Orchestration of MEC Computation Jobs and Energy Consumption Challenges in 5G and Beyond

RASHA SAMIR, HADIA EL-HENNAWY, (Member, IEEE), AND HESHAM M. EL-BADAWY, (Senior Member, IEEE)

1Department of Electronics and Communications, Faculty of Engineering, Ain Shams University, Cairo 11517, Egypt
2Department of Network Planning, National Telecommunications Institute, Cairo 11768, Egypt
Corresponding author: Rasha Samir (g19083499@eng.asu.edu.eg)

ABSTRACT Mobile Edge Computing (MEC) technology philosophy inspires the next generation mobile networks to provide cloud computing capabilities in addition to a diverse range of Information Technology (IT) services with ultra-low latency and higher bandwidth at the edge. One of the most common challenges of 5G-MEC is the management and orchestration across all networks and infrastructure resources as well as end-to-end quality of experience. The decentralized architecture of MEC with independent and non-collaborative servers results in the situation of having underutilized servers with wasted energy. Moreover, the consequences of having highly utilized servers with highly consumed energy are not only the incapability to accommodate all the load of the computing jobs and the dramatic increase in the total OPEX cost, but it also creates some environmental problems. Orchestrating servers’ workload and control offloading the computation jobs is one of the technical advantages of MEC since it satisfies the increasing requirements of modern mobile applications while optimizing the energy consumption and cost. In this work, we consider cluster-based energy-aware offloading framework. The proposed work consists of dual-tier domain divided into clusters of Edge Servers $ES_i$. We have presented the results of our simulation as a proof of our concept that the formulated adaptive strategy to minimize the optimization problem calculation per cluster reduces the energy consumption and enhances the quality of experience while achieving the conservation of the related computing and storage resources cost.

INDEX TERMS MEC, IT, 5G, OPEX (operating expense), edge servers.

I. INTRODUCTION

The Information and Communication Technology (ICT) touches almost everything in the human life today, including education, commerce, agriculture, environment management etc. The highly increased demand on mobility services and smartphones used to access the cloud services and internet from anywhere at anytime securely has encouraged the mobile service providers to invest more and develop new mobile networks business models that translate the future challenges into solutions.

Fifth Generation (5G) mobile communication technology that introduced high data bit rates and ultra-low latency meets the challenging demands of future compute, storage, database, and other selected Information Technology (IT) services, while the sheer ubiquity of delay-sensitive and mission critical applications add more challenges into the 5G.

Mobile Edge Computing (MEC) moves the computing resources and storage capacity services from a centralized cloud to the edge of the mobile network in a partial or full process which analyzes and stores the data closer to the user with real-time processing and enhanced quality of latency-sensitive applications. MEC allows the users to deploy virtual machines, containers, managed databases and other IT infrastructure at edge locations; it also reduces the extra network hops to the internet and takes full advantage of the bandwidth and latency advancements of 5G to connect to an application from a mobile device.

The energy consumption of MEC edge server is influenced by multiple components, such as Central Processing Unit (CPU), memory, network interface card, etc. The most two parameters that affect the energy consumption are CPU
and network communication. The server performance is usually presented by its CPU utilization. Server CPU consumes energy even when the server is underutilized or in an idle state as CPU consumes high energy to run in a full utilization state and process complex computing work. Servers also consume high energy to transmit Protocol Data Units (PDUs) over the network in a high bit rate.

The collaboration between the MEC servers is playing a vital role in orchestrating the computational jobs and energy consumption as the MEC architecture of non-collaborative and independent server will result in the situation of having poorly managed environment with underutilized servers with inefficient use of its energy while other servers are overutilized by unpredicting end-user computation.

The distance between the end-user and server influences the quality of service; MEC is shortening the communications distance by bringing the servers into the edge of the mobile network so instead of sending all data to a central cloud for processing, the network edge analyzes, processes and stores the data with minimum latency.

The increasing number of the servers at the edge in order to provide high coverage and to minimize the application’s response time to the end-user requests has enormous advantages, however it results in huge energy consumption and cost.

