanjing 211100, China
[2]Department of Electronics BCC, Central South University of Forestry and Technology, Changsha 410004, China
[3]College of Computer and Information Science, Prince Sultan University, Riyadh 11586, Saudi Arabia
[4]Department of Information Science and Engineering, Kyoto Sangyo University, Kyoto 603-8555, Japan

Corresponding author: Mohit Mittal (mohitmittal@ieee.org)

ABSTRACT The distributed cooperative offloading technique with wireless setting and power transmission provides a possible solution to meet the requirements of next-generation Multi-access Edge Computation (MEC). MEC is a model which avails cloud computing the aptitude to smoothly compute data at the edge of a largely dense network and in nearness to smart communicating devices (SCDs). This paper presents a cooperative offloading technique based on the Lagrangian Suboptimal Convergent Computation Offloading Algorithm (LSCCOA) for multi-access MEC in a distributed Internet of Things (IoT) network. A computational competition of the SCDs for limited resources which tends to obstructs smooth task offloading for MEC in an IoT high demand network is considered. The proposed suboptimal computational algorithm is implemented to perform task offloading which is optimized at the cloud edge server without relocating it to the centralized network. These resulted in a minimized weighted sum of transmit power consumption and outputs as a mixed-integer optimization problem. Also, the derived fast-convergent suboptimal algorithm is implemented to resolve the non-deterministic polynomial-time (NP)-hard problem. In conclusion, simulation results are performed to prove that the proposed algorithm substantially outperforms recent techniques with regards to energy efficiency, energy consumption reduction, throughput, and transmission delay performance.

INDEX TERMS Energy efficiency, MEC, NP-hard problem, SCD, cooperative offloading.

I. INTRODUCTION

In recent years, the implementation of the internet of things (IoT) network allows smart communicating devices (SCDs) embedded with sensors the capacity to interconnect through internet infrastructures. However, the SCDs are resource-constraint with low processing capacity and limited battery lifetime [1] to fully satisfy the demands of the mobile users. The exponential growth of smart communicating devices requires a high demand for network bandwidth and storage capabilities. In order to overcome the above challenge, multi-access edge computing (MEC) has been initiated.

The associate editor coordinating the review of this manuscript and approving it for publication was Eklas Hossain.

The introduction and recent innovations of MEC have provided SCDs with a systematic network model which avails the usability of cloud computing aptitudes at the radio access network (RAN) edge. MEC is gradually changing cloud computing services to facilitate the high performance of distributed IoT networks [2], [3]. Specifically, MEC offloading can enhance the IoT smart devices by offloading high computation tasks to the proximity of edge servers with the priority to minimize their energy consumption. We can justify the fact that offloading is needed regularly due to limited computational power, low mobile device storage capacity, and high energy consumption. Since the MEC cloud server has a high computation capability, its deployment will significantly boost faster data processing in IoT networks [4], without adding extra processing power. The MEC computation offloading [5] can be performed either locally or the computational task is transfer to the nearest cloud edge server for data processing.

Cooperative allocation of the computation task among mobile users can mitigate uneven computation workloads distribution and computation resources. The authors in [6], [7] stated that to achieve a reliable MEC computation offloading in IoT networked devices remains a challenging issue. This has resulted and formed the basis of this research.

In this paper, we propose an energy-efficient suboptimal algorithm for cooperative offloading based on a Lagrangian offloading algorithm for MEC computation in distributed IoT networks. The SCDs compete for limited computation resource that impedes smooth task offloading for multi-access MEC in high demand environments. In the proposed system, task scheduler in each SCD contains dual servers; the local central processor (LCP) which locally executes tasks, and the wireless transmitters (i.e., LTE-A or Wi-Fi) which offloads the task non-locally. Without relocating to the central network, the offloading task will be optimized at the cloud server. The next target will be to minimize the total energy consumed by the SCDs. For this, we mutually optimized the computation speed for data transmission, the transmit power allocation per sub-channel, and the offloading ratio, leading to a mixed-integer optimization problem which is an NP-hard power problem. Using the Lagrangian dual decomposition approach, an algorithm based on suboptimal convergent is proposed to mitigate this problem, and to improve the data transmission throughput which in turn minimizes the energy consumption of computational tasks offloading.

Therefore, we summarized our contributions in this paper as follows:

- We propose an energy-efficient cooperative offloading algorithm for multi-access edge computing in the distributive IoT network, where the offloading task is optimized at the cloud edge server.
- With the aim of minimizing the total transmission and computational power consumed by the SCDs, our techniques mutually optimized the computation of data transmission speed, offloading ratio, and transmit power allocation per sub-channel and the outcome becomes a problem of mixed-integer optimization. The challenge inner- and inter-coupling that influences each transmitting SCDs is tackled by joint optimization of the computational speed through Dynamic Voltage Scaling (DVS) technique, subchannel distribution, subchannel transmit power, amount of data transmitted per subchannel, and the subchannel offloading ratio.
- We propose a suboptimal convergent algorithm by applying the Lagrangian dual decomposition technique, to improve the NP-hard problem and enhance latency requirements so as to minimize the energy consumed in the computation of SCDs tasks.
- Finally, our proposed technique is validated and the simulation results show a significant out-performance of existing techniques as regards data throughput and energy consumption.

The rest of this paper is arranged as follows: Section II contains related works while the system model is contained in Section III. The research problem is formulated in Section IV. Section V describes the Multi-access computation offloading scheme analysis, and the Simulation results for the system performances are presented in Section VI while the Conclusion is stated in Section VII.

II. RELATED WORKS

The advent of MEC has formed a novel computing model to enhance data processing in proximity to SCDs and connected things at the network edge [8]. Many recent works surveyed for cooperative SCDs are aimed to improve the performance of IoT networks, subject resource-intensive constraints such as network bandwidth capacity [9], computation offloading, and energy consumption budgets [10]. In [11], a distribution of computation load for smart mobile devices was studied by employing both computations offloading and radio resource constraints. [12] proposed a clustering algorithm for load balancing in heterogeneous networks to minimize energy consumption while sustaining and satisfying users' demands. The effective computation offloading among the energy-constrained MEC is important to avoid huge computation latency in order to achieve a high QoS in the network.

The overall MEC systems performance solely depends on offloading technique design, which has a close relationship with the type of applications SCDs run. While multiple SCDs simultaneously transmit similar or different highly intensive computation tasks [13], because of parallel local and cloud execution, the ideal system performances do not only assume inner-coupling for each transmitting SCD, but is correspondingly inter-coupled amid SCDs because of the competition for limited transmission resource. The inner- and inter-coupling also affects each other, and therefore complicates multiple SCDs offloading technique and strategy [11].

Recent researches have broadly studied the offloading technique and strategy for multiuser MEC schemes [14]–[16]. Most recent studies have jointly investigated the computation offloading, not just with resource allocation constraints, but coupled with caching techniques [17]. In [18], an energy-efficient autonomic offloading (EEAO) technique that jointly applies the physical layer design and latency for application running was designed. The energy consumption was modeled for computation task for identical mobile consumptions in MEC environment. Thus, the computation offloading by mobile SCDs is randomly derived as a partly Markov decision approach to reduce the cost of MEC systems, consisting of the offloading execution time and the energy consumption [19]. In [20], the optimal energy efficiency performance in mobile-edge computing was investigated. The computation offloading technique was observed to have achieved reduced energy consumption for each mobile user in allocated time slot. The authors in [21] studied an energy efficient task offloading (EETO) in 5G MEC based on a two-tier small-cell network setting. By jointly examining the energy cost in offloading task for backhaul communication links, the minimized energy consumption problem was formulated. An algorithm is designed to improve the computation offloading task in order to realize global convergence. Therefore, a joint optimized transmit power, required number of data transmission throughput and the CPU cycles for mobile devices achieved a low energy consumption, however, the systems latency was not greatly improved as expected [22], [23].

