IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 28, 2021, accepted June 8, 2021, date of publication June 28, 2021, date of current version July 6, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3092819

Cloudlet Selection in Cache-Enabled Fog Networks for Latency Sensitive IoT Applications

RABEEA BASIR[1], SAAD QAISAR[1,2], (Senior Member, IEEE), MUDASSAR ALI[1,3], (Member, IEEE), AND MUHAMMAD NAEEM[4], (Senior Member, IEEE)

[1]School of Electrical Engineering and Computer Science, National University of Science and Technology, Islamabad 44000, Pakistan
[2]Department of Electrical and Electronic Engineering, University of Jeddah, Jeddah 23218, Saudi Arabia
[3]Department of Telecommunication Engineering, University of Engineering and Technology, Taxila 47050, Pakistan
[4]Department of Electrical Engineering, COMSATS University Islamabad, Wah Campus, Wah Cantt 47040, Pakistan

Corresponding author: Rabeea Basir (rbasir.dphd17seecs@seecs.edu.pk)

This work did not involve human subjects or animals in its research.

ABSTRACT Over the coming years, the foresighted enormous increase in smart devices supporting Internet-of-Things (IoT) applications demand novelty in network design. A promising solution to the ever-increasing low-latency requirement of IoT applications is the development of fog network architecture. However, the presence of an enormous number of smart devices in fog networks affects the performance of the network. To harvest the benefits of fog networking necessitates finding optimal cloudlet selection strategies. This article formulates a mixed-integer non-linear programming (MINLP) problem that has the objective of latency minimization. An exhaustive search on our cache-enabled (CE) fog architecture cannot be applied because of the problem's combinatorial and NP-hard nature. Similarly, the genetic algorithm (GA) cannot be used to find the solution because of the calculation of the number of generations. The increase in the number of IoT and fog nodes increases the solution search space, hence an Outer Approximation Algorithm (OAA) is proposed to arrive at the solution. Low complexity, convergence, and effectiveness of the proposed algorithm ensures the ϵ-optimal solution = 10^{-3}, obtained through standard problem solvers.

INDEX TERMS Cache, cloudlet selection, fog networks, Internet of Things (IoT), MINLP.

I. INTRODUCTION

Exponential industrial development in all fields expected to result in an enormous number of smart devices deployment. Prediction is that more than 50 billion smart devices will be deployed which results in the production of 13 times more data than non-smart devices, by 2020 [1]. Deployment of IoT nodes supporting latency-sensitive applications is challenging because of their stringent ultra-reliable, low-latency communication (URLLC) and different quality-of-service (QoS) requirements. This scalability and low-latency requirement of IoT nodes results in the inefficiency of cloud computing. Computation/processing of many different applications at the same time on the cloud server results in added delays. Propagation delays are also added because of cloud placement at the further place. Because of these latency-sensitive applications running on the user's end, an extension of cloud computing named as fog computing has emerged. It will act as a key enabling technology to support the future 5th Generation (5G) IoT applications [2], [3]. For the fourth industrial revolution, fog computing will be an enabling technology because it supports mobility, location-awareness, heterogeneous IoT smart devices, realtime and secure communication [3], [4], [5]. Fog computing is a decentralized solution for latency-sensitive applications, which involves the computation and storage resources of data near the edge of the network, near to the IoT nodes. Cloudlet nodes are an extension of a cloud server placed close to the IoT nodes. In comparison, cloudlet nodes have limited data storage capacity and computation/processing capability than the cloud server.

To access, process, and forward information between cloudlet and cloud server, the backhaul link's burden increases. This burden results in propagation and processing delays. Similarly, throughput increases due to sharing a single spectrum and heavy traffic burden between cloudlet node and IoT devices. This results in the fronthaul link's burden, resulting in a transmission delay. Because of the burden limitations of these links, the concept of caching at cloudlet nodes has been proposed. The presence of cache on cloudlet

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

nodes in fog networks will result in the minimization of the overall network's latency [6]. As the requested data available in close vicinity of users at the cloudlet node, there will be no computation and propagation delays. Cache-enabled (CE) cloudlet nodes in fog network will support latency-sensitive IoT applications.

Fog computing as an enabler for industry 4.0 applications has some serious implementation challenges of cache-placement, resource management, network deployment, network modeling, energy consumption, and user association. For minimizing the overall network latency, efficient cloudlet-selection and user-association techniques for CE-fog networks are required. Recently researchers have done some work on resource management aiming different objective functions of maximizing energy efficiency, minimizing latency, and maximizing network utility in fog networks. Section II provides a detailed literature survey on existing relevant work on the cloudlet selection, mentioning their limitations in Table 1. System model and the problem formulation is explained in the Section III. Section IV discusses the proposed algorithm followed by algorithm's convergence proof and its complexity analysis. Numerical results that are observed under the proposed algorithm are presented in Section V. Finally, we conclude this work in the last Section VI.

II. LITERATURE REVIEW

Globally, many researchers and industrial assemblies have noticed the need for new research techniques for real-time communication supporting IoT applications. In the upcoming future, new techniques for the deployment of networks, management of resources, and energy will be under consideration to minimize the latency for IoT applications. The fog computing paradigm causes the upcoming shift in network architecture that supports reliable, economical, and real-time communication for future IoT applications. Cacheable Small Cell Base Stations (SCBs) are widely deployed in small cell networks (SCNs) for managing traffic load across the network. Network Utility Aware (NUA) load balancing scheme was proposed by authors in [7]. Offloading, buffering, and resource allocation are three parallel algorithms proposed by authors to propose an optimal resource allocation on the basis of throughput, balancing, and delay [8]. A graph-based caching approach with less complexity was proposed in comparison with the brute-force approach in [9]. The proposed approach gives remarkable improvements in terms of traffic offloading parameters. Authors have formulated a 0-1 integer programming problem for fog radio access network (F-RAN) under constraints of maximum distance and maximum traffic load threshold.

A new bio-inspired optimization algorithm named as Bees Life Algorithm (BLA) was proposed by authors in [10]. Their objective function is to distribute tasks among fog nodes optimally. They proposed better results after studying an optimal trade-off between execution time and memory. In [11], authors have proposed an optimal resource allocation scheme among fog nodes (FNs) and data service operators (DSOs). First Stackelberg game was proposed to analyze the pricing problem and then a many-to-many matching algorithm is used for FN-DSS (data service subscribers) pricing problem. For the joint optimization problem of cloudlet selection and bandwidth allocation, an iterative algorithm was proposed by authors in [12] for a triple-stage Stackelberg game. To achieve maximum utility by minimizing latency during association authors have proposed the Boltzmann-Gibbs learning algorithm [13]. User clustering was done based on the user's request and using the proposed algorithm was used to choose a cache-enabled cloudlet. Similarity-Aware popularity-based Caching (SAPoC) is proposed in [14] to improve the performance of the network in terms of the cache hit ratio and energy consumption. The arrival and departure of mobile devices are considered for the wireless edge computing network. If the requested file is already stored at the edge of the network, the cache hit ratio is triggered, otherwise, the request is sent to the cloud for processing. The freshness of stored content, similarity, and frequency of request is determined by content popularity prediction. The performance of the proposed algorithm is compared with other caching algorithms, in terms of energy consumption and cache hit ratio. In [15], authors have proposed a solution for new emerging Mobile Edge Computing (MEC) systems using the GNU Linear Program kit (GPLK) for the formulated problem. User association and admission control were studied under delay and cloudlet's storage capacity. A tradeoff between queue delay and power consumption was investigated for the proposed F-RAN scenario. The stochastic-based mixed-integer, joint optimization on mode selection and resource allocation problem is formulated. Authors have studied two reinforcement learning-based (RL) algorithms and solved the problem using the Lyapunov optimization [16].

For resource management, in [17] authors have proposed an energy-aware algorithm. and an evolutionary algorithm for less energy consumption respectively. A tradeoff between power consumption and delay was studied in [18]. The authors consider the power and delay constraints for F-RAN architecture. They proposed a delay-aware energy-efficient computation offloading scheme for minimizing the consumption of grid energy [19]. A power control algorithm and Greedy algorithm (GA) was studied with the aim of energy-efficient resource allocation in [20] and [21] respectively. To identify a suitable node for caching in a distributed way, a content-placement algorithm named as Content-Based Centrality (CBC) was proposed in [22] for information-centric fog computing. Authors have formulated a non-convex optimization problem aiming maximization of content at fog nodes subjected to buffer size threshold.