There are multiple factors that affect the tasks offloading process such as wireless radio connection, end-system hardware configuration, end-user’s software configuration, and available bandwidth. Therefore, we have formulated an adaptive algorithm to control this sophisticated process and to support a wide range of use scenarios. In addition, we propose cluster-based energy-aware offloading framework to orchestrate servers’ workload and control offloading of the computation jobs to build more flexible, dynamic, programmable and centrally controlled telecommunications network. To the best of our knowledge, the proposed work will add more technical advantages into MEC which satisfies the modern mobile applications increasing requirements while optimizing the energy consumption and environmental impacts intelligently.

This paper contributions may be summarized as follows:

- We address the system model and introduce a solution of optimization problem in [1] using a proposed Iterative System Algorithm (ISA).
- We propose cluster-based energy-aware offloading framework that fully automates the collaboration between the edge servers and orchestrates the offloading of computation jobs efficiently in order to fully benefit from the total available energy as well as to minimize the cost.
- We compare the optimization problem size as a main factor affects the computational jobs, versus the number of nodes before and after the clustering framework.

The remainder of the paper is organized as follows: section II represents the related work, section III discusses the discussed system model, the proposed system description with system model parameters, section IV illustrates the proposed Simulation discussion and results, and finally the paper is concluded in section V.

II. RELATED WORK

The limited and distributed control of computation jobs, that may results in inefficient resource utilization problem, has motivated the work on optimizing the use of the resources and centrally orchestrated the distribution of the computation jobs. Offloading the computation jobs is one of the technical advantages of MEC. It achieves the modern mobile applications requirements and it optimizes the energy consumption.

For energy-aware MEC structures, optimization problems were presented in [1], [2]. In [1] an optimization problem was formulated to tackle predictions of harvested energy and to predict the cost of grid power needed to be purchased. Considering that offload may occur in regard to saving the power grid cost. However, no clear offloading strategy was discussed in addition to the presented work considers the edge servers power level without taking the actual CPU and buffer utilization into consideration. The discussed system model assumed fully-meshed adjacency sessions between nodes and that adds more overhead to the node’s CPU and memory (These issues are taken into consideration in our proposed work). [2] was scoped to two novel optimization problems that formulated to investigate reliable and seamless Network Function Virtualization (NFV) service provisioning. The author in [3] investigated the cost-sensitive service caching problem without resource sharing. [4] presented the study of the architectural models and features of industrial edge computing and virtualization. [5] proposed a reinforcement-learning-based method for offloading making decisions. [6] discussed the research achievements from different perspectives combining edge cloud collaboration and edge intelligence with edge computing security. In [7] a hardware prototype was implemented to provide extensive experiments on a deep neural networks (DNNs) to achieve minimum training time. [8] proposed a model for running application monitoring in an Internet of Thing (IoT) environment. [9], [10] introduced power saving problem on IoT devices. [11] proposed a hand-off model between a specific model named (AGCM) to reduce the waste of energy at both cloud and mobile edge.

It is always motivating to develop new networking concepts, optimize and address the increasing needs of the modern intensive latency-sensitive applications running on battery powered mobile terminals [12]–[14]. There are many challenges of 5G management and orchestration across MEC network infrastructure resources as well as end-to-end service configuration [15]–[18]. Other applications have been discussed to be deployed through the migration procedures in 4G-5G MEC based infrastructure as in [19], [20] and [27]. To satisfy these needs, many researchers discussed these challenges from different perspective.

Therefore, we propose the cluster-based energy-aware offloading framework. The contribution is not only limited to discuss the challenges that affected the MEC environment
Our proposed work conquered the observed challenges in [1] by dividing the domain of fully-meshed tier-1 ESs and tier-2 orchestrators into clusters. Each cluster consists of group of tier-1 ESs and tier-2 orchestrators as in fig. 2. Tier-1 ESs establish full mesh connectivity with the local cluster tier-2 orchestrators and other ESs. Tier-1 ESs share its utilization information with the local cluster tier-2 orchestrators. Also receive the updates from the orchestrators regarding the other ESs with available resources and the best path toward it. Tier-2 orchestrators establish full connectivity with the peer- ing clusters tier-2 orchestrators in order optimally exchange the whole domain availability and utilization updates.