Moreover, due to the high demands for SCDs in a computation system, the energy-efficient offloading is critical target in the construction of an effective computation offloading system in MEC networks. In [24], the authors considered some energy-efficient offloading strategies in task computational transcoding for edge-cloud mechanism. An algorithm based on online offloading was proposed with a focus of achieving a minimized energy consumption while reaching low latency. To enhance energy efficiency in MEC, a multiuser-based computation offloading problem was investigated by [25], and [26] used a game theoretic technique to design a distributive algorithm for a wireless network based on multi-channel. Different from these studies, our focus is on improved MEC which does not only minimize energy consumption but also improves throughput and latency between transmitting mobile SCDs. Hence, we propose an optimal cooperative offloading technique to enhance energy efficiency in dynamic IoT networks of the cloud computing system. However, recent works did not give much consideration to the influence terminal execution technique has on multi-access computation which results in highly deteriorated network performance [27]–[31]. This also contributes to the focus of our research.

III. SYSTEM MODEL

The system model consisting of the network, communication and computation models of proposed techniques is analyzed in this section.

A. NETWORK MODEL

As described in Figure 1, a MEC-based cooperative computation offloading setting in IoT Network is considered. In the network, the numbers of SCDs are alternatively enclosed by a base station (BS), a nearby small cell (SC) transmitters, such as Wi-Fi AP or an LTE-A, coupled with a wireless relay (WR). With the aim of availing all transmitting SCDs the MEC services, either a single or multiple MEC servers is linked to both the (BS) and (SC) through the fiber links. Hence, SCDs tasks computations is either offloaded at the (BS) via the connected MEC server, the (SC) connected MEC server, or indirectly to the (BS) through the (WR) and the wireless transmitters (WT). The fiber links is used in connecting the (SCs) and the (BS) to the central network. With the aim of achieving an efficient spectrum reuse, the same frequency band is shared between the (WR) and the (BS) during computation and task transmission.

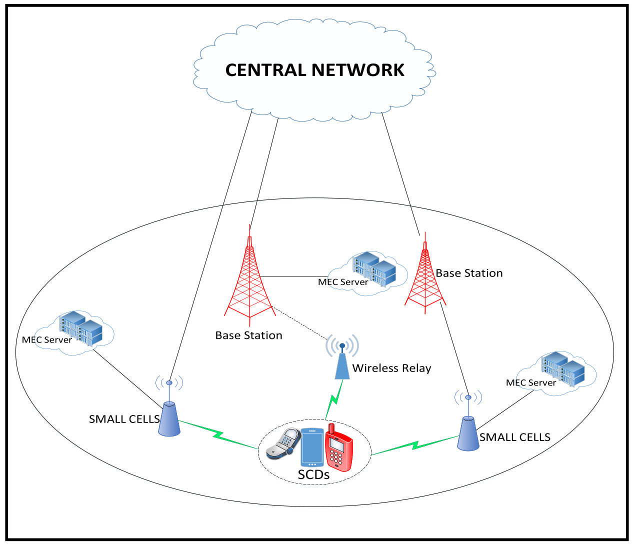

FIGURE 1. MEC cooperative computation offloading in IoT network.

The rate at which the task arrives at the SCD i's scheduler is given as δ_{SCD_i}. The decision of the SCD i task scheduler is implemented according to the offloading probability α_i, rate. This represents the probability of offloading an inbound task is to the MEC server. The tasks computed are presumed to be indivisible, therefore it is impossible to further break them down into sub-tasks. The system further assumes a task scheduler is incorporated in all the MEC servers, which either selects to compute an inbound task or to offload the task to the remote cloud. For the SCDs, the inbound task through the MEC server S will either be offloaded remotely to the cloud with a probability β_j or can be locally executed with probability $(1 - \beta_j)$.

In this MEC scenario, assuming SCDs in unit cell will function as [26]. During data processing, the SCDs send SC a request. Using the information gathered the SC fashion out a strategy for computation task offloading and updating independent SCD.

Considering a group of SCDs represented by $R = \{1, 2, \ldots, R\}$ is presented. It is assumed that only one computation task C_i is available to be executed on SCD i during the period of computation offloading. Recall that all the computation tasks are indivisible. Therefore, there are two terms that may clearly define the computation task C_i i.e., $C_i = (s_i, p_i)$, where S_i represents the size of input data of task C_i, while p_i represents the amount of LCP sequences required in achieving C_i. We assume that there are M uplink channels connected to the BS and represented as $M = \{1, 2, \ldots, M\}$, and N uplink channels connected to the SC and characterized as $N = \{1, 2, \ldots, N\}$.

Considering the SCDs features and their tasks, for example the density of workload, the computing capacity, the energy consumption, and the data size of the task, the SCD users are categorized into two. Let $\lambda_i \in \{0, 1\}$ denote the offloading result of task C_i, where $\lambda_{i(0)}$ implies that C_i is intended to be accomplished locally at SCD i's LPC, and $\lambda_{i(1)}$ implies that

C_i will be offloaded either locally or to the MEC server that is indirectly connected to the k through the WR and the wireless transmitter. This kind of offloading decision is dependent of the quality of communication and the channel.

B. COMMUNICATION MODEL

An identical bandwidth B in the subchannels is solely allocated without channel interference among the SCDs [12], [13]. Assuming $P_{r,m}$, $e_{r,m}$ and σ^2 indicate the transmit power of SCD r on subchannel m, Gaussian noise, and the coefficient of uplink channel fading from SCD r to subchannel m respectively. Thus, the rate of transmission of SCD r on subchannel m is given by

$$L_T = Blog_2\left(1 + \frac{P_{r,m}d_l^{-k}|e_{r,m}|^2}{\sigma^2}\right) \tag{1}$$

C. COMPUTATION MODEL

This section describes the computation offloading technique in two scenarios, where U_i and V_i indicates size of the input data and the latency requirements:

- Local computing: Power consumed P by CPU is $P = \tau S^3$, where τ is the coefficient subject to chip design and S is the computational speed of CPU based on dynamic voltage scaling technique that improves energy management scheme. Assuming the computational speed of SCD is S_{u_i} and the task execution time t_{u_i}. Therefore, the task execution time is given by

$$t_{u_i} = \frac{\alpha_i U_i.\gamma_i}{S_{u_i}} \tag{2}$$

  And the energy consumed E_{u_i} is

$$E_{u_i} = \alpha_i U_i.\gamma_i.\tau S^3 \tag{3}$$

- Computation Offloading to Central network: Let $(1-\gamma_i)U_i$ denotes the bits offloaded to the MEC cloud server at uplink data $\psi(1-\gamma_i)U_i$, where ψ is the uplink transmission overhead. Let $u_{r,m}$ represents the size of the data offloaded to subchannel m. Thus, we expressed transmission time as

$$l_{c_i} = \max\left\{\frac{u_{i,m}}{L_T}, \forall_m\right\} \tag{4}$$

  And the energy consumed for transmitting data to the MEC cloud server as

$$E_{c_i} = \sum_{m=1}^{M}\left[\frac{\mu(\beta_{r,m}P_{r,m}u_{i,m})}{L_T}\right] \tag{5}$$

where μ denotes the reverse efficiency of power amplifier. It is assumed that cloud task computation capacity is infinite. Thus, we can ignore the cloud computing time. In addition, the downlink transmission cost could be ignored due to less data received compared to the large downlink transmission rate [4], [31]. Table 1 presents a description of all mathematical parameters and their derivations.

