Multiradio Parallel Offloading in Multiaccess Edge Computing: Optimizing Load Shares, Scheduling, and Capacity

Asad Ali and Kanza Ali

***Abstract*—The future will be marked by a highly intelligent, automated, and ubiquitous digital world, requiring fast and reliable connectivity across physical, digital, and biological realms. While multiaccess edge computing (MEC) has facilitated swift connectivity between mobile devices and resource-rich cloud servers, current state-of-the-art solutions may struggle to meet the demands of compute- and bandwidth-intensive applications in the envisioned digital society. To make up for the capacity, 5G and the upcoming 6G expand the channel bandwidth, exacerbating spectrum scarcity, and increasing network costs. To enhance Quality of Service (QoS) and minimize expenses, a recent proposal suggests parallel offloading using multiple radio access technologies (RATs) available on mobile devices, such as Wi-Fi Direct, Wi-Fi, and 5G. However, these technologies differ in performance, including throughput, delay, and response to physical conditions. Inappropriate marshaling of RATs can lead to issues like out-of-sequence packets, resource wastage, and reduced throughput, resulting in longer service delays. To address this problem, we evaluate RAT performance and develop a convex continuous nonlinear program (CNLP) to optimally utilize their capacities, ensuring load distribution aligns with their performance. Additionally, we optimize capacity distribution at relay nodes to ensure smooth MEC data transfer based on incoming load. Numerical results demonstrate significant improvements in terms of throughput, delay, and QoS compared to other techniques involving multiple RATs for computation offloading.**

***Index Terms*—5G, delay minimization, multiaccess edge computing (MEC), multiradio access technology (Multi-RAT) transmission, Wi-Fi, Wi-Fi Direct.**

I. Introduction

MULTIACCESS edge computing (MEC) has been paramount to research for about a decade [1]. It is a fabulous way to make mobile devices appear to execute high-processing gaming, demanding scientific algorithms and computations despite their limited processing power, storage, and battery size. Notably, significant advancements have been made in this field. However, as we are forging forward to a fully connected digital world, the current state of the art may fall short to accommodate the explosive growth in the number of wireless devices and real-time bandwidth-intensive applications, such as virtual/augmented reality, holographic telepresence, Internet of Everything, smart grid 2.0, Industry 5.0, and robotics for their stringent requirements [2]. These applications demand ultrahigh data rates, real-time access to powerful computing resources, ultralow latency, and exceptional reliability and availability, which already surpass the network capabilities of current infrastructure [3]. To realize the envisioned digital world of the next decade and support such services and applications, both 5G and the upcoming 6G networks enhance communication capacity by extending the channel bandwidth [4]. This exacerbates the already daunting spectrum resource scarcity and adds to the cost of the network. Finally, the advent of Massive IoT, 6G, and MEC itself will result in a massive proliferation of devices, leading to disruptive changes in networks and applications. With large number of clients and requests, servers, and their variables, the existing solutions for MEC can no longer withstand the service requirement of future applications [5].

Our next-generation wireless networks are expected to have enormous capacity to accommodate the