To maximize the overall delivery rate for cache-enabled F-RANs architecture, a convex approximation method was studied by authors in [23]. The joint optimization problem was formulated under fronthaul capacity, file size, and transmit power for user association to fog node. In [24],

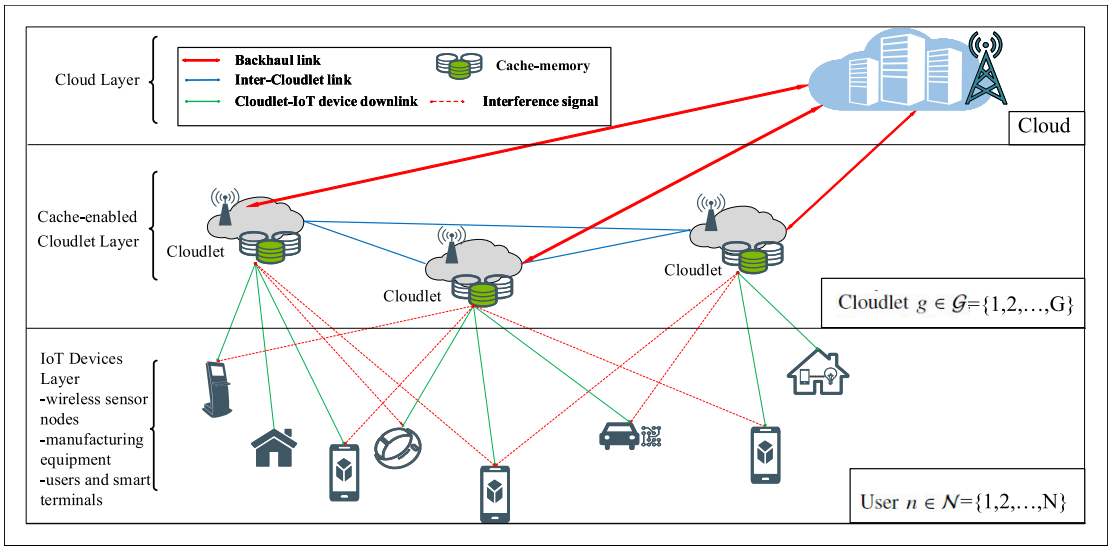

FIGURE 1. System Model - Downlink in fog network with IoT nodes.

two different (maximum data rate and minimum latency) optimization problems were studied. A distributed algorithm is proposed for solving the first joint problem of caching placement and association strategy. Optimal power allocation is derived in a closed-form expression for the second joint problem of mode selection and subchannel assignment. Resource management under storage capacity and bounded delay constraints to minimize latency were studied in [25] and [26]. The optimal workload allocation problem under optimal tradeoff between latency and power was studied in [27]. Authors have proposed two distributed algorithms supporting novel fog node cooperation strategy. A self-organized many-to-one matching game approach was used in [28], [29], and [30] to minimize the latency during cloudlet selection. Computational caching algorithm and Regret-Based algorithm were proposed by authors in [31] and [32] respectively, to a latency minimization problem under a storage size constraint. For load balancing, user association in [33], [34], [35], authors have studied cloud-fog integrated Industrial Internet of Things (CF-IIoT) network to get ultra-low latency. For this objective, they have proposed a real-coded genetic algorithm for constrained optimization problem (RCGA-CO) algorithm. They have formulated a min-max optimization problem for resource management. Table. I summarize the mentioned related work mentioning respective contributions.

A. CONTRIBUTIONS

The key motivation of our work is that none of the previous work considered joint resource selection, admission control, and power allocation scheme for a cache-enabled (CE)-fog network that minimizes the overall network latency. This work is the first attempt that provides a solution for the future upcoming industrial revolutionized era which will be composed of real-time low latency IoT applications. To the best of our knowledge and comparison of some previous work mentioned in Table 1, our contributions in this article are:

1) We propose a mathematical framework for latency minimization in the CE-fog network.
2) A joint optimization problem is presented, which is of mixed-integer non-linear programming type. The formulated problem jointly takes into account cloudlet selection, admission control, and power allocation parameters.
3) Due to the combinatorial and NP-hard nature of the problem, problem complexity will increase as the number of integer variables increases. In this article, with finite convergence, an ϵ-optimal solution is achieved using the outer approximation algorithm (OAA).
4) We also compared the performance of OAA algorithm with the genetic algorithm (GA) and the exhaustive search algorithm (ESA). The results show OAA gives better results than both.
5) Extensive simulations have been done to find ϵ-optimal solution using the low-complex OAA, which means that an optimal solution is just only $\epsilon = 10^{-3}$ away.

III. SYSTEM MODEL AND PROBLEM FORMULATION

A proposed fog network is shown in Fig. 1 in which there are g number of cloudlets or fog nodes such that $g \in \mathcal{G} = \{1, 2, \ldots, G\}$ and a single cloud is deployed. There is n number of IoT nodes in the network such that $n \in \mathcal{N} = \{1, 2, \ldots, N\}$, connected to cloudlets via a wireless link. To share load and information, cloudlet nodes have dedicated links among each other. To mitigate the interference between cloudlet-to-cloudlet link and IoT node-to-cloudlet node, the frequency bands are chosen that are different or non-overlapping. Dedicated backhaul links with high-speed capability are used for the connection of

TABLE 1. Previous work (U.A=User Association, P.A=Power Allocation, C.E=Cache Enabled, A.C=Admission Control, MILP=Mixed Integer Linear Programming, NLP=Non-Linear Programming, MINLP=Mixed Integer Non-Linear Programming).

Ref. No.	Objective	Constraints	Problem Type	Solution Technique	U.A	P.A	C.E	A.C
[7]	Min. Traffic load	Power utilization, min.latency, cache hit ratio	Optimization Problem: tradeoff between power utilization & latency	Network Utility Awareness (NUA) having 3 stages	✓		✓	
[8]	Optimal Resource Allocation	Capacity, Queue, Time	MINLP problem	Lyapunov drift, Three parallel algorithms	✓			
[9]	Max. Offload Gain	Max. Distance & Max. Load Threshold	Optimization (0-1 Integer Progamming) Problem	Graph-based Cooperative Caching Approach			✓	
[10]	Job Scheduling	Min. Cost	Convex Optimization Problem	Bees Life Algorithm (BLA) using Graph Theory	✓			
[11]	Max. Utility	Service (queuing + network) & delay cost	Non-Convex Optimization Problem	Stackelberg game and Many-to-many matching game	✓			
[12]	Max. Utility	Bandwidth & Power threshold	Non-cooperative Game Problem	Iterative Algorithm for Stackelberg equilibrium	✓	✓		✓
[13]	Max. Utility	Latency threshold	Coalition Game Theory Problem	Boltzmann- Gibbs Learning Algorithm	✓		✓	
[14]	Min. Energy Consumption	Power & capacity threshold		Similarity-Aware Popularity-based Caching (SAPoC) algorithm		✓	✓	
[15]	Min. Power Consumption	Delay, Capacity	MILP Problem	GNU Linear Programming Kit (GLPK) tool	✓			✓
[16]	Min. Power Consumption	Rate & power threshold, association	Non-Convex Optimization Problem	Lyapunov optimization, two RL-based approaches	✓	✓		
[17]	Min. Energy Consumption	Max. Workload, Processing rate threshold	Bin Packing Penalty Aware Convex Optimization Problem	Greedy techniques (Energy-Aware Algorithm)	✓			
[19]	Min. Grid Energy Consumption	Max. Delay, Max. Power, Max. Computational	Convex Optimization Problem	Offloading Decision Algorithm	✓	✓		✓
[20]	Max. Energy Efficiency	Association, Max. Power, QoS(min. rate), Caching (limited space)	Non-Convex Optimization Problem	Power Control Algorithm (alternative direction method of multipliers (ADMM))	✓	✓	✓	
[21]	Max. Energy Efficiency	Max. Power, Cache status, Max. Capacity	Non-Convex Optimization Problem	Fractional Programming + Greedy Algorithm	✓	✓	✓	
[22]	Max. Cache-memory size	Max. node buffer size	Non-Convex Optimization Problem	Content-Based Centrality (CBC) Algorithm			✓	
[23]	Max. Delivery Rate	Fronthaul capacity, file size, and transmit power constraint	Non-convex Optimization Problem	Convex Approximation Method	✓		✓	
[24]	Max. Throughput & Min. Latency	Power & capacity constraints	integer non-linear Optimization Problem	Distributed algorithm supported by McCormick envelopes and Lagrange partial relaxation method	✓	✓	✓	
[25]	Min. Average Delay	Association, QoS, caching capacity constraint	Non-convex NP-hard Optimization Problem	Two-Step Iterative Algorithm	✓		✓	
[26]	Min. Task Completion Time	Storage capacity, Bounded Delay	MINLP Problem	low-complexity three-stage Heuristic Algorithm	✓		✓	
[27]	Min. Response Time	Max. Power Efficiency	Dual-Decomposition Optimization Problem	subgradient method + ADMM-via Variable Splitting (ADMM-VS)	✓			
[28]	Min. Latency	Association & QoS constraints	Game Theory Problem	Many-to-one matching game approach	✓		✓	
[31]	Min. Latency	Memory size	Convex Optimization Problem	Computational Caching Algorithm	✓		✓	
[32]	Min. Latency	Storage capacity	Decoupling Convex Optimization Problem	Regret- Based Learning Algorithm	✓		✓	
[29]	Min. Computing Latency	Reliability, Max. Latency, Max. Storage capacity	Combinatorial nonConvex Optimization Problem	Matching Algorithm	✓		✓	
[30]	Min. Latency	Max. Workload, Max. Latency	MINLP Problem	Many-to-one Matching Algorithm	✓	✓		
[33]	Min. Latency	Communication + Computing + Processing constraints	NP-hard Problem	latency-driven cooperative Fog algorithm + cooperative task computing(CTC) algorithm	✓			
[34]	Min. Latency	Quality loss, Association, capacity threshold & service latency threshold	MILP NP-hard problem	Dynamic Task Allocation (DTA) algorithm	✓			
[35]	Min. Latency	Max. Delay threshold	Min-Max Optimization Problem	Real-Coded Genetic Algorithm for Constrained Optimization problem (RCGA-CO)	✓			
This work	Min. Latency	Max. Storage capacity, Max. Power threshold, QoS	MINLP Problem	Outer Approximation Algorithm (OAA)	✓	✓	✓	✓

TABLE 2. Notations.