TABLE 1. Network parameters and derivations.

Parameter	Description
U_i	Input data size
V_i	Network Latency requirement
α_i	Offloading probability
β_j	Probability of offloading to remote cloud
$(1-\beta_j)$	Probability of local execution
C_i	System Computation task
$P_{r,m}$	Transmit power of SCDs
$e_{r,m}$	Coefficient of uplink channel coding
σ^2	Gaussian Noise
$P = \tau S^3$	CPU power consumption
τ	Coefficient of chip design
S	CPU computational speed based on DVS
S_{u_i}	Computational speed of SCD at local computing
t_{u_i}	Execution time of local computing
E_{u_i}	Energy consumed at local computing
E_{c_i}	Energy consumed at MEC server transmission
$P_1 \ldots P_3$	MCO formulated problems
$D_1 \ldots D_9$	Constraints for measuring energy consumption
$P_{r,m}^*$	Optimal power auxiliary
ζ	Pre-defined error tolerance point
$\Delta\mu_r$	Lagrange Multipliers Update
λ_r^*	Optimal offloading ratio
$s_{r,m}^*$	Optimal communicated bits
ϖ_i	SCDs offloading priority
$g_{i,k}$	System channel gain

For the $\lambda_{i(3)}$ category of SCDs whose task offloading is directed either locally or to the transmitting MEC server which is indirectly connected to base station (k), we set different offloading priorities for them and is defined as

$$\varpi_i = \frac{g_{i,k}P_{r,m}}{\sqrt{E_{c_i}}} \tag{6}$$

where $g_{i,k}$ is set as the systems channel gain. **Algorithm 1** illustrates a comprehensive SCD user offloading priority selection.

D. COMPUTATION PROBLEM FORMULATION

The Multi-access computational offloading (MCO) problems $P1 \ldots P3$ are formulated in this section.

$$P_1: \min_{g_l, B, P, L, \lambda} \sum_{r=1}^{R} x_r E_r(g_{l_r}, b_{r,m}, P_{r,m}, l_{r,m}, \lambda_r)$$

$$s.t. D_1: \max\{t_{l_r}, t_{a_r}\} \leq L_r, \quad \forall r,$$

$$D_2: 0 \leq g_{l_r} \leq F_{l_r}, \quad \forall r,$$

$$D_3: \sum_{m=1}^{M} b_{r,m}P_{r,m} \leq P_{T_r}, \quad \forall r,$$

$$D_4: P_{r,m} \geq 0, \quad \forall m, r,$$

$$D_5: \sum_{m=1}^{M} b_{r,m}k_{r,m} \geq \beta(1-\lambda_r)J_r, \quad \forall r,$$

$$D_6: l_{r,m} \geq 0, \quad \forall m, r,$$

$$D_7: \sum_{r=1}^{R} b_{r,m} \leq 1, \quad \forall m,$$

$$D_8: b_{r,m} \in \{0, 1\}, \quad \forall m, r,$$

$$D_9: 0 \leq \lambda_r \leq 1, \quad \forall r,$$

Algorithm 1 Priority Selection Algorithm in Offloading Process

Initialize:
Smart Communicating Devices: $R = \{1, 2, \ldots, R\}$;
Wireless uplink channels:$N = \{1, 2, \ldots, N\}$;
SCD Computation task: $C_i = (s_i, p_i)$;
SCD Transmission power: $P_{r,m}, i \in R$;
SCD category: $\lambda_{i(0)} = \lambda_{i(1)} = \phi$;
Selection priority: $\varpi = \phi$;

for SCD $i = 1 to R$ **do**.
for transmission channel $j = 1 to N$ **do**.
    compute the channel data execution rate t_{ui} of each SCDs as stated in (2), and the energy consumption E_{ui} represented in (3);
for transmission channel $j = 1 to N$ **do**
    if $E_r \leq E_{ui}$ **then**;
    $i \Rightarrow \lambda_{i(0)}$;
    else
    $i \Rightarrow \lambda_{i(1)}$;
    $\varpi_i = g_{l,r} P_{r,m} / \sqrt{E_{c_i}}$;
    end if
end for
Output:
    Selected SCD category: $\lambda_{i(0)}, \lambda_{i(1)}$;
    Selection priority of SCD: $\varpi = \{\varpi_i\}, i \in \lambda_{i(1)}$.

where $E_r(g_{l_r}, b_r, _m, P_{r,m}, l_{r,m}, \lambda_r)$ denotes the energy consumption (EC) of SCDs r, and can be represented as $E_r(g_{l_r}, b_{r,m}, P_{r,m}, l_{r,m}, \lambda_r) = E_{l_r} + E_{i_r}$. Furthermore, $x_r, L_{r,}, G_{l_r}$, and P_{T_r} are respectively considered as the weighting measure, computational latency, optimal velocity for computation, and optimal transit power allocation (of SCDs r).

Our objective in P_1 is to lessen the weighted amount of power consumed by communicating devices subject to $D_1 - D_9$. In this regard, the weighted sum is considered as the tradeoff of power consumed by the SCDs. The weighting sum value measures relatively in reflection to the importance of SCDs. Technically, SCDs with less residual energy could be assigned greater weighting measures. We therefore explains the constraints as follows; D_1 ensures a guaranteed response time of communicating SCDs; D_2, D_3 and D_4 represents the maximum computational velocity and the maximum transit power allocation assigned to individual SCD; D_5 and D_6 ensures that all are task offloaded and transmitted over an allocated sub-channel; D_7 and D_8 guarantees the assignment of maximum of one SCD for each sub-channel for uplink communication. Considering that $b_{r,m}$ assumes an integer state, P_1 is a non-deterministic polynomial-time hardness (NP-harder).

IV. MULTI-ACCESS COMPUTATIONAL OFFLOADING SCHEME

In this section, we attempt to derive the multi-access computational offloading, hence, in close forms, transmit power allocation (TPA) and an optimal computation velocity (OCV), is calculated while a suboptimal algorithm based on Lagrangian dual decomposition (LDD) is proposed.

A. OPTIMAL COMPUTATIONAL VELOCITY AND TRANSMIT POWER ALLOCATION

In this subsection, simplifying P_1 by minimizing the variables over g_{l_r} and first $P_{r,m}$ we attempt to compute the OCV and TPA. Hence, the optimal power g_{l_r} generated for computation is then obtained as

$$g^*_{l_r}(\lambda_r) = \frac{\alpha_r \lambda_r J_r}{L_r} \tag{7}$$

While the optimal power auxiliary $P_{r,m}$ derived from (1) is given as

$$P^*_{r,m}(b_{r,m}, l_{r,m}) = \frac{1}{f_{r,m}} \left(2^{\frac{l_{r,m}}{XL_r}} - 1 \right). \tag{8}$$

Further analyses of the multi-access computational problem in P_1 indicates that at every increase of the optimal power (g_{l_r}), the power consumption of SCDs r also increases in a monotonic form. Hence, $t_{l_r} \leq L_r$ is further derived and this further results in $g_{l_r} \geq \frac{\alpha_r \lambda_r J_r}{L_r}$.