Controlling and centralizing the information exchange only through the tier-2 orchestrators reduces the ESs CPU, memory and energy overhead by eliminating the control plane information exchange with other ESs. Tier-2 orchestrators use the learned information to build topological database, calculate the best offloading next hop for each ES and distribute that calculation results with local cluster ESs.

To ensure the high availability, another scenario is also proposed with redundant cluster orchestrators that provides tier-2 orchestrators failure protection. In this scenario, local tire-2 orchestrators will represent one active orchestrator at a time, while it maintains active/standby relationship with each others and synchronize their topological database. Tire-1 will exchange the updates with the active orchestrator only while it will exchange heartbeats only with the standby orchestrators of the local cluster as shown in fig. 3.

**B. PROPOSED SYSTEM MODEL PARAMETERS**

To facilitate the comparison in this work we used the same parameters, listed in table 1, as in [1] as follows; each cluster consists of \( \mathcal{N} \) of ESs with \( i \) nodes where each node has a battery with a maximum capacity \( B_{\text{max}} \). Time is divided in slotted manner with \( k \) time slot where \( k \) is considered the unit of time measured per hour. Each node needs computing energy \( c_i(k) \), while the energy drained \( d_i(k) \) for \( i \) node, and \( h_i(k) \) is considered as the harvested energy.
which is the sum of harvested energy part saved in battery $h_i^b(k)$, and for the computation jobs $h_i^c(k)$ as in (1).

$$h_i(k) = h_i^b(k) + h_i^c(k). \quad (1)$$

Moreover, the purchased energy $g_i(k)$ per each node is also considered the sum of amount for battery charging $g_i^b(k)$ while $g_i^c(k)$ is the energy consumption once charged for computation as in (2).

$$g_i(k) = g_i^b(k) + g_i^c(k). \quad (2)$$

(3) represents the needed energy per each node defined as $c_i(k)$ is related to the jobs $r_{ij}$ assigned to the node $i$ while $r_{ij}$ is the offloading job to node $j$. Also, $c_i(k)$ refers to the sum of the energy per node $i$ as in (4).

$$c_i(k) = \alpha(r_{ii} + \sum_{j \in \mathcal{N} \setminus i} r_{ji}) + \beta(\sum_{j \in \mathcal{N} \setminus i} r_{ij}) \quad (3)$$

The contribution coefficients $\alpha$ and $\beta$ as assumed in [1] where: $\alpha > 0$, and $\alpha > 0$, $\beta < 0$,

$$c_i(k) = g_i^c(k) + h_i^c(k) + d_i(k). \quad (4)$$

The remaining purchased energy or from harvested energy generated from the solar cell attached to $ES_i$ is stored in the battery. The $b_i(k)$ is initialized with zero saved power. At each time slot $k$, the available battery power as in (5) is considered as the sum of the available power in the battery from the previous time slot $(k-1)$ after subtracting the drained energy $d_i(k)$ for the same time slot $k$ and the superfluous energy from even the harvested or grid.

$$b_i(k) = \eta_i((b_i(k-1) - d_i(k)) + \mu_i(h_i^b(k) + g_i^b(k))). \quad (5)$$

\( \eta \) and \( \mu \) represent the battery coefficients, where $\eta$ is the percentage reminder of the battery usage. and $\mu$ represents the percentage reminder of the battery coming from even the harvested energy and/or power grid. These coefficients are tend to unity as assumed in [1]. (6) to (11) are the system boundaries as follows;

$$g_i^b(k), g_i^c(k), h_i^b(k), h_i^c(k), d_i(k), r_{ij} \geq 0, \quad \forall i, \forall j, \forall k. \quad (6)$$

$$b_i^\text{max} \geq b_i(k), \quad \forall i, \forall k. \quad (7)$$

The power drained from the battery $d_i(k)$ considers less than or equal the remaining power from the previous time slot as per (8):

$$d_i(k) \leq b_i(k-1), \quad \forall i, \forall k. \quad (8)$$

(9) represents the boundary of the harvested energy whether used for computation or saved in battery, could not exceed the total harvested energy $h_i^\text{max}$.