Notation	Description
$\mathcal{G}, \mathcal{N}, \mathcal{C}$	Sets of cloudlets g, IoT nodes n, and content files c respectively.
g, n	Cloudlet node $g \in \mathcal{G}$ and user $n \in \mathcal{N}$
x_g^n	Binary indicator for connection mode of an IoT node n with a cloudlet node g.
y_g^n	Binary indicator representing the IoT node n selection for connection with a cloudlet node g.
z_g^n	Binary indicator related to cache availability at cloudlet node g.
d_c	Size of content file c.
P_g	Maximum transmit power of a cloudlet node g.
S_g	Maximum storage capacity of a cloudlet node g to store cached files c.
L	Maximum latency threshold for an IoT node n.
R	Minimum rate threshold required by an IoT node n.
p_g^n	Power allocated to an IoT node n when served by a cloudlet node g.
l_g^n	Latency experienced by an IoT node n when served by a cloudlet node g.
$l_{g,n}^T$	Transmission delay experienced by an IoT node n in downloading files from cloudlet node g.
$l_{g,n}^B$	Backhaul delay experienced by an IoT node n in downloading files from cloudlet node g.
$l_{g,n}^D$	Propagation delay due to backhaul links between cloudlet node g and cloud.
$l_{g,n}^P$	Processing time at cloudlet node g.
D_b	Distance between cloudlet node g and cloud.
v	Speed of light.
r_g^n	Data rate of an IoT node n when served by a cloudlet node g.
b_g^n	Assigned bandwidth by a cloudlet node g to an IoT node n.
G_o	Antenna Gain of cloudlet node g.
ζ	Zero mean Gaussian variable.
d_o	Antenna far field reference distance.
d_g	Distance of receiving IoT node n from cloudlet node g.
h_g^n	Channel gain when an IoT node n is connected to a cloudlet node g.
h_g^n	Rayleigh random variable when an IoT node n is connected to cloudlet node g.
N_o	Additive White Gaussian Noise.

cloudlets with the cloud server. The same frequency band is used for communication between cloudlets and IoT nodes, hence, there is interference experienced by IoT nodes from other close vicinity cloudlets. Each cloudlet g with a limited capacity of S_g bits is equipped with a set of data files given as $c \in \mathcal{C} = \{1, 2, \ldots, C\}$. IoT node n requests a file content c with the size of d_c bits, this request arrival rate has a Poisson distribution. This distribution has a mean of $\lambda_{n,c}$ where a higher mean arrival rate of a file shows that it is more popular in the system. If requested file c is available at the cloudlet's cache, it is directly delivered to the node, otherwise, the request will be handled by the cloud itself. This cloud request will increase in latency and overall system cost.

At a particular time, a cloudlet can serve multiple IoT nodes but a node can only be connected to only one cloudlet g. Let an IoT node $n \in \mathcal{N}$ be present in the coverage of multiple cloudlets and intend to download some files. Let $x_g^n = \{0, 1\}\ \forall n \in \mathcal{N},\ g \in \mathcal{G}$ be the binary indicator representing association of node with a cloudlet. It has value 1 when there is a connection between the node and a cloudlet, otherwise its value will be 0. Let $y_g^n = \{0, 1\}\ \forall n \in \mathcal{N},\ g \in \mathcal{G}$ be the binary indicator, which decides whether a node is admissible for connection or not, such that, $x_g^n \leq y_g^n$. IoT node admission will be done based on QoS requirements, given by a cloudlet to a user node. Variable y_g^n, represents admission control on cloudlet nodes, while x_g^n is a node association variable which ensures that a user node will be transmitting data with only one selected cloudlet. If an IoT node is admissible on a particular cloudlet node it has value 1, otherwise, if rate and latency constraints are not satisfied its value will be 0. Let $z_g^n = \{0, 1\}\ \forall n \in \mathcal{N},\ g \in \mathcal{G}$ be the binary indicator that shows the availability of requested file c in the cache memory of a cloudlet g. The proposed system model is shown in Fig. 1. We define total communication latency experienced by node n to get the requested content c from cloudlet g as l_g^n that can be stated as:

$$l_g^n = l_{g,n}^T + l_{g,n}^B, \tag{1}$$

where $l_{g,n}^T$ and $l_{g,n}^B$ are wireless transmission delay and backhaul delay, respectively. These are the delays experienced by an IoT node n after requesting files from the associated cloudlet node g. During transmission of a file c with size d_c from cloudlet g to IoT node n, the transmission delay of the wireless link can be calculated as:

$$l_{g,n}^T = \frac{d_c}{r_g^n} \tag{2}$$

In practice, traffic load and average link distance are related to backhaul delay, which is greater than the value of the transmission delay. $l_{g,n}^B$ is a combination of fog processing delay $l_{g,n}^P$ and propagation delay $l_{g,n}^D$. Here, for simplicity, we assume that processing delay for all fog nodes has a fixed value. If the IoT node received the requested files from the CE-associated cloudlet node, there will be no backhaul delay. However, if the requested file is not available in the cache memory of the cloudlet, the IoT node will experience both the delays as mentioned in Eq. (1). First, there will be a transmission delay and then there will be a backhaul delay, as the cloudlet has to fetch the requested files from the cloud via backhaul links. These delays will cause an overall increase in latency experienced by an IoT node. In Eq. (1), the backhaul delay $l_{g,n}^B$ experienced in fetching files from the cloud can be calculated as [25]:

$$l_{g,n}^B = \left(1 - z_g^n\right)\left(l_{g,n}^P + l_{g,n}^D\right) \tag{3}$$

The propagation delay $l_{g,n}^D$ is dependent on the D_b, which is the distance between the fog node and cloud. We have assumed that fog nodes are placed at different distances from cloud causes different propagation delays, which is given as $\frac{D_b}{v}$. Using Eq. (2) and Eq. (3), Eq.(1) can be updated as follows:

$$l_g^n = \frac{d_c}{r_g^n} + \left(1 - z_g^n\right)\left(l_{g,n}^P + l_{g,n}^D\right), \tag{4}$$

where r_g^n is the downlink (DL) transmission rate that is given by Shannon's capacity formula as:

$$r_g^n = b_g^n log_2\left(1 + SINR_{DL,g}^n\right), \tag{5}$$

here b_g^n is the assigned bandwidth by cloudlet to the node and $SINR_{DL,g}^n$ is the signal-to-interference plus noise ratio experienced by IoT node given as $SINR_{DL,g}^n = x_g^n \frac{p_g^n h_g^n}{\sum_{g' \neq g \in \mathcal{G}} p_g^{'n} h_g^{'n} + N_o}$; p_g^n and h_g^n are the assigned power and channel gain, respectively. While power and channel gain received by other interfering IoT nodes are given as $p_g^{'n}$ and $h_g^{'n}$, respectively. Channel gain is defined as $h_g^n = \bar{h}_g^n \zeta G_o \left(\frac{d_o}{d_g}\right)^{\alpha}$, where other parameters are antenna gain (G_o), log-normal shadowing ($\zeta 10^{\zeta/2}$), the distance between cloudlet and IoT node (d_g), the antenna far-field reference distance (d_o), path loss exponent (α), Rayleigh random variable ($\bar{h}_g^n$) and ζ is given as the zero-mean Gaussian random variable with standard deviation σ [36]. A summary of the symbol notation used in this article is given in Table 2.