Therefore, the optimal computational velocity problem (P_2) is formulated as;

$$\begin{aligned}
P_2\text{: } & \min_{B,L,\lambda} \sum_{r=1}^{R} x_r \left[\tfrac{\varepsilon(\alpha_r J_r)^3}{L_r^2} \lambda_r^3 + E_{ir}(b_{r,m}, l_{r,m}). \right] \\
& s.t. D_{10} : \lambda_r \leq \tfrac{L_r F_{lr}}{\alpha_r J_r}, \quad \forall r, \\
& D_{11} : \sum_{m=1}^{M} \tfrac{b_{r,m}}{f_{r,m}} \left(2^{\frac{l_{r,m}}{XL_r}} - 1 \right) \leq P_{T_r}, \quad \forall r, \\
& D_5, D_6, D_7, D_8, D_9
\end{aligned}$$

B. SUBOPTIMAL LAGRANGIAN DUAL DECOMPOSITION ALGORITHM

This subsection applies the Lagrangian dual decomposition Algorithm (LDDA) to resolve the aforementioned simplified computational problem (P_2). Diminishing $b_{r,m}$ to $0 \leq \widetilde{b}_{r,m} \leq 1$, we introduce a fresh variable $s_{r,m} = \widetilde{b}_{r,m} l_{r,m}$, hence, the diminished problem can be stated as P_3 below:

$$\begin{aligned}
P_3\text{: } & \min_{\tilde{B},L,\lambda} \sum_{r=1}^{R} x_r \left[\tfrac{\varepsilon(\alpha_r J_r)^3}{L_r^2} \lambda_r^3 + E_{ir}(\tilde{b}_{r,m}, s_{r,m}) \right] \\
& s.t. D_{12} : \sum_{m=1}^{M} \tfrac{\tilde{b}_{r,m}}{f_{r,m}} \left(2^{\frac{s_{r,m}}{XL_r \tilde{b}_{r,m}}} - 1 \right) \leq P_{T_r}, \quad \forall r, \\
& D_{13} : \sum_{r=1}^{R} \tilde{b}_{r,m} \leq 1, \quad \forall m, \\
& D_{14} : 0 \leq \tilde{b}_{r,m} \leq 1, \quad \forall m, r, \\
& D_{15} : \sum_{m=1}^{M} s_{r,m} \geq \beta(1 - \lambda_r) J_r, \quad \forall r, \\
& D_{16} : s_{r,m} \geq 0, \quad \forall m, r, \\
& D_9, D_{10,}
\end{aligned}$$

where

$$E_{ir}(\tilde{b}_{r,m}, s_{r,m}) = \sum_{m=1}^{M} \frac{\rho L_r}{f_{r,m}} \tilde{b}_{r,m} \left(2^{\frac{s_{r,m}}{XL_r \tilde{b}_{r,m}}} - 1 \right). \quad (9)$$

In the expression $\psi(\tilde{b}_{r,m}, s_{r,m}) = \left(2^{\frac{s_{r,m}}{XL_r \tilde{b}_{r,m}}} - 1 \right)$, it can be seen that $\psi(\tilde{b}_{r,m}, s_{r,m})$ is jointly convex in $\tilde{b}_{r,m}$ and $s_{r,m}$, since it's a positive semi-definite Hessian matrix. Similarly, the objective and feasible regions of (P_2) are convex and (P_3) is convex as well. Therefore, obtaining the optimal solutions by applying the LDDA by denoting: $\mu = (\mu_1 ... \mu_R)0$, $v = (v_1 ... v_M)0$, as the respective Lagrange multipliers which are consistent with D_{12},D_{13} and D_{15}, hence, the (P_3) LDD is expressed as;

$$\begin{aligned} L(\tilde{B}, S, \lambda, \mu, v, \gamma) = & \sum_{r=1}^{R} x_r \left[\frac{\varepsilon(\alpha_r J_r)^3}{L_r^2} \lambda_r^3 + E_{ir}(\tilde{b}_{r,m}, s_{r,m}) \right] \\ & + \sum_{r=1}^{R} \mu_r \left(\sum_{m=1}^{M} \frac{\psi(\tilde{b}_{r,m}, s_{r,m})}{f_{r,m}} - P_{T_r} \right) \\ & + \sum_{r=1}^{R} \gamma_r \left(\beta J_r (1 - \lambda_r) - \sum_{m=1}^{M} s_{r,m} \right) \\ & + \sum_{m=1}^{M} v_m \left(\sum_{r=1}^{R} \tilde{b}_{r,m} - 1 \right). \end{aligned} \quad (10)$$

1) OPTIMIZATION VARIABLE UPDATE

The optimized LDD function can be expressed as;

$$H(\mu, v, \gamma) = \inf_{\{\tilde{B}, S, \lambda \in \Phi\}} L(\tilde{B}, S, \lambda, \mu, v, \gamma), \quad (11)$$

as Φ is considered the area consistent with D_9,D_{10},D_{14}, and D_{16}. Combining D_9 and D_{10} will yield

$$0 \leq \lambda_r \leq \min \left\{ 1, \frac{L_r G_{lr}}{\alpha_r J_r} \right\} \triangleq \lambda_{r,\max}. \quad (12)$$

By denoting the optimal solutions of (13) as $\tilde{b}^*_{r,m}$, $s^*_{r,m}$ and λ^*_r, giving the KKT conditions as:

$$\frac{\vartheta L(\tilde{B}, S, \lambda, \mu, v, \gamma)}{\vartheta \tilde{b}^*_{r,m}} \begin{cases} > 0, \tilde{b}^*_{r,m} = 0 \\ = 0, 0 < \tilde{b}^*_{r,m} < 1, \\ < 0, \tilde{b}^*_{r,m} = 1 \end{cases} \forall m, r, \quad (13)$$

$$\frac{\vartheta L(\tilde{B}, S, \lambda, \mu, v, \gamma)}{\vartheta s^*_{r,m}} \begin{cases} > 0, s^*_{r,m} = 0 \\ = 0, s^*_{r,m} > 0 \end{cases}, \quad \forall m, r, \quad (14)$$

$$\frac{\vartheta L(\tilde{B}, S, \lambda, \mu, v, \gamma)}{\vartheta \lambda^*_r} \begin{cases} > 0, \lambda^*_r = 0 \\ = 0, 0 < \lambda^*_r < \lambda_{r,\max}, \\ < 0, \lambda^*_r = \lambda_{r,\max} \end{cases} \forall r \quad (15)$$

To minimize $L(\tilde{B}, S, \lambda, \mu, v, \gamma)$ for given parameters (μ, v, γ), we solved three subproblems (Q_1, Q_2) and (Q_3) as follows:

(Q_1)**: *Optimal communicated bits (OCBs) on each sub-channel:***

By and differentiating $L(\tilde{B}, S, \lambda, \mu, v, \gamma)$ for a given parameter $\tilde{B}$, subject to $s_{r,m}$ further substituting the result into (14), we obtained:

$$s^*_{r,m} = \left[XL_r \tilde{b}_{r,m} \log_2 \left(\frac{\gamma_r f_{r,m} X}{\left(\rho + \frac{\mu_r}{L_r}\right) ln2} \right) \right]^+, \quad (16)$$

since $[x]^+ = \max\{0, x\}$, therefore, $l^*_{r,m}$ is expressed as:

$$l^*_{r,m} = \frac{s^*_{r,m}}{\tilde{b}_{r,m}} = \left[XL_r log_2 \left(\frac{\gamma_r f_{r,m} X}{\left(\rho + \frac{\mu_r}{L_r}\right) ln2} \right) \right]^+, \quad (17)$$