$$h_i^c(k) + h_i^b(k) \leq h_i^\text{max}, \quad \forall i, \forall k. \quad (9)$$

The sum of offloading jobs $r_{ij}(k)$ from node $i$ to node $j$ minus the number of offloaded jobs $r_{ji}(k)$ from node $j$ to node $i$ is limited to the number of local computed jobs $r_{ij}(k)$ to the same node $i$ at each time slot $k$ as per (10)

$$\sum_{j \in \mathcal{N} \setminus i} (r_{ij}(k) - r_{ji}(k)) \leq r_{ij}(k), \quad \forall i, \forall k. \quad (10)$$

The number of local processed jobs even received on the same node or offloaded from another nodes should be less than or equal to the maximum number of jobs $h_i^\text{max}$ to be processed per time slot $k$ as in (11)

$$r_{ii} + \sum_{j \in \mathcal{N} \setminus i} (r_{ij}(k) - r_{ji}(k)) \leq h_i^\text{max}(k), \quad \forall i, \forall k. \quad (11)$$

The optimization problem in (12) minimizing the equations from (1) to (11).

$$\psi = \text{PTC (Power Total Cost)} \quad \text{that formulated the power cost in the following equation:}$$

$$\psi = \min \left\{ \sum_{k=1}^{T} p(k) \sum_{i \in \mathcal{N}} g_i(k) \right\} \quad (12)$$

### IV. SIMULATION DISCUSSION AND RESULTS

To build the experimental setup, we have used computer-based MATLAB 2016 simulation on a computer with Intel i7 Quad-core 3.2 GHz CPU and 64 GB RAM to evaluate the proposed model performance. We implemented this environment by considering a group of $ES_i$ that use input power as

#### TABLE 1. List of symbols and acronyms.

| Symbol | Description |
|--------|-------------|
| $\mathcal{N}$ | set of Edge Servers |
| $ES$ | Edge System |
| $B_i^\text{max}$ | Battery max energy capacity [kWh] |
| $c_i(k)$ | communication and computation jobs needed energy |
| $h_i^b(k)$ | harvested energy |
| $g_i^b(k)$ | grid power |
| $r_{ij}$ | local computed jobs |
| $d_i^b(k)$ | drained power from battery |
| $r_{ij}$ | offloaded jobs to another j nodes |
| $b_i^b(k)$ | battery power |

FIGURE 3. Orchestrated cluster-based energy-aware offloading framework with high availability.
harvested energy, power battery and a power grid. The optimization problem in [1] is formed and tested over 360 hours. In our simulated environment, we have built similar setup over 360 hours however we have achieved nearly predictive energy output in the first 24 hours followed by consistent values. We focused on comparing our proposed model and the discussed model performance within the first 24 hours to evaluate the proposed ISA algorithm with focusing on the outcome orchestrated clusters approach.

A. SIMULATION DISCUSSION

In our work to measure and solve the optimization problem, we initialized the parameters for weighted set \( N \) of nodes where \( i \) represents any node and \( w \) is the prediction window size. The considered time horizon is 24 hours. Each node is establishing fully meshed sessions with all other nodes resulting in total number of sessions \( N \times (N-1)/2 \). The main challenge was how to minimize the purchased energy. The lab configuration of 10 nodes in a fully meshed topology is implemented in 170 configuration row. The extensive number of established sessions made it a sub-optimal solution as it is directly proportional to the number of nodes. Changing the flat topology into clusters with local orchestrators has minimized the optimization problem size as the configuration rows decreased to 22 rows in our lab test. The tier-1 nodes establish sessions with local cluster members only, which makes it more scalable solution. Although considering the high availability and failure protection required adding a secondary orchestrator to each cluster, it did not add high processing overhead. Moreover, increasing the number of nodes does not affect the optimization problem size as in flat topology.

B. SIMULATION RESULTS

Our work begins with validating the discussed work in [1] on different prediction window starting with the first prediction window \( w \) at \( k = 1 \) hour with increments of two. The discussed work consists of 5 constraints (first 5 equations) and 6 boundary. For each node \( i \) in time slot \( k \), these variables are solved by applying ISA into the optimization problem as mentioned. Table 2 presents the simulation applied parameters value.