For downlink transmission under cache-storage capacity, power allocation and QoS constraints, the formulated problem of joint cloudlet selection and latency minimization for CE-fog network, with objective function $\mathcal{J}$, can be mathematically stated as:

$$\mathcal{J}_{(x,y,z,p)} = \min \sum_{g \in \mathcal{G}} \sum_{n \in \mathcal{N}} x_g^n l_g^n$$

subject to constraints C1 to C9:

$$\begin{aligned}
&C1: \sum_{g \in \mathcal{G}} x_g^n \leq 1; \quad \forall n \in \mathcal{N} \\
&C2: r_g^n \geq y_g^n \frac{d_c}{L} \quad \forall n \in \mathcal{N} \\
&\textit{where} \\
&x_g^n \leq y_g^n; \quad \forall n \in \mathcal{N}, g \in \mathcal{G} \\
&C3: \sum_{n \in \mathcal{N}} p_g^n \leq P_g; \quad \forall g \in \mathcal{G} \\
&C4: p_g^n \leq y_g^n P_g; \quad \forall n \in \mathcal{N}, g \in \mathcal{G} \\
&C5: \sum_{n \in \mathcal{N}} z_g^n d_c \leq S_g; \quad \forall g \in \mathcal{G} \\
&C6: l_g^n \leq y_g^n L; \quad \forall n \in \mathcal{N} \\
&C7: r_g^n \geq y_g^n R; \quad \forall n \in \mathcal{N} \\
&C8: p_g^n \geq 0; \quad \forall n \in \mathcal{N} \\
&C9: x_g^n, z_g^n \in \{0, 1\}; \quad \forall n \in \mathcal{N}, g \in \mathcal{G}
\end{aligned} \tag{6}$$

At a time, user association constraint as C1 ensures that any node n can be connected to only one cloudlet. QoS constraint as C2 ensuring the user admission to a cloudlet that whether the user gets services from the associated cloudlet or not, based on minimum data rate and latency. C3 is the maximum power constraint that power assigned to all connected nodes must not exceed the maximum transmit power of the cloudlet (P_g). If the node is not connected to any cloudlet its power will not be considered, this is given in C4. C5 is the file caching constraint in terms of the limited storage capacity of a cloudlet (S_g). The total data sent by all users associated with a particular cloudlet must not increase the total storage capacity. Latency constraint as C6 ensures that latency must not exceed the threshold upper bound latency value (L). Constraint C7 is the minimum data rate threshold constraint, ensuring that the associated cloudlet provides a minimum data rate (R) to its admitted user node. Constraint C8 ensures that the power of a particular user that is connected to a single cloudlet must be greater than 0. C9 is the constraint that limits the value of node association and cache placement indicator to binary values of 0 and 1.

The formulated problem falls in a class of mixed-integer non-linear programming (MINLP) problem that is NP-hard. It is impossible to find an optimal solution in polynomial time because of the NP-hard nature. As the number of IoT and cloudlet nodes increases, the search space to find a solution increases exponentially. For a global optimal solution, an exhaustive search algorithm (ESA) can be used, but the search on binary variables results in high complexity. It gives search space having an order of $2^{|n|}$, which means there are $2^{|n|}$ optimization problems that need to be solved. Because of the exponential increase in search space, ESA cannot be applied to find a solution. Therefore, we propose the outer approximation algorithm (OAA), which requires relatively low computations to find a near-optimal solution [37]. In the literature, there are some other algorithms to solve the MINLP problem used in literature namely the branch and reduce (BR) algorithm [38], and the method by Lawler and Bell [39].

IV. PROPOSED TECHNIQUE

The Eq. (6) is very complex as it is a combination of binary, continuous, and integer variables. This combination makes a big challenge to solve this problem. With the increase in the number of IoT and cloudlet nodes, many variables need handling which makes it NP-hard. To solve the non-linearity and integrality of MINLP, OAA is applied, which works with the convergence of upper and lower bounds. MINLP is solved after decomposition in the primal problem (NLP) and master problem (MILP). An upper bound is achieved after fixing the binary variables (x_g^n, y_g^n, z_g^n) which is further used to find a lower bound. The detailed implementation of OAA for the proposed latency minimization problem is given in the next section.

A. OAA DESCRIPTION

In Eq. (6), let the constraints from C1 to C9 that are subject to objective function $\mathcal{J}$, be denoted by a set as F_{c1-c9}, $P = \{p_g^n\}$ as a set of continuous variables and $\Omega = \varphi \cup P$ as set of discrete variables. It is observed that the following four propositions are satisfied by the formulated problem:

1) P is a compact set of variables having properties of non-emptiness and convex.
2) $\mathcal{J}$ and F_{c1-c9} are continually differentiable for making P convex.
3) The NL problem can be obtained after fixing the value of Ω.

4) After fixing the values of all discrete variables, the solution of continuous primal problem satisfies all the constraints.

OAA will converge in an infinite number of iterations with convergence capability ϵ [37]. OAA follows a non-decreasing lower bound and a non-increasing upper bound. After solving the primal-subproblem, the *upper-bound* sequence is calculated, whereas the *lower-bound* sequence is calculated after solving the master problem. According to proposition 4, the primal problem is formed using fixed values of Ω. The primal problem with Ω^n integer values, at the n^{th} iteration of the algorithm, can be written as:

$$\begin{aligned} &\min_{P} \quad -\mathcal{J}(\Omega^n, P) \\ &\text{subject to}: \quad F_{c1-c9}(\Omega^n, P) \leq 0 \end{aligned} \tag{7}$$

The master problem is formed by the solution (P^n) of the above primal problem. After applying OAA on the Eq. (6), the solution of the primal problem sets the upper bound while the solution of the master problem sets the lower bound. This results in the linearity of both functions [40], [41]. Every next iteration uses integer variables $\Omega^{(n+1)}$ i.e solution of master problem. After the result of every iteration, the bounds get close to each other. A point comes where the difference between bounds remains less than ϵ, the algorithm stops. We can rewrite the problem as:

$$\begin{aligned} &\min_{\Omega} \min_{P} \quad -\mathcal{J}(\Omega^n, P) \\ &\text{subject to}: \quad F_{c1-c9}(\Omega^n, P) \leq 0 \end{aligned} \tag{8}$$

or

$$\min_{\Omega} \quad -\gamma(\Omega) \tag{9}$$

here

$$\begin{aligned} \gamma(\Omega) = \min_{P} \quad & -\mathcal{J}(\Omega^n, P) \\ \text{subject to}: \quad & F_{c1-c9}(\Omega^n, P) \leq 0 \end{aligned} \tag{10}$$

The master problem in Eq. (9) is the projection of our formulated optimization problem in Eq. (6) on Ω space discrete variables. There is a constraint qualification for every primal problem solution Ω^n, which implies that the projected problem will have a similar solution written as:

$$\begin{aligned} &\min_{\Omega} \min_{P} \quad -\mathcal{J}(\Omega^n, P^n) - \nabla\mathcal{J}(\Omega^n, P^n)\binom{P-P^n}{\Omega-\Omega^n} \\ &\text{subject to}: \; F_{c1-c9}(\Omega^n, P^n) - \nabla F_{c1-c9}(\Omega^n, P^n)\binom{P-P^n}{\Omega-\Omega^n} \leq 0 \end{aligned} \tag{11}$$

Equivalent minimization problem with new variable $\mathcal{W}$ can be written as:

$$\begin{aligned} &\min_{\Omega,P,\mathcal{W}} \quad \mathcal{W} \\ &\text{subject to}: \; \mathcal{W} \geq -\mathcal{J}(\Omega^n, P^n) - \nabla\mathcal{J}(\Omega^n, P^n)\binom{P-P^n}{\Omega-\Omega^n} \\ &\qquad F_{c1-c9}(\Omega^n, P^n) - \nabla F_{c1-c9}(\Omega^n, P^n)\binom{P-P^n}{\Omega-\Omega^n} \leq 0 \end{aligned} \tag{12}$$

Lower bounds are calculated using the solution of the master problem Eq. (12). This master problem is equal to the formulated problem in Eq. (6), only if all the mentioned propositions are satisfied. The Eq. (12) is of MILP type and can be solved using an iterative approach. The pseudo-code of the proposed OAA is given in Algorithm 1.