(Q_2)**: *Optimal sub-channel assignment:***

Since the OCBs on each sub-channel is achieved, we therefore generate their optimal sub-channel assignment (OSA) as follows:

$$\frac{\vartheta L(\tilde{B}, S, \lambda, \mu, v, \gamma)}{\vartheta \tilde{b}_{r,m}} = \frac{\left(\left(1 - \frac{s_{r,m} \ln 2}{XL_r \tilde{b}_{r,m}} \right) 2^{\frac{s_{r,m}}{XL_r \tilde{b}_{r,m}}} - 1 \right)}{\frac{\rho L_r + \mu_r}{f_{r,m}} + v_m} \quad (18)$$

As we substitute (16) into (18) using (13), we obtained the equation below:

$$\tilde{b}^*_{r,m} = \begin{cases} 0, & v_m > D_{r,m} \\ 1, & v_m < D_{r,m}, \end{cases} \quad (19)$$

where $D_{r,m}$ is expressed as

$$D_{r,m} = \frac{\rho L_r + \mu_r}{f_{r,m}} \left(1 - \left(1 - \frac{l^*_{r,m} \ln 2}{XL_r} \right) 2^{\frac{l^*_{r,m}}{XL_r}} \right). \quad (20)$$

Beginning from (20), it is expected that the sub-channel will be allocated to the transmitting SCDs with the maximal $D_{r,m}$, particularly

$$b^*_{r,m} = \begin{cases} 1, & r = \arg\max_{1 \leq r \leq R} D_{r,m} \\ 0, & else, \end{cases} \quad (21)$$

which rounds $b_{r,m}$ to an integer.

(Q_3)**: *Optimal offloading ratio (OOR):***

In an attempt to solve the OOR, differentiating $L(\tilde{B}, S, \lambda, \mu, v, \gamma)$ w.r.t. λ_r and further substituted the result into (15) to obtain

$$\lambda^*_r = \min \left\{ \frac{L_r}{\alpha_r J_r} \sqrt{\frac{\beta \gamma_r}{3\alpha_r \varepsilon}}, \lambda_{r,\max} \right\}. \quad (22)$$

2) LAGRANGE MULTIPLIERS UPDATE

It is observed that for a given μ and γ, $\tilde{b}^*_{r,m}$, $l^*_{r,m}$ and λ^*_r can be obtained. A subgradient projection of $H^-(\mu, v, \gamma)$ can therefore be expressed as

$$\Delta \mu_r = P_{T_r} - \sum_{m=1}^{M} \frac{\tilde{b}^*_{r,m} \left(2^{\frac{s^*_{r,m}}{XL_r \tilde{b}^*_{r,m}}} - 1 \right)}{f_{r,m}}, \quad (23)$$

$$\Delta \gamma_r = \sum_{m=1}^{M} s^*_{r,m} - \beta J_r (1 - \lambda^*_r), \quad (24)$$

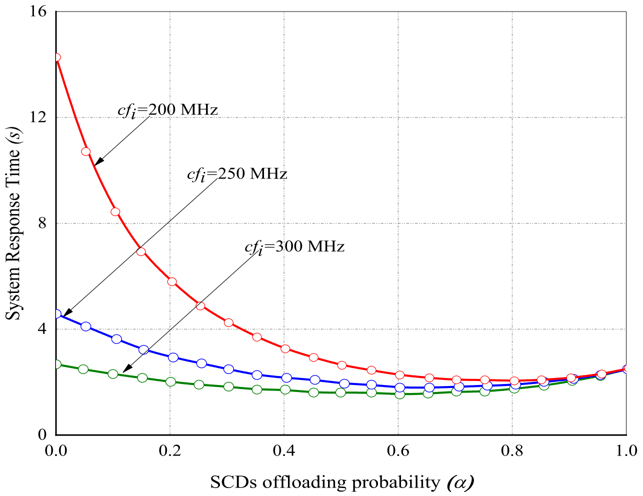

FIGURE 2. Probability of SCD offloading vs System response time.

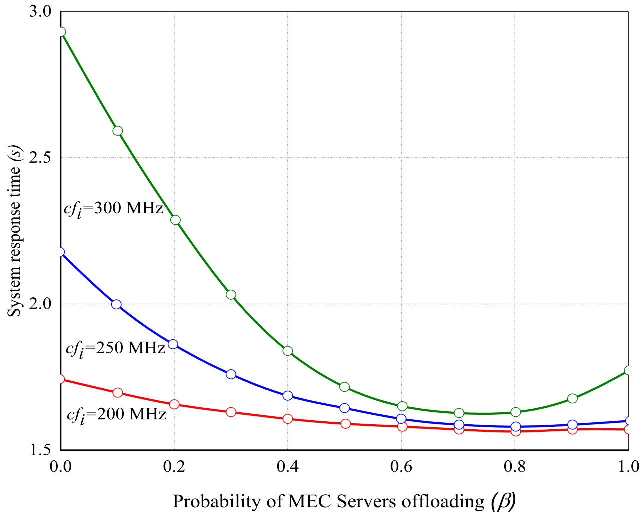

FIGURE 3. Probability of MEC server offloading vs system response time.

Algorithm 2 Lagrangian Suboptimal Convergent Computation Offloading Algorithm (LSCCOA)

Initialization
$\zeta, \mu, \nu, \gamma, t$
while $\| \mu_r(t+1) - \mu_r(t)\|_2 + \| \gamma_r(t+1) - \gamma_r(t)\|_2 > \zeta$ **do**
  Calculate $b^*_{r,m}$, $l^*_{r,m}$ and λ^*_r
  Update μ and γ
end while
  Calculate $g^*_{l_r}(\lambda_r)$ and $P^*_{r,m}(b_{r,m}, l_{r,m})$
end

Algorithm 2 shows a detailed analysis of the suboptimal offloading performance, as ζ is set as an initial error tolerance point.

V. SIMULATION RESULTS

Using an LTE-A network coupled with a fiber Wi-Fi network parameter, we performed the following numerical analysis. For our experiment, 80 smart communication devices (SCDs) were randomly deployed within the range of $80m$ from each network base station (N-BS). Additionally, six SCDs are positioned across the area of network coverage of each access point (AP). Channel gain of the cellular access mode is set as $g_{i,k,} = c^{-\sigma}_{i,k}$ amid SCD base station k and SCD i, representing $c_{i,k}$ as the spatial separation between base station k and SCD i, while $\sigma = 6$ as the factor for network path loss. We also set $\alpha_i = \alpha(\forall i = 1, 2, 3 \ldots)$ and $\beta_s = \beta(\forall s = 1, 2, 3 \ldots)$.

Considering $|\Re| = 6$ MEC servers, each linked to η end-users, which are capable of controlling their task offloading flexibly as their probability of computation offloading change. Figure 2 illustrates the performance of the network average response time against the system offloading probability α of SCDs. From the illustration, the observed SCDs experienced a reduction in their latency by configuring their individual offloading probabilities subsequent to technique we proposed. It is important to note that the latency of local computation severely rely on the competence of computation of each SCDs, however, the reliance system latency on the capabilities of computation is substantially minimized in our proposed cooperative offloading scheme. For example, assuming the clock frequency cf_i (LCP processing cycles/second) for SCD i's increases from $200MHz$ to 300MHz, the local computations average response time will decrease from 14.5 s to $3.2s$, as against $1.9s$ to $1.4s$ performance in our proposed technique.