In [1] the energy cost should be predicted 1 day ahead to estimate when is the best time to purchase energy. [1] assumed that each node \( i \) is able to offload to any of other \( j \) nodes in the fully meshed flat topology; the domain size is directly proportional with the optimization problem size. In our work, the orchestrators are fully aware of the whole domain members’ energy utilization. Each orchestrator establishes fully meshed control plan sessions with all other orchestrators in the local domain. That allows to exchange the utilization and available energy information and create a full picture of the whole domain resources utilization. The orchestrator will instruct the local cluster member to offload its excess computation jobs to the nearest available domain member. The following Algorithm 1 illustrates the non-linear optimization problem solution for the cluster-based energy-aware offloading Framework. In the algorithm, we created an array cycled with two for-loop one started from \( w = 1 \) with step 2 till \( w = 24 \) and the other loop for moving the window till reach 360 hours.

\[
\text{Algorithm 1: The Proposed Cluster-Based Energy-Aware Framework}
\]

1: Initialize Constant
2: \( W = 1 \)
3: for \( J = 1:360 \) do
4: \( \text{Reference Constraint function} \)
5: \( \text{Reference min function} \)
6: \( \text{call } F_{\text{mincon}} \)
7: \( \text{add } F_{\text{val}} \text{ to the sum} \)
8: \( \text{Reference the battery stored energy} \)
9: end for
10: \( \text{Save the purchased energy cost} \)
11: for \( W = 2:2:24 \) do
12: \( \text{for } J = 1:360 \) do
13: \( \text{Solve the optimization problem for the given duration} \)
14: \( \text{Divide this sum by the sum above to get the saving percentage} \)
15: end for
16: \( \text{Plot the results with respect to } W \)

In the following fig. 4, we evaluated the prediction of energy cost in percentage per time and compared the results of [1] with our proposed work results across different \( w \). ISA algorithm is implemented as a solution to the discussed non-linear optimization problem and the results are nearly the same after applying the ISA over \( w \) within the 24 hours.

With the increments of \( k \), moving from time slot to the next time slot, a new prediction calculation should start to solve optimization problem for the new time window parameters. We compares the running simulation with the discussed model within the first 24 hours however the simulation for \( T_{\text{sim}} = 360 \) hours to discuss the overall system performance.

Fig.5 represents the optimization problem size versus the number of nodes. These nodes are distributed in a fully meshed flat design without cluster formation. ISA algorithm is applied to solve the optimization problem. As shown, increasing the number of nodes increases the optimization

| Parameter | Value |
|-----------|-------|
| \( \alpha \) | 0.108 Wh/(job/h) |
| \( \beta \) | 0.036 Wh/(job/h) |
| \( \eta_l \) | 0.9999 |
| \( \mu_i \) | 0.9900 |
| \( T_{\text{sim}} \) | 360 h |
| \( R_{\text{max}} \) | 1.00 kWh |
| \( h_{(0)} \) | 0.00 Wh |
| \( R_{\text{max}} \) | 3330 job/h |

TABLE 2. Simulation parameters value.
problem size which consumes more CPU and memory resources.

In our lab simulation, we started with the assumption of 10 nodes in a fully meshed topology over different prediction window $w = 1, 12$ and $24$. We ran the optimization problem on different number of nodes from 10 to 100 nodes. As shown, increasing the number of nodes over different prediction $w$ enlarges the optimization problem size. Based on this observation, we noticed that with the increasing number of fully meshed nodes will also increase the optimization problem size that impacts the computational tasks and the amount of resources consumption.

In order to minimize this impact on the computation resources, we proposed the cluster formation model where each group of nodes is segregated in a cluster to segment the optimization problem into smaller optimization problems. The optimization problems will be addressed within any formed cluster over different $w$. To evaluate the cluster formation, we measured the problem size when the number of nodes is 16 nodes (where we assumed that the cluster could formed from the relation $[\text{cluster formed from} \sqrt{N}]$ where the minimum number of nodes to form a cluster is two nodes.) and measured the optimization problem size with the increase of $N$ up to 100 nodes.