B. ALGORITHM CONVERGENCE AND OPTIMALITY

Proof of linear convergence rate of OAA is given in mixed-integer programming literature [41]. The branch and bound architecture make OAA optimal in $\epsilon = 10^{-3}$. In this procedure, discrete values of Ω are fixed which means that any combination of nodes and cloudlets will never be used twice. If all the four propositions are satisfied and there are a limited number of discrete variables Ω, then Algorithm 1 terminates with the optimal solution in finite steps [37]. The solution is guaranteed using ϵ-optimal algorithms within the ϵ of the optimal solution for any $\epsilon > 0$. The guaranteed accurate value of the solution is given by lower values of ϵ. For a specific choice of discrete variables Ω^n, the optimality of P in master problem Eq. (12) might be:

1. if $\mathcal{W} \geq \mathcal{J}(\Omega^n, P^n) \rightarrow$ *feasible solution*
2. otherwise $\mathcal{W} \leq \mathcal{J}(\Omega^n, P^n) \rightarrow$ *not* $-$ *feasible solution*

The algorithm eliminates such values of Ω^n for which there is no feasible solution that exists for the master problem. This results in finite algorithm convergence. For any fixed values of Ω, the algorithm optimality follows from the convexity of the objective and constraint function. A detailed convergence proof of the OAA algorithm is given in [42]. Using ESA for Eq. (6), a globally optimal solution can be calculated, but there is an exponential computational load increase as it has to enumerate all nodes and fog selection options. Denoting $\mathcal{C}_{ESA}$ as complexity and k as the number of nodes (in our case: fog+IoT) in the network, the computational complexity of the ESA will be given as:

$$\mathcal{C}_{ESA} = 2^{2k} \tag{13}$$

Similarly, the stochastic nature of GA makes it more complex. Its complexity depends on the operators and their implementation. The $\mathcal{C}_{GA}$, complexity of GA will be given as:

$$\mathcal{C}_{GA} = \sigma k, \tag{14}$$

here σ is the number of generations [43]. Space complexity is twice the population size (in our case fog+IoT nodes). But in an infinite number of iterations using OAA, an ϵ-optimal solution can be found [37]. In a generalized way, the complexity $\mathcal{C}_{OAA}$, for OAA will be given by:

$$\mathcal{C}_{OAA} = \frac{k^2\rho}{\gamma}, \tag{15}$$

where ρ is the number of constraints and error tolerance of ϵ-optimal solution from the global optimal solution is given by γ. One more advantage of OAA over ESA is that it ensures to provide an ϵ-optimal solution. In this paper, we compare OAA results with the results obtained

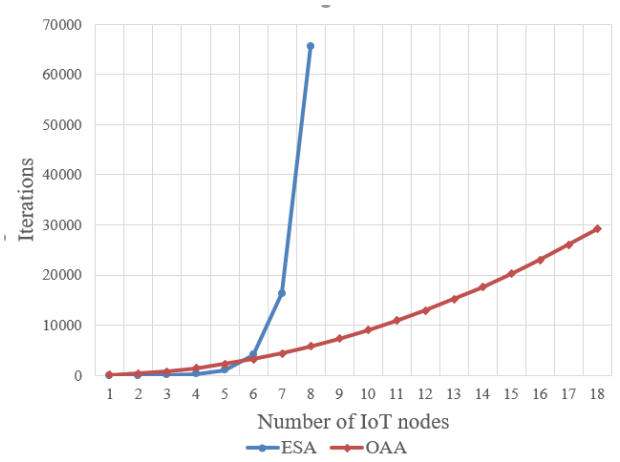

FIGURE 2. Computational complexity of ESA and OAA vs number of IoT nodes.

from GA. After benchmarking GA results, it can be seen that the performance of OAA is better than GA. GA could not perform well, as it cannot guarantee any optimal or ϵ-optimal solution. Also, there is no convergence proof for GA. The computational complexity trend of ESA and OAA is presented in Fig. 2. The calculation of the number of iterations for GA is not possible because of the statistic nature which depends on the number of generations (Eq. (14)).

Algorithm 1 Outer Approximation Algorithm

1: n $\leftarrow$ 1
2: Initialize Ω^n
3: $\epsilon \leftarrow 10^{-3}$
4: *Convergence* $\leftarrow$ *FALSE*
5: *While Convergence* == *FALSE* **do**
6: $P^n \leftarrow \begin{cases} arg\ min, & -\mathcal{J}(\Omega^n, P); \\ subject\ to, & F_{c1-c9}(\Omega^n, P) \leq 0 \end{cases}$
7: $Upper\_Bound \leftarrow \mathcal{J}(\Omega^n, P^*)$
8: $(\Omega^*, P^*, \mathcal{W}^*) \leftarrow \begin{cases} arg\ min_{\Omega,P,\mathcal{W}}\ \mathcal{W}, \\ subject\ to, \\ \mathcal{W} \geq -\mathcal{J}(\Omega^n, P^n), \\ -\nabla\mathcal{J}(\Omega^n, P^n)\binom{P-P^n}{0}, \\ F_{c1-c9}(\Omega^n, P^n), \\ -\nabla F_{c1-c9}(\Omega^n, P^n)\binom{P-P^n}{0} \leq 0 \end{cases}$
9: $Lower\_Bound \leftarrow \mathcal{W}$
10: **if** $Upper\_Bound - Lower\_Bound \leq \epsilon$ **then**
11: *Convergence* $\leftarrow$ *TRUE*
12: **else**
13: $n \leftarrow n + 1$
14: $\Omega^n \leftarrow \Omega^*$
15: **end_if**
16: **end_while**

C. ALGORITHM COMPLEXITY FOR THE PROPOSED PROBLEM

In this subsection, we will be calculating OAA complexity more specifically for our formulated problem. The total number of flops F gives the complexity of an algorithm. In [44], a real floating-point operation is used to represent a flop. Every operation has its corresponding number of flops such as one flop for the operations of addition, multiplication, or division; one flop for additive and removal operator; one flop for a logical operator (e.g. comparison etc.) and assignment operator; two flops for complex addition, four flops for complex multiplication, and $2mno$ flops for matrix multiplication having mxn and nxo dimension. To find the complexity of the proposed algorithm for our scenario, we have to count the number of flops. In our proposed algorithm, the first five statements take one flop each. Statement 6 takes two flops as 2NG, statement 7 and 8 take 4NGβ flops each, statement 9 takes 2NGβ, statement 10 takes two flops, statement 11 takes one flop, and statement 13 takes two flops. The total flop count F_{OAA} for our system having N number of IoT nodes, G number of fog/cloudlet nodes and β is a constraint count for each node, mathematically given as:

$$F_{OAA} = 5 + 2NG + 4NG\beta + 4NG\beta + 2NG\beta + 1 + 2 + 1$$
$$F_{OAA} \approx 2NG + 10NG\beta \quad (16)$$

V. SIMULATION AND RESULTS

To solve Eq. (6) and calculate the overall latency of the network, experimental validation is done for the proposed system model. The experimental results portray the performance of the OAA approach. The results also give some insight into the convergence of the proposed algorithm. To implement the outer approximation linearization technique, *b*asic *o*pen-source *n*onlinear *m*ixed *in*teger programming *(BONMIN)* [45] solver is used.

A. SIMULATION SETUP

The simulation assumptions of parameters are summarized in Table 3. For all the simulation maximum coverage distance for each fog is set to 50m. The maximum transmitted power for a fog/cloudlet node P_g is set as 41dBm. The minimum data rate and latency requirement for any node is set to 200kbps and 1ms, respectively. Reference distance as per antenna far-field d_o is set to 5m and d_g is always greater than d_o. Path loss exponent α is set to 2 and zero mean gaussian variable for shadowing ζ is set to 10dB. The minimum IoT nodes allowed are 2 in the network, whereas the maximum IoT nodes allowed are 18 with an increment of 2. IoT nodes are supposed to be uniformly distributed in the network. Minimum fog nodes are 3, whereas maximum fog nodes allowed are 5, with an increment of 1. These fog nodes are at different distances from the cloud, resulting in different propagation delays. The processing delay at the fog node is set as 0.1ms. The storage capacity for cached data is set as 40MB.

B. DISCUSSION ON RESULTS

In this work effect of power, cache size, and QoS requirements were evaluated using system-level simulations. In the end, we compare the performance of OAA with the ESA

TABLE 3. System assumptions [29], [32], [46].

Parameter	Value
Max. transmit power of fog/cloudlet, P_g	41dBm
far field reference distance, d_o	5m
Max. fog/cloudlet coverage distance, d_g	50m
Min. data rate requirement of an IoT node R	200kbps
Min. latency requirement of an IoT node, L	1ms
Processing latency at fog/cloudlet, $l^P_{g,n}$	0.1ms
Storage capacity at fog/cloudlet, S_g	40MB
Path loss exponent, α	2
Antenna gain, G_o	50
Zero mean gaussian variable for shadowing, ζ	10dB
Min. number of IoT nodes	2
Max. number of IoT nodes	18
Min. number of fog/cloudlet nodes	3
Max. number of fog/cloudlet nodes	5

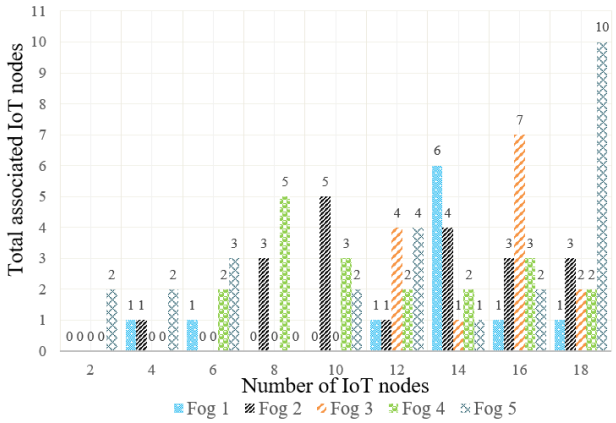

FIGURE 3. User association with five fog nodes vs number of IoT nodes.