We further examine the relevance of the coordination between the remote cloud and MEC servers in Figure 3. Our observation proves that although response time is minimized in the remote cloud-computing when compared to the edge-only computation, however, our proposed technique significantly outperforms both techniques, as long as the MEC servers' offloading probability α is set accordingly.

Our proposed Lagrangian LSCCOA algorithm is compared against EEAO proposed in [18] and EETO in [21] for energy efficiency for different number of SCDs in Figure 4. Considering EEAO and EETO, their objectives concentrated mainly on minimizing energy consumed in sub-channel assignment and the distribution of power in the system, respectively, However, in our proposed technique, by setting the maximum power for transmission at $40dBm$ and the system sum-rate coefficient at 0.42.

It is observed that the proposed LSCCOA algorithm outperforms the compared EEAO and EETO in terms of energy efficiency. As the SCDs increases, the EE slightly increase and then stabilizes as computation continues in all the technique. Due to the co-channel interference minimized effect, the transmitting density of all SCDs is minimized as well. However, due to the allocation of more subchannels, the system performance is constrained by the limited system resources as the number of SCDs increases. A decline of EE is also observed with the increasing requirement of minimum data rate in the technique, because, there is need for the base stations to optimize the subchannels transmit power in order

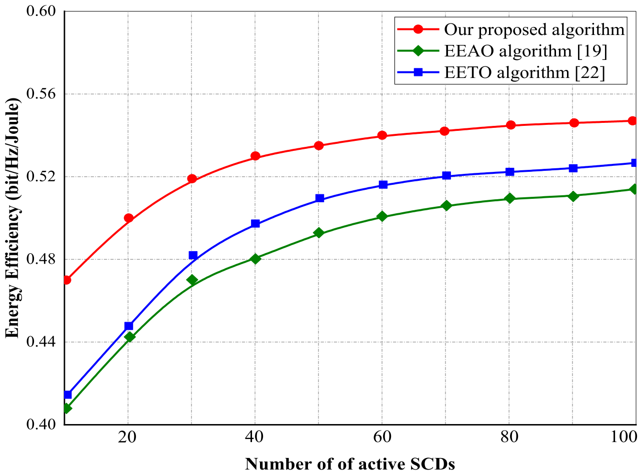

FIGURE 4. Energy efficiency computing performance under different algorithms.

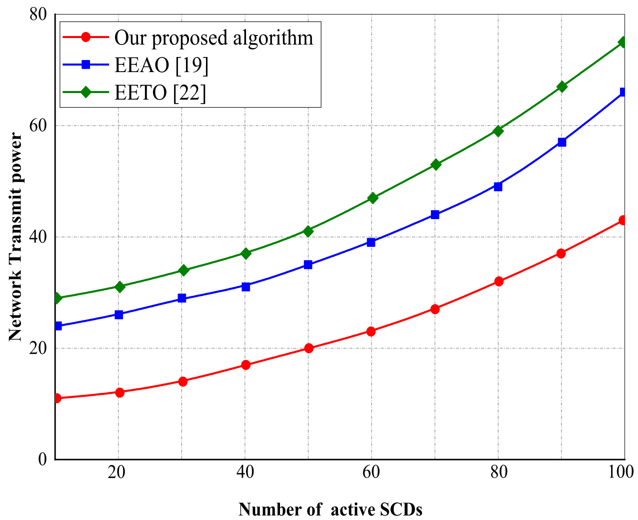

FIGURE 5. Transmit power performance under different algorithms.

to preserve the requirements of the system throughput which negatively affects energy efficiency.

In Figure 5, relative transmission energy consumed by different active SCDs in LSCCOA is observed to be less than that of EEAO and EETO, this is because, our proposed optimal computational transmit power distribution sturdily controls the transmit power existing at the unallocated sub-channels necessitated in minimized levels of transmit power. With an increase of the active SCDs, all available spectrums are coordinated at different layers, and thus, significantly improving the interference of the co-channel. Simultaneously, it indicates that the energy efficiency increases alongside the downward minimum targets rate for both LSCCOA, EEAO and EETO, respectively, while the rate of rise declines. This is because when the threshold is high, more active SCDs experience difficulty in attaining the requirements, this in turn exhausts the transmission power required to optimize the systems performance. Therefore, this enhanced performance indicates how the proposed LSCCOA algorithms outperforms the compared techniques. Figure 6 evaluates the

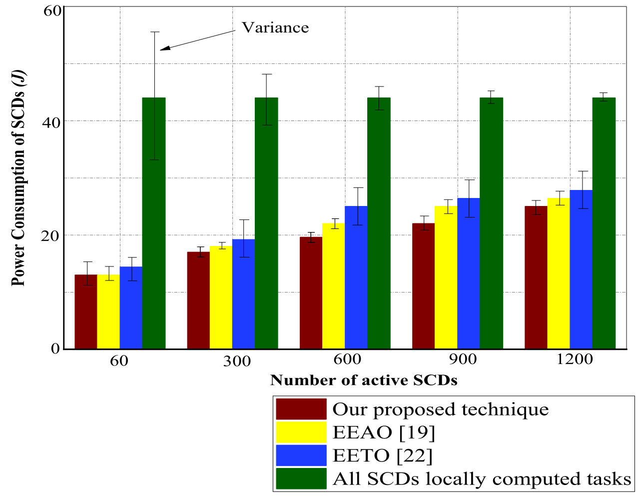

FIGURE 6. Comparison of average power consumption under different algorithms with different amount of SCDs.

average power consumed by the LSCCOA algorithm. In the evaluation, the SCDs power consumption performance with an increase in the number of active SCDs from 60 to 1200 is expressed, and compared against three other techniques. The average power consumption of each SCD in the entire system computation task is approximately $44.206J$. As compared with the local system computing technique, although the two compared algorithms save energy to an extent, however, in computation, optimally intended energy conservation is achieved by the proposed LSCCOA through task offloading.

By initiating 60 SCDs in the task computation process, all three techniques manifest an independent power consumption rate of $13.341J$, $13.412J$, and $14.751J$. When the number of SCDs is gradually increased, at 1200 transmitting SCDs, the average power consumed by these SCDs independently increases to $25.002J$, $26.521J$, and $28.063J$, respectively. This is because the same wireless channel resource is accessed concurrently by multiple SCDs for a simultaneous task offloading implementation. Hence, this results in increase of system interference. This interference between each transmitting SCD in the system as described in (13), will result in a minimized quality of communication, and thus, the computational offloading rates. Therefore, as SCDs increase to 1200, several users subscribe to use the local computing technique which results in an increase of the average power consumption of SCDs. Our proposed technique can save at least 58.6% of the power consumption.