Fig.6 shows the impact of cluster formation when tested with different number of nodes. However we started with 10 nodes (we set the node 16 is our reference just to facilitate the comparison with and without forming the cluster).

---

**TABLE 3.** The estimated saving percentage of processing the discussed optimization problem.

| $w$ value, $N=64$ | Size without Clustering | Size with Clustering | Saving percentage |
|-------------------|-------------------------|----------------------|-------------------|
| $w = 1$           | 1.0944                  | 0.584                | 48.67%            |
| $w = 12$          | 947.2                   | 550.4                | 43.1%             |
| $w = 24$          | 1.485 e+04              | 1.088 e+04           | 27.27%            |

Fig.7 illustrates a comparison between the optimization problem size with respect to the number of nodes at different $w$. We evaluated the optimization problem size versus the number of nodes with and without forming the clustering. As seen from the comparison, the processing done on solving the optimization problem is positively affected with cluster formation which in turn enhance the overall processing utilization.

To discuss the impact of forming the clusters, a cross section taken at estimated number of nodes $= 64$ nodes with respect to different $w$ as shown in Fig.8. We observed dividing the optimization problem size into smaller problems to be solved in each cluster which minimize the computations and processing used to solve the main optimization problem and that enhancing the overall system performance.

The following table 3 represents the estimated saving percentage of processing the discussed optimization problem.
As per the results in Table 3, the optimization problem size dramatically decreased at different w. As shown the saving percentage stating from around 12% (at the randomly selected $N = 64$) to reach around 70% saving in the processing. Forming the clusters minimizes the optimization problem size which in turn minimizing the overall processing and the overall energy consumed.

V. CONCLUSION

The proposed MEC cluster-based energy-aware offloading framework that consists of dual tier domain divided into clusters of ESs and central orchestrators provides more efficient 5G-MEC workload orchestration, energy consumption and cost management. It is designed to operate efficiently in a highly scalable domain of computation resources as well as large workloads. We address the system model and introduce a solution for the discussed non-linear optimization problem using the proposed Iterative System Algorithm (ISA). Our solution considers sourcing the power from energy harvesting system, energy batteries in addition to the power grid. The simulation outcomes illustrate the positive impact of cluster formation and intelligent orchestration layer on minimizing the optimization problem size while satisfying the environmental concerns. The proposed model demonstrates a considerable enhancements that lead to save the overall computations and processing with percent more than 50% by orchestrating the computation jobs within the 5G(and beyond)-MEC architecture in cluster-based model.

REFERENCES

[1] D. Cecchinato, M. Bemo, F. Esposito, and M. Rossi, “Allocation of computing tasks in distributed MEC servers co-powered by renewable sources and the power grid,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., May 2020, pp. 8971–8975.

[2] Y. Ma, W. Liang, J. Li, X. Jia, and S. Guo, “Mobility-aware and delay-sensitive service provisioning in mobile edge-cloud networks,” IEEE Trans. Mobile Comput., vol. 21, no. 1, pp. 196–210, Jan. 2020.

[3] Z. Xu, “Collaborate or separate? Distributed service caching in mobile edge clouds,” in Proc. IEEE Conf. Comput. Commun., Jul. 2020, pp. 2066–2075.

[4] A. Aziz, O. Schelén, and U. Bodin, “A study on industrial IoT for the mining industry: Synthesized architecture and open research directions,” IoT, vol. 1, no. 2, pp. 529–530, Dec. 2020.

[5] T. Alfakh, M. M. Hassan, A. Gumaei, C. Savaglio, and G. Fortino, “Task offloading and resource allocation for mobile edge computing by deep reinforcement learning based on SARSA,” IEEE Access, vol. 8, pp. 54074–54084, 2020.

[6] H. Zeyu, X. Geming, W. Zhaochang, and Y. Sen, “Survey on edge computing security,” in Proc. Int. Conf. Big Data, Artif. Intell. Internet Things Eng. (ICBAIE), 2020, pp. 96–105.