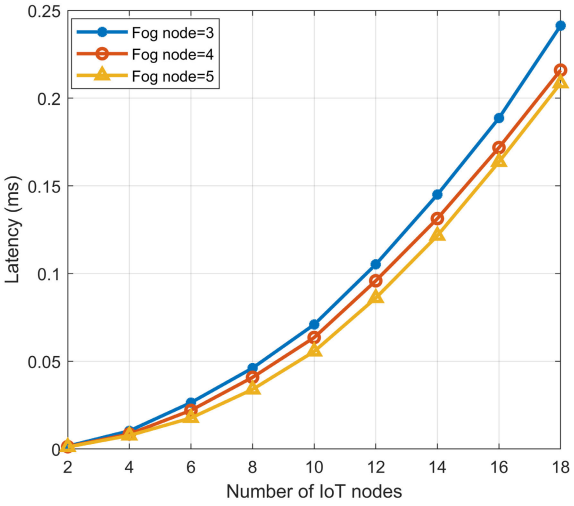

FIGURE 4. Total system latency vs number of IoT nodes.

and standard continuous GA [47]. OAA algorithm is used to calculate network latency under constraints. Fig. 3 shows the association when there are five patches of fog nodes available in the system. The number of associated IoT nodes with fog nodes versus total nodes in the fog-IoT network is observed. Node gets associated with a particular fog node based on the best channel which gives the minimum latency. If an IoT node doesn't receive the minimum QoS requirements based on constraint C2, it is not admitted by any fog node for transmission, and hence such nodes do not contribute in the calculation of overall network latency. With an increase in IoT nodes in the network, user association is maximized while keeping the QoS constraints (C4, C6, and C7) into consideration. The admission of nodes at fog nodes is a random pattern, depending on QoS (power, latency, and rate) requirements. The user association is done with an aim of latency minimization objective (Eq. (6)).

Fig. 4 depicts network latency relation with the number of IoT nodes, in presence of different numbers {3-5} of cloudlet nodes in the network. The total system latency is calculated using Eq. (4) and Eq. (6) considering all the constraints. For a fixed number of fog nodes present in the network, the increasing number of IoT nodes increases total system latency, which is obvious as more time is required to serve all nodes. Observing the effect of the number of fog nodes on latency, at the start, the number of fog nodes in a system has no effect on latency as the number of IoT nodes in the network is less. But as the number of IoT nodes increases, the presence of more fog nodes depicts effects on the total latency. For three fog nodes, all nodes have to be served with only these available three fog nodes, resulting in more latency. However, if there are five fog nodes in the system, a positive effect can be seen. The total network latency decreases as there are more resources available for downloading the requested files. For three fog nodes, 2 IoT nodes will incur 0.01ms latency, while for 18 IoT nodes will incur 0.2ms (around 95% of the increase in latency). This proofs the fact of increasing latency with an increase in the number of nodes in the system. The calculated latency values show that the proposed network model can be used to support real-life applications in the future, for example, smart city, healthcare, and industrial floor. The typical QoS (delay) requirements of internet traffic are: RT data (1ms-1s), image(1s), audio and video (0.25s). Similarly based on the application scenario the forecast QoS delay requirements are: industry 4.0 ($\leq$5ms), internet of energy ($\leq$200ms), big data streaming ($\leq$100ms), smart city($\leq$10ms), factory automation (0.25-10ms), healthcare (1-10ms), robotics, and virtual reality (1ms). Similar more use cases with delay and data rate requirements can be seen in [48], [49]. Table 2 in [3] also summarized the use of fog computing to support the application case studies.

Fig. 5 depicts the relation between the number of IoT nodes and mean user throughput. For a specific node, the throughput is almost the same satisfying the minimum rate requirement. As the number of fog nodes increases, it is observed that there is an increase in instantaneous power

TABLE 4. Percentage reduction in the latency for various cache capacity with respect to no cache.

Number of IoT nodes	2	4	6	8	10	12	14	16	18
%age Reduction for S_g=20MB	11.10%	57%	52%	53.60%	47.65%	44.07%	42.12%	40.02%	35.70%
%age Reduction for S_g=60MB	38.80%	71%	65.50%	67.08%	64.40%	60.86%	57.33%	53.85%	49.27%

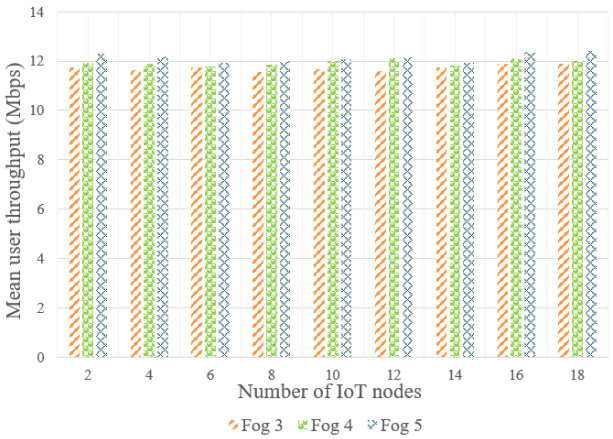

FIGURE 5. Mean user throughput vs number of IoT nodes.

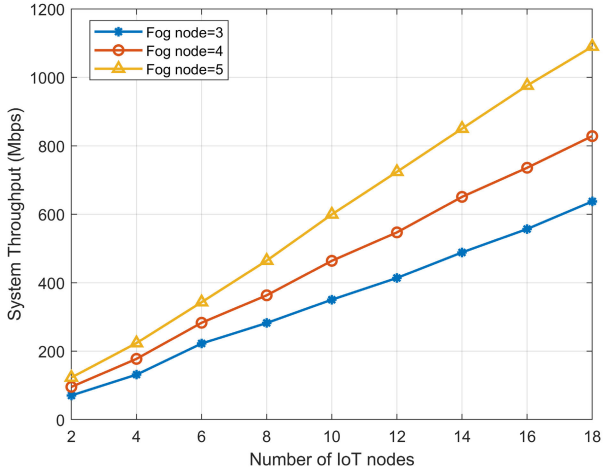

FIGURE 6. Total system throughput vs number of IoT nodes.

which gives more throughput to the user. The user will get more options for receiving the required files, hence the mean user throughput increases with more resources available. This relation i.e., Eq. (5) depicts the direct relation between capacity and power. Figure shows the relation between the number of IoT nodes in the system and the mean user throughput of an IoT node. The graph shows that the mean user throughput is almost the same satisfying constraint C7 of the problem (Eq. 6). The increasing total system throughput can be evident from Fig. 6 in relation to increasing IoT nodes in the system. More IoT nodes will generate more data as compared to less number of nodes. Similarly, the system throughput is maximum for five patches of fog nodes as there

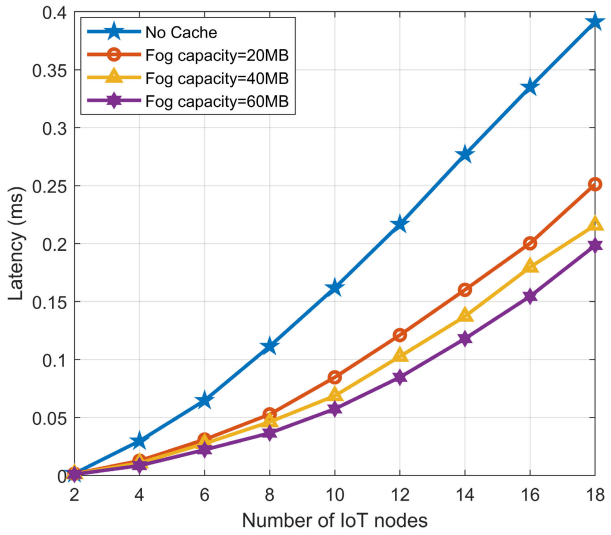

FIGURE 7. Latency vs cache memory size.

will be more resources available for allocation. IoT nodes have more options to fetch their requested files. The sum of the data rate received by all associated IoT nodes in the system gives the total system throughput. The data rate of a single user is calculated using Eq. (5). Precisely, we could say that by increasing the number of IoT or fog nodes into the network, the overall system throughput increases. But there will be a point comes, when any further increase in the number of IoT nodes will not affect the total system throughput, at this point the system's throughput reaches its maximum capacity limit. After this point, the system throughput will be constant.

Fig. 7 shows the effects of cache size on total latency. Total system latency is calculated using the objective function of the formulated problem (Eq. (6)), under different sizes of the storage capacity of fog nodes given in constraint C5. The figure depicts that if the number of IoT nodes increases in the network there will be an increase in system latency. As more number of IoT nodes need more time to fulfill their requirements which causes more delay, hence increasing the overall system latency. To observe the effect of cache size in the figure, it is evident that if there is no cache available at the fog node, the system experiences maximum latency. A node has to experience all the delays namely; transmission, processing, and propagation. A fog node will have to send the request to the cloud server using backhaul links. The sum of all delays results in a maximum latency calculation that does not satisfy the maximum delay threshold for an IoT node. The presence of cache at the fog node has clear positive effects on latency, resulting in the efficient performance of

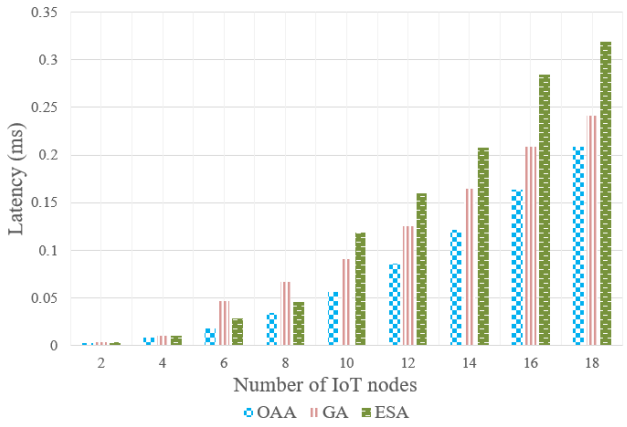

FIGURE 8. Latency comparison of OAA with ESA and GA algorithm.

the network. For 20MB capacity, latency is minimum as compared to $no-cache$, as some demanded files from nodes are available at fog. As we increase the size of the cache at the fog, overall system latency decreases. Table 4 shows the percentage reduction of latency for various cache memory sizes in relation to no-cache. The percentage reduction is more for 60MB size capacity as compared to 20MB. This is because more number of files are present near the vicinity of IoT nodes. At first latency reduction increases, but as the number of IoT nodes increases in the network, the latency reduction decreases. This is due to the fact of being served at the same time, more IoT nodes in the system will take more time.

In Fig. 8, the performance of the network in terms of latency calculation is observed under different algorithms; namely OAA, GA, and ESA. For all algorithms, the parameter assumptions are the same, and latency is calculated using Eq. (6). We can see the performance comparison of the OAA with ESA and GA, for five patches of fog nodes. OAA performs better because of its ϵ-optimal nature over GA. The GA can give good results, but there is no guaranteed optimal solution and convergence proof because of the stochastic nature of GA. When there is a fewer number of IoT nodes are present in the network, the performance of all algorithms is almost the same. But with the increase of IoT nodes, the algorithm's behavior starts some different behavior. ESA is the most complex and the time taken algorithm gives more latency than OAA and GA. In comparison with ESA, OAA is less complex and converges in a finite number of iterations. ESA might give the best optimal solution with a trade-off of the algorithm's complexity. Complexity comparisons of all algorithms in a theoretical way are previously discussed in Section IV-B. The choice of the algorithm also has effects on latency calculations, ESA and GA take more time in finding the optimal solution. This behavior can be numerically observed using equations (Eq. (13), Eq.(14), Eq. (15)).

VI. CONCLUSION AND FUTURE DIRECTIONS

In the very dense IoT application scenario, an extraordinary burden is on the cloud every time. In this paper, CE cloudlet nodes are integrated into the system to reduce the overall latency of the network. A joint MINLP optimization problem is formulated which considers resource association, IoT node admission, power allocation, and cache-availability constraints. The IoT-CE cloudlet system was studied for the minimization of latency. A less complex, branch and bound OAA technique is used to find an $\epsilon - optimal$ solution. Extensive evaluation results show the effectiveness of the proposed approach for our system. A comparison is made in terms of storage capacity at fog nodes and the total number of fog nodes. An increase in intelligently placed data storage at fog node and overall fog nodes in a system deployment results in the minimization of system latency. Designing and optimization of future networks with low latency, more energy-efficient attributes is a vast area of research and many questions need to be answered. For future work, we will find an energy-efficient resource allocation strategy for the proposed system model. We will upgrade the current problem with the reliability factor. For future, mission-critical IoT applications, this ultra-low latency and reliable communication (URLLC) is the major requirement. In this work we have used the standard optimization technique (OAA) to solve the problem, for the future, a heuristic approach can also be used.

REFERENCES

[1] Cisco Visual Networking Index, "Forecast and methodology, 2016–2021," Cisco, San Jose, CA, USA, White Paper C11-481360-01, Sep. 2017.

[2] M. Chiang and T. Zhang, "Fog and IoT: An overview of research opportunities," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854–864, Dec. 2016.

[3] R. Basir, S. Qaisar, M. Ali, M. Aldwairi, M. I. Ashraf, A. Mahmood, and M. Gidlund, "Fog computing enabling industrial Internet of Things: State-of-the-art and research challenges," *Sensors*, vol. 19, no. 21, p. 4807, 2019.

[4] G. Peralta, M. Iglesias-Urkia, M. Barcelo, R. Gomez, A. Moran, and J. Bilbao, "Fog computing based efficient IoT scheme for the industry 4.0," in *Proc. IEEE Int. Workshop Electron., Control, Meas., Signals Appl. Mechtron. (ECMSM)*, May 2017, pp. 1–6.

[5] J. K. Zao, T. T. Gan, C. K. You, S. J. R. Mendez, C. E. Chung, Y. T. Wang, T. Mullen, and T. P. Jung, "Augmented brain computer interaction based on fog computing and linked data," in *Proc. Int. Conf. Intell. Environ.*, Jun. 2014, pp. 374–377.

[6] R. Wang, R. Li, P. Wang, and E. Liu, "Analysis and optimization of caching in fog radio access networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 8279–8283, Aug. 2019.

[7] T. Han and N. Ansari, "Network utility aware traffic load balancing in backhaul-constrained cache-enabled small cell networks with hybrid power supplies," *IEEE Trans. Mobile Comput.*, vol. 16, no. 10, pp. 2819–2832, Oct. 2017.

[8] L. Li, Q. Guan, L. Jin, and M. Guo, "Resource allocation and task offloading for heterogeneous real-time tasks with uncertain duration time in a fog queueing system," *IEEE Access*, vol. 7, pp. 9912–9925, 2019.

[9] X. Cui, Y. Jiang, X. Chen, F. Zhengy, and X. You, "Graph-based cooperative caching in fog-RAN," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, Mar. 2018, pp. 166–171.

[10] S. Bitam, S. Zeadally, and A. Mellouk, "Fog computing job scheduling optimization based on bees swarm," *Enterprise Inf. Syst.*, vol. 12, no. 4, pp. 373–397, Apr. 2018.

[11] H. Zhang, Y. Xiao, S. Bu, D. Niyato, F. R. Yu, and Z. Han, "Computing resource allocation in three-tier IoT fog networks: A joint optimization approach combining Stackelberg game and matching," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1204–1215, Oct. 2017.

[12] S. Meng, Y. Wang, Z. Miao, and K. Sun, "Joint optimization of wireless bandwidth and computing resource in cloudlet-based mobile cloud computing environment," *Peer Peer Netw. Appl.*, vol. 11, no. 3, pp. 462–472, May 2018.

[13] A. Abouaomar, M. Elmachkour, A. Kobbane, H. Tembine, and M. Ayaida, "Users-fogs association within a cache context in 5G networks: Coalition game model," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2018, pp. 14–19.

[14] X. Wei, J. Liu, Y. Wang, C. Tang, and Y. Hu, "Wireless edge caching based on content similarity in dynamic environments," *J. Syst. Archit.*, vol. 115, May 2021, Art. no. 102000.

[15] Y. Jararweh, M. Al-Ayyoub, M. Al-Quraan, L. A. Tawalbeh, and E. Benkhelifa, "Delay-aware power optimization model for mobile edge computing systems," *Pers. Ubiquitous Comput.*, vol. 21, no. 6, pp. 1067–1077, Dec. 2017.

[16] H. Xiang, M. Peng, Y. Sun, and S. Yan, "Mode selection and resource allocation in sliced fog radio access networks: A reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4271–4284, Apr. 2020.

[17] Z. Pooranian, M. Shojafar, P. G. V. Naranjo, L. Chiaraviglio, and M. Conti, "A novel distributed fog-based networked architecture to preserve energy in fog data centers," in *Proc. IEEE 14th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS)*, Oct. 2017, pp. 604–609.

[18] A. Mebrek, L. Merghem-Boulahia, and M. Esseghir, "Efficient green solution for a balanced energy consumption and delay in the IoT-fog-cloud computing," in *Proc. IEEE 16th Int. Symp. Netw. Comput. Appl. (NCA)*, Oct. 2017, pp. 1–4.

[19] 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.

[20] H. Zhang, L. Zhu, K. Long, and X. Li, "Energy efficient resource allocation in millimeter-wave-based fog radio access networks," in *Proc. 2nd URSI Atlantic Radio Sci. Meeting (AT-RASC)*, May 2018, pp. 1–4.

[21] Z. Yan, M. Peng, and M. Daneshmand, "Cost-aware resource allocation for optimization of energy efficiency in fog radio access networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2581–2590, Nov. 2018.

[22] J. A. Khan, C. Westphal, and Y. Ghamri-Doudane, "A content-based centrality metric for collaborative caching in information-centric fogs," in *Proc. IFIP Netw. Conf. (IFIP Netw.) Workshops*, Jun. 2017, pp. 1–6.

[23] S. He, W. Huang, J. Wang, J. Ren, Y. Huang, and Y. Zhang, "Cache-enabled hierarchical transmission scheme for fog radio access networks," in *Proc. 10th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2018, pp. 1–5.

[24] R. Rai, H. Zhu, and J. Wang, "Performance analysis of NOMA enabled fog radio access networks," *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 382–397, Jan. 2021.

[25] L. Tang, X. Zhang, H. Xiang, Y. Sun, and M. Peng, "Joint resource allocation and caching placement for network slicing in fog radio access networks," in *Proc. IEEE 18th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2017, pp. 1–6.

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

[27] Y. Xiao and M. Krunz, "Distributed optimization for energy-efficient fog computing in the tactile Internet," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2390–2400, Nov. 2018.

[28] B. Assila, A. Kobbane, and M. El Koutbi, "A many-to-one matching game approach to achieve low-latency exploiting fogs and caching," in *Proc. 9th IFIP Int. Conf. New Technol., Mobility Secur. (NTMS)*, Feb. 2018, pp. 1–2.

[29] M. S. Elbamby, M. Bennis, W. Saad, M. Latva-Aho, and C. S. Hong, "Proactive edge computing in fog networks with latency and reliability guarantees," *EURASIP J. Wireless Commun. Netw.*, vol. 2018, no. 1, p. 209, Dec. 2018.

[30] M. Ali, N. Riaz, M. I. Ashraf, S. Qaisar, and M. Naeem, "Joint cloudlet selection and latency minimization in fog networks," *IEEE Trans. Ind. Informat.*, vol. 14, no. 9, pp. 4055–4063, Sep. 2018.

[31] G. Lee, W. Saad, and M. Bennis, "Online optimization for low-latency computational caching in fog networks," in *Proc. IEEE Fog World Congr. (FWC)*, Oct. 2017, pp. 1–6.

[32] M. S. ElBamby, M. Bennis, W. Saad, and M. Latva-Aho, "Content-aware user clustering and caching in wireless small cell networks," in *Proc. 11th Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2014, pp. 945–949.

[33] T.-C. Chiu, A.-C. Pang, W.-H. Chung, and J. Zhang, "Latency-driven fog cooperation approach in fog radio access networks," *IEEE Trans. Services Comput.*, vol. 12, no. 5, pp. 698–711, Sep. 2019.

[34] C. Zhu, G. Pastor, Y. Xiao, Y. Li, and A. Ylä-Jääski, "Fog following me: Latency and quality balanced task allocation in vehicular fog computing," in *Proc. 15th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Jun. 2018, pp. 1–9.

[35] C. Shi, Z. Ren, K. Yang, C. Chen, H. Zhang, Y. Xiao, and X. Hou, "Ultra-low latency cloud-fog computing for industrial Internet of Things," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2018, pp. 1–6.

[36] M. F. Hanif and P. J. Smith, "On the statistics of cognitive radio capacity in shadowing and fast fading environments," *IEEE Trans. Wireless Commun.*, vol. 9, no. 2, pp. 844–852, Feb. 2010.

[37] R. Fletcher and S. Leyffer, "Solving mixed integer nonlinear programs by outer approximation," *Math. Program.*, vol. 66, nos. 1–3, pp. 327–349, Aug. 1994.

[38] H. S. Ryoo and N. V. Sahinidis, "A branch-and-reduce approach to global optimization," *J. Global Optim.*, vol. 8, no. 2, pp. 107–138, Mar. 1996.

[39] E. L. Lawler and M. D. Bell, "A method for solving discrete optimization problems," *Oper. Res.*, vol. 14, no. 6, pp. 1098–1112, Dec. 1966.

[40] C. A. Floudas and P. M. Pardalos, *Encyclopedia of Optimization*. Berlin, Germany: Springer, 2008.

[41] C. A. Floudas, *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*. Oxford, U.K.: Oxford Univ. Press on Demand, 1995.

[42] M. A. Duran and I. E. Grossmann, "An outer-approximation algorithm for a class of mixed-integer nonlinear programs," *Math. Program.*, vol. 36, no. 3, pp. 307–339, Oct. 1986.

[43] M. Pelikan and F. Lobo, "Parameter-less genetic algorithm: A worst-case time and space complexity analysis," Illinois Genetic Algorithms Lab., Univ. Illinois Urbana-Champaign, Champaign, IL, USA, IlliGAL Rep. 99014, 1999.

[44] C. Van Loan and G. Golub, *Matrix Computations*, vol. 3. Baltimore, MD, USA: Johns Hopkins Univ. Press, 2012.

[45] P. Bonami. *Basic Open-Source Nonlinear Mixed Integer Programming*. Accessed: Aug. 1, 2019. [Online]. Available: http://www.coin-or.org/Bonmin/

[46] R. Basir, S. B. Qaisar, M. Ali, M. Naeem, K. C. Joshi, and J. Rodriguez, "Latency-aware resource allocation in green fog networks for industrial IoT applications," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2020, pp. 1–6.

[47] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*, vol. 53. Berlin, Germany: Springer, 2003, p. 18.

[48] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A survey on low latency towards 5G: RAN, core network and caching solutions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 3098–3130, May 2018.

[49] M. S. Elbamby, C. Perfecto, C.-F. Liu, J. Park, S. Samarakoon, X. Chen, and M. Bennis, "Wireless edge computing with latency and reliability guarantees," *Proc. IEEE*, vol. 107, no. 8, pp. 1717–1737, Aug. 2019.

RABEEA BASIR received the bachelor's and master's degrees in telecom engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2013 and 2015, respectively. She is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Computer Science, National University of Sciences and Technology (NUST). Her research interests include fog networking, quality of service provisioning, and the IoT networks.

SAAD QAISAR (Senior Member, IEEE) received the master's and Ph.D. degrees in electrical engineering from Michigan State University, East Lansing, MI, USA, in 2005 and 2009, respectively. Since September 2011, he has been the principal investigator or a joint principal investigator of seven research projects spanning cyber-physical systems, applications of wireless sensor networks, network virtualization, communication and network protocol design, wireless and video communication, Internet measurements analysis, multimedia coding, and communication. He is currently an Assistant Professor with the School of Electrical Engineering Computer Science (SEECS), National University of Sciences and Technology (NUST), Pakistan. He is a Lead Researcher and the Founding Director of the CoNNekT Lab: Research Laboratory of Communications, Networks and Multimedia, NUST. He has published more than 80 articles at reputed international venues with a vast amount of work in the pipeline.

MUDASSAR ALI (Member, IEEE) received the B.S. degree in computer engineering and the M.S. degree in telecom engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2006 and 2010, respectively, with a major in wireless communication, and the Ph.D. degree from the School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Pakistan, in 2017. From 2006 to 2007, he worked as a Network Performance Engineer with Mobilink (An Orascom Telecom Company). From 2008 to 2012, he worked as a Senior Engineer in radio access network optimization with Zong (A China Mobile Company). Since 2012, he has been an Assistant Professor with the Telecom Engineering Department, University of Engineering and Technology. His research interests include 5G wireless systems, heterogeneous networks, interference coordination, and energy efficiency in 5G green heterogeneous networks.

MUHAMMAD NAEEM (Senior Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from the University of Engineering and Technology at Taxila, Taxila, Pakistan, in 2000 and 2005, respectively, and the Ph.D. degree from Simon Fraser University, Burnaby, BC, Canada, in 2011. From 2000 to 2005, he was a Senior Design Engineer with Comcept (Private) Ltd., Islamabad, Pakistan, where he participated in the design and development of smart card-based GSM and CDMA pay phones with the Department of Design. From 2012 to 2013, he was a Postdoctoral Research Associate with the Wireless Networks and Communications Research (WINCORE) Laboratory, Ryerson University, Toronto, ON, Canada. Since 2013, he has been an Assistant Professor with the Department of Electrical Engineering, COMSATS Institute of Information Technology at Wah Cantonment, Wah Cantonment, Pakistan, and a Research Associate with the WINCORE Laboratory. He is currently a Microsoft Certified Solution Developer. His research interests include the optimization of wireless communication systems, nonconvex optimization, resource allocation in cognitive radio networks, and approximation algorithms for mixed integer programming in communication systems. He was a recipient of the NSERC CGS Scholarship.