Considering system throughput, our proposed algorithm is compared against two other techniques (EEAO and EETO) in the research. Figure 7 indicates that, as 60 SCDs were initiated in the computation process, the techniques all display an average throughput of 5823.52, 6848.51, and 7038.56 (all in bit per seconds), respectively. Although the SCDs throughput in our proposed technique is minimal at the initial stage, but, as the amount of SCDs increased from 60 to 100, further minimization is observed at the throughput rate slope than the compared two techniques. Throughout the entire

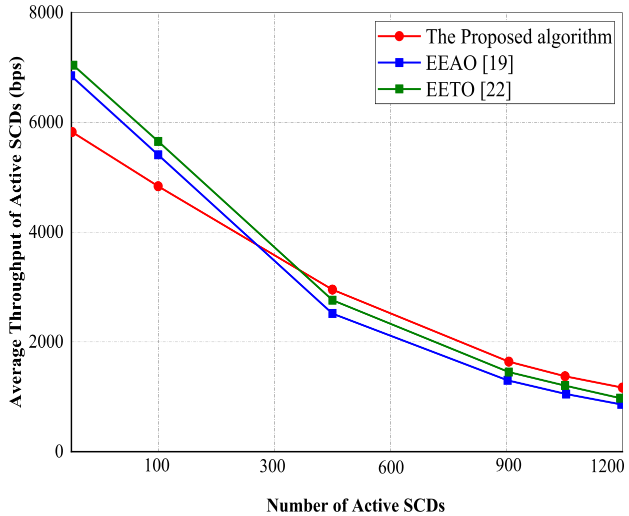

FIGURE 7. Average Throughput vs number of active SCDs under different algorithms.

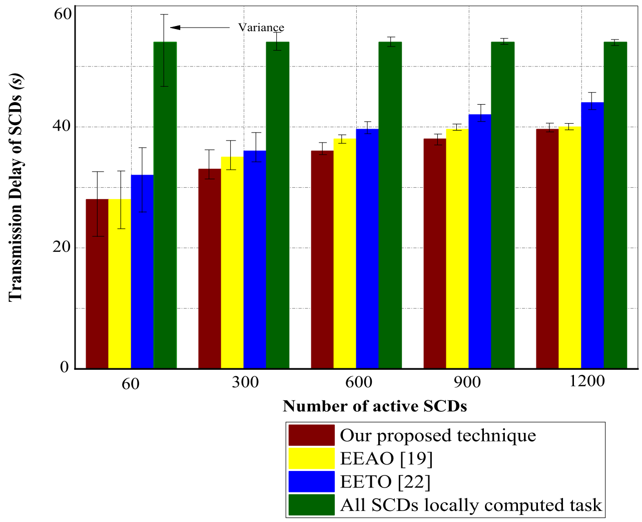

FIGURE 8. Comparison of transmission latency under different algorithms with different amount of SCDs.

computation process, it is observed that as the amount of SCDs increases, the throughput of SCDs with respect to our proposed technique grew higher than that in the other techniques. Finally, with 1200 SCDs, the techniques manifest an average throughput of 1166.2, 860.9, and 970.2 (all in bps), respectively. With an increase exponential in the amount of deployed active SCDs, there will be a relative and intensive rise in the joint interference between SCDs.

In addition, uplink data transmission rate will be decreased, which will result in the intensive surge of the power consumed by cloud computing offloading much higher the power consumption of the local MEC computing. Thus, many of the SCDs will implement their computation through the conventional MEC computing, which substitutes for offloading process. Comparing our proposed technique in this regard with the EEAO and EETO algorithms, respectively, it is observed that the throughput tends to go higher while the corresponding decline rate is relatively slowed when measured with our proposed technique.

Comparing the average latency of SCDs task execution in the proposed technique with other two techniques, Figure 8 shows that the average latency of SCDs per task execution is approximately 55.215s within the entire locally SCDs computation technique. Deploying 60 SCDs, the latency of these three techniques are 28.641s, 28.682s, and 32.543s. In comparison with the system overall computation technique, a minimum of 48.90% of this latency is conserved by the proposed technique. When 1200 SCDs are deployed, the respective latency gained by using the four techniques are 39.952s, 40.001s, 44.532s, and 55.215s. When compare with the local computation method, our proposed technique conserves about 45.41% of the computation time. This to some extent higher than what is obtainable in the performance of the compared algorithms.

Table 2 below shows a comparison analysis for the performance of all three algorithms with respect to Energy efficiency, average energy consumption, average throughput performance and Latency performances different numbers of deployed SCDs.

TABLE 2. Performance comparison of different algorithms.

Parameter	LSCCOA	EETO	EEAO
Energy Efficiency (100 SCDs)	0.55	0.528	0.51
Average Energy Consumption (60-1200 SCDs)	25.002J (saved 58.6% Energy)	26.521J	28.063J
Average Throughput (60-1200 SCDs)	1166.2bps	860.9bps	970.2bps
Average Latency (60-1200 SCDs)	39.952s (saved 41.41% consumption time)	40.001s	44.532s

VI. CONCLUSION

This paper has examined and formulated multi-access cooperative computation offloading of SCDs in an IoT network based on the DVS-enabled MEC system. The target is to improve energy efficiency by minimizing the weighted sum of energy consumed by the SCDs. The formulated NP-hard problem was addressed via the application of our proposed LSCCOA Algorithm. Finally, simulation results validate that our proposed technique attains better performance in computation offloading and also optimally performs well at every increase in the amount of SCDs.

REFERENCES

[1] H. Guo, J. Liu, and H. Qin, "Collaborative mobile edge computation offloading for IoT over fiber-wireless networks," *IEEE Netw.*, vol. 32, no. 1, pp. 66–71, Jan. 2018.

[2] X. He, Y. Chen, and K. K. Chai, "Delay-aware energy efficient computation offloading for energy harvesting enabled fog radio access networks," in *Proc. IEEE 87th Veh. Technol. Conf. (VTC Spring)*, Jun. 2018, pp. 1–6.

[3] S.-H. Kim, S. Park, M. Chen, and C.-H. Youn, "An optimal pricing scheme for the energy-efficient mobile edge computation offloading with OFDMA," *IEEE Commun. Lett.*, vol. 22, no. 9, pp. 1922–1925, Sep. 2018.

[4] K. Zhang, S. Leng, Y. He, S. Maharjan, and Y. Zhang, "Mobile edge computing and networking for green and low-latency Internet of Things," *IEEE Commun. Mag.*, vol. 56, no. 5, pp. 39–45, May 2018.

[5] S. Mao, S. Leng, K. Yang, Q. Zhao, and M. Liu, "Energy efficiency and delay tradeoff in multi-user wireless powered mobile-edge computing systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2017, pp. 1–6.

[6] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 974–983, Apr. 2015.

[7] 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, Aug. 2017.

[8] 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.

[9] 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.

[10] C. Iwendi, Z. Zhang, and X. Du, "ACO based key management routing mechanism for WSN security and data collection," in *Proc. IEEE Int. Conf. Ind. Technol. (ICIT)*, Lyon, France, Feb. 2018, pp. 1935–1939, doi: 10.1109/ICIT.2018.8352482.

[11] X. Tao, K. Ota, M. Dong, H. Qi, and K. Li, "Performance guaranteed computation offloading for mobile-edge cloud computing," *IEEE Wireless Commun. Lett.*, vol. 6, no. 6, pp. 774–777, Dec. 2017.

[12] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[13] S. K. Mishra, D. Puthal, B. Sahoo, S. Sharma, Z. Xue, and A. Y. Zomaya, "Energy-efficient deployment of edge dataenters for mobile clouds in sustainable IoT," *IEEE Access*, vol. 6, pp. 56587–56597, 2018.

[14] 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.

[15] W. Zhang, Y. Wen, and H.-H. Chen, "Toward transcoding as a service: Energy-efficient offloading policy for green mobile cloud," *IEEE Netw.*, vol. 28, no. 6, pp. 67–73, Nov. 2014.

[16] X. Lyu, H. Tian, C. Sengul, and P. Zhang, "Multiuser joint task offloading and resource optimization in proximate clouds," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3435–3447, Apr. 2017.

[17] T. G. Rodrigues, K. Suto, H. Nishiyama, and N. Kato, "Hybrid method for minimizing service delay in edge cloud computing through VM migration and transmission power control," *IEEE Trans. Comput.*, vol. 66, no. 5, pp. 810–819, May 2017.

[18] H. Guo and J. Liu, "Collaborative computation offloading for multiaccess edge computing over fiber–wireless networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4514–4526, May 2018.

[19] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong, "Mobile data offloading: How much can WiFi deliver?" *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, pp. 536–550, Apr. 2013.

[20] 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.

[21] A. Ebrahimzadeh and M. Maier, "Distributed cooperative computation offloading in multi-access edge computing fiber–wireless networks," *Opt. Commun.*, vol. 452, pp. 130–139, Dec. 2019.

[22] C. Iwendi, M. Uddin, J. A. Ansere, P. Nkurunziza, J. H. Anajemba, and A. K. Bashir, "On detection of Sybil attack in large-scale VANETs using spider-monkey technique," *IEEE Access*, vol. 6, pp. 47258–47267, 2018, doi: 10.1109/ACCESS.2018.2864111.

[23] C. Iwendi, M. A. Alqarni, J. H. Anajemba, A. S. Alfakeeh, Z. Zhang, and A. K. Bashir, "Robust navigational control of a two-wheeled self-balancing robot in a sensed environment," *IEEE Access*, vol. 7, pp. 82337–82348, 2019, doi: 10.1109/ACCESS.2019.2923916.

[24] H. Guo, J. Liu, and J. Zhang, "Efficient computation offloading for multi-access edge computing in 5G HetNets," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.

[25] R. Li, N. Cao, M. Mao, Y. Chen, and Y. Hu, "Joint user association and energy offloading in downlink heterogeneous cellular networks," *Mobile Inf. Syst.*, vol. 2018, pp. 1–9, Aug. 2018.

[26] H. Guo, J. Zhang, J. Liu, and H. Zhang, "Energy-aware computation offloading and transmit power allocation in ultradense IoT networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4317–4329, Jun. 2019.

[27] J. H. Anajemba, Y. Tang, J. A. Ansere, and C. Iwendi, "Performance analysis of D2D energy efficient IoT networks with relay-assisted underlaying technique," in *Proc. 44th Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Washington, DC, USA, Oct. 2018, pp. 3864–3869.

[28] M. Li, F. R. Yu, P. Si, and T. Zhang, "Energy-efficient machine-to-machine (M2M) communications in virtualized cellular networks with mobile edge computing (MEC)," *IEEE Trans. Mobile Comput.*, vol. 18, no. 7, pp. 1541–1555, Jul. 2019.

[29] T. Zhu, T. Shi, J. Li, Z. Cai, and X. Zhou, "Task scheduling in deadline-aware mobile edge computing systems," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4854–4866, Jun. 2019.

[30] C. Iwendi, S. Khan, J. H. Anajemba, A. K. Bashir, and F. Noor, "Realizing an efficient IoMT-assisted patient diet recommendation system through machine learning model," *IEEE Access*, vol. 8, pp. 28462–28474, 2020, doi: 10.1109/ACCESS.2020.2968537.

[31] J. Zhang, X. Hu, Z. Ning, E. C.-H. Ngai, L. Zhou, J. Wei, J. Cheng, B. Hu, and V. C. M. Leung, "Joint resource allocation for latency-sensitive services over mobile edge computing networks with caching," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4283–4294, Jun. 2019.

JOSEPH HENRY ANAJEMBA (Member, IEEE) received the master's degree in information communication technology from the National Open University of Nigeria (NOUN), in 2016. He is currently pursuing the degree in computer science from the Federal Polytechnic Oko, Nigeria. He is also pursuing the Ph.D. degree in information and communication engineering with the College of Internet of Things, Department of Communication engineering, Hohai University, China (under the Chinese University FULL scholarship).

He has to his credit several articles and conference articles. His current research interests include cellular wireless communications, antenna and V2V technology and 5G cellular networks and security, and several other IoT related areas.

TANG YUE received the B.Eng. degree form Tianjin University, in 2004, China, the M.Eng. degree from the Shanghai Spaceflight Technology Institute, China, in 2008, and the Ph.D. degree from Nanyang Technological University, Singapore, in 2014. From September 2014 to July 2017, he worked in Hohai University as an Assistant Professor and from August 2017 to January 2020 as an Associate Professor. Since January 2020, he has been working with the Binjiang College, Nanjing University of Information Science and Technology, Wuxi, Jiangsu, China, as an Associate Professor. His current research interests include DBF radar systems, MIMO radar systems, random space array-based radar systems, array signal processing and antennas, and radio location.

CELESTINE IWENDI (Senior Member, IEEE) received the second master's degree in communication hardware and microsystem engineering from Uppsala University, Sweden, in 2008, ranked under 100 in the world University ranking, and the Ph.D. degree in electronics from the University of Aberdeen, U.K., in 2013.

He is currently an Associate Professor from Sweden. He is also a Senior Lecturer with the Bangor College China. He is also a Wireless Sensor Network Chief Evangelist, a Researcher, and a Designer. He is highly motivated researcher with a wireless sensor network security book, and more than 100 publications. He has strong teaching emphasis on communication, hands-on experience, willing-to-learn and 18 years technical expertise and currently teaches the Engineering team Project, Circuit Theory, Data Networks and Distributed Systems, and Control Systems. He has developed operational, maintenance, and testing procedures for electronic products, components, equipment, and systems; provided technical support and instruction to staff and customers.

Dr. Iwendi has been a Board Member of the IEEE Sweden Section, since 2017, and a Fellow of The Higher Education Academy, U.K., to add to his teaching and professional experiences. His research focuses on wireless sensor networks, security of things (SoT), machine learning, AI, communication controls, the Internet of Things (IoT), electromagnetic machines, 5G networks, and low power communication protocols. He is the Co-Chair of the special session on Wireless Sensor Networks: Hardware/Software Design aspects for Industry at the Prestigious International Conference of Industrial Technology ICIT. He is an Editor of the *International Journal of Engineering and Allied Disciplines*, 2015, a Newsletter Editor of the IEEE Sweden Section, from 2016 to 2018, an Editor-in-Chief of *Wireless Sensor Network Magazine*, 2009, a Committee Member of the International Advisory Panel, International Conference on Marine, Ocean and Environmental Sciences and Technologies (MAROCENET), from 2014 to 2016, an Editor-in-Chief of the *Journal of Wireless Sensor Network*, 2009, and an Advisory Board of the *International Journal of Innovative Computer Science and Engineering* (IJICSE), 2013.

MAMDOUH ALENEZI received the M.S. degree from DePaul University, in 2011, and the Ph.D. degree from North Dakota State University, in 2014. He is currently the Dean of educational services with Prince Sultan University. He has extensive experience in data mining and machine learning, where he applied several data mining techniques to solve several Software Engineering problems. He conducted several research areas and development of predictive models using machine learning to predict fault-prone classes, comprehend source code, and predict the appropriate developer to be assigned to a newly reported bug.

MOHIT MITTAL (Member, IEEE) received the B.Tech. and M.Tech. degrees in computer science and engineering from Guru Nanak Dev University, Amritsar, in 2010 and 2011, respectively, and the Ph.D. degree from GKV, Uttrakhand, India, in 2018. He is currently a Postdoctoral Researcher with the Department of Information Science and Technology, Kyoto Sangyo University, Kyoto, Japan. He has published more than 35 research articles in various international journals and conferences. His area of research is wireless sensor networks and artificial intelligence.