[7] D. Liu, X. Chen, Z. Zhou, and Q. Ling, “HierTrain: Fast hierarchical edge AI learning with hybrid parallelism in mobile-edge-cloud computing,” IEEE Open J. Commun. Syst., vol. 1, pp. 634–645, 2020.

[8] C. X. Mavromoustakis, G. Mastorakis, and J. M. Batalla, “A mobile edge computing model enabling efficient computation offload-aware energy conservation,” IEEE Access, vol. 7, pp. 102295–102303, 2019.

[9] W. Kuo and Y. Wang, “An energy-saving edge computing and transmission scheme for IoT mobile devices,” in Proc. 8th Global Conf. Consum. Electron. (GCCE), 2019, pp. 1–2.

[10] I. Sarrigiannis, K. Ramantas, E. Kartsaiki, P.-V. Mekikis, and A. Antonopoulos, and C. Verikoukis, “Online VNF lifecycle management in an MEC-enabled 5G IoT architecture,” IEEE Internet Things J., vol. 7, no. 5, pp. 4183–4194, May 2020.

[11] A. Ismail, “Congestion-aware and energy-efficient MEC model with low latency for 5G,” in Proc. 7th Int. Japan-Africa Conf. Electron., Commun., Comput. (JAC-ECC), 2019, pp. 156–159.

[12] W. Chen and L. Han, “Time-efficient task caching strategy for multi-server mobile edge cloud computing,” in Proc. 21st Int. Conf. High Perform. Comput. Commun., 2019, pp. 1429–1436.

[13] B. Priya, R. Sri, A. Nimmagadda, and K. Garudkar, “Mobile edge communication an overview of MEC in 5G,” in Proc. 5th Int. Conf. Adv. Comput. Commun. Syst. (ICACS), 2019, pp. 271–276.

[14] K. Tocó, “Distributed orchestration framework using mobile edge devices,” in Proc. 3rd Int. Conf. Fog Edge Comput. (ICFEC), 2019, pp. 1–10.

[15] Z. Wang and Y. Cai, “Management optimization of mobile edge computing (MEC) in 5G networks,” in Proc. Int. Conf. Comput. Commun. Workshops (ICC Workshops), 2019, pp. 1–6.

[16] J. Wu, “Edge-cloud collaborative computation offloading model based on improved partial swarm optimization in MEC,” in Proc. 25th Int. Conf. Parallel Distrib. Syst. (ICPADS), 2019, pp. 959–962.

[17] M. P. Anastasopoulos, A. Tzanakaki, and D. Simeonidou, “Scalable service chaining in MEC-assisted 5G networks,” J. Lightw. Technol., vol. 37, no. 16, pp. 4115–4124, Aug. 15, 2019.

[18] R. Srinivasa, “Minimizing latency for 5G multimedia and V2X applications using mobile edge computing,” in Proc. 2nd Int. Conf. Intell. Commun. Comput. Techn. (iCCT), 2019, pp. 213–217.

[19] A. Hairuman, A. Zahra, G. Kusuma, and D. F. Murad, “MEC deployment with distributed cloud in 4G network for 5G success,” in Proc. 6th Int. Conf. Inf. Technol., Comput. Electr. Eng. (ICITACEE), 2019, pp. 1–6.

[20] S. Lee, S. Lee, and M. Shin, “Low cost MEC server placement and association in 5G networks,” in Proc. Int. Conf. Inf. Technol. Converg. (ICTC), 2019, pp. 879–882.

[21] J. Luo, F. R. Yu, Q. Chen, and L. Tang, “Adaptive video streaming with edge caching and video transcoding over software-defined mobile networks: A deep reinforcement learning approach,” IEEE Trans. Wireless Commun., vol. 19, no. 3, pp. 1577–1592, Mar. 2020.

[22] M. Munir, A. Shirajum, F. Sarder, and C. Hong, “Artificial intelligence-based service aggregation for mobile-agent in edge computing,” in Proc. 20th Asia–Pacific Netw. Oper. Manage. Symp. (APNOMS), 2019, pp. 1–6.

[23] B. Dab, N. Aitsaadi, and R. Langar, “A novel joint offloading and resource allocation scheme for mobile edge computing,” in Proc. 16th IEEE Annua. Conf. Consumer Electron. (CCNC), 2019, pp. 1–2.

[24] H. Jeong, “Lightweight offloading system for mobile edge computing,” in Proc. Int. Conf. Pervasive Comput. Commun. Workshops (PervCom Workshops), 2019, pp. 451–452.

[25] C. Wu, Q. Peng, Y. Xia, and J. Lee, “Mobility-aware tasks offloading in mobile edge computing environment,” in Proc. 7th Int. Symp. Comput. Commun. Netw. (CANDAR), 2019, pp. 204–210.

[26] L. Dong and R. Li, “Distributed mechanism for computation offloading task routing in mobile edge cloud network,” in Proc. Int. Conf. Comput. Commun. Netw. (ICCCN), 2019, pp. 630–636.
RASHA SAMIR received the B.Sc. degree in electronics and communications engineering from Benha University, Egypt, in 2005, and the M.Sc. degree in electronic and communication engineering from the Electronics and Communications Department, Faculty of Engineering, Arab Academy for Science and Technology and Maritime Transport (AASTMT), in 2018. She is currently pursuing the Ph.D. degree with the Faculty of Engineering, Ain Shams University, Egypt. She joined Global Knowledge world’s largest IT and Business skills training multinational company, as a Cisco Systems Instructor, early in 2007. She delivered Cisco Systems technical courses in different fields “routing & switching, service provider, IPv6, network design and management” as well as providing consultancy services. She moved to the National Telecommunication Institute (NTI), in 2010, as a Cisco Academy Instructor and a Program Manager managing and supporting 45 Cisco academies in Egypt and MEA region. She got certified CCIE # 64658, CCNP, CCNA Security, CCNA Cybersecurity, CCIP, CCJA, CCSI ID# 31826, HCIA, AWS Certified Cloud Practitioner, AWS Certified Solutions Architect–Associate, JNCIA-Junos, Juniper Networks Certified Internet Associate–Junos, and JNCIS-ENT Juniper Networks Certified Internet Specialist.

HADIA EL-HENNAWY (Member, IEEE) received the B.Sc. and M.Sc. degrees from Ain Shams University, Cairo, Egypt, in 1972 and 1976, respectively, and the Ph.D. degree from the Technische Universität Braunschweig, Germany, in 1982. In 1982, she returned to Egypt and joined the Electronics and Communications Engineering Department, Ain Shams University, as an Assistant Professor. She was nominated as an Associate Professor, in 1987, and then a Professor, in 1992. In 2004, she was appointed as the Vice-Dean for graduate study and research. In 2005, she was appointed as the Dean of the Faculty of Engineering, Ain Shams University. She has been the Head of the Microwave Research Laboratory, since 1982. She is deeply involved in the Egyptian branch activities. She has focused her research on microwave circuit design, antennas, microwave communication, and recently wireless communication. She has published more than 100 journal articles and conference papers and supervised more than 50 Ph.D. and M.Sc. students. She is a member of the Industrial Communication Committee of the National Telecommunication Regulatory Authority (NTRA), the Educational Engineering Committee of the Ministry of Higher Education, and the Space Technology Committee of the Academy of Scientific Research. She was the Editor-in-Chief of the Faculty of Engineering, Ain Shams University, Scientific Bulletin, from August 2004 to August 2005.

HESHAM M. EL-BADAWY (Senior Member, IEEE) is currently a Full Professor and the Head of the Networks Planning Department, National Telecommunication Institute, Ministry of Communication and Information Technology, Cairo, Egypt. He has authored over 90 journals and conference papers supervising over 65 Ph.D. and M.Sc. students. His research interests include networking, performance evaluation techniques, teletraffic modeling, MIMO systems, and WiMAX networks, ATM networks, digital communication, and recently broadband wireless communication systems, such as WCDMA/HSPA and LTE. He is also a Recognized Instructor and an Academia Member with the International Telecommunication Union. Since May 2014, he has been elevated to be the IEEE Senior Member in the Vehicular Technology Society and the Communications Society.

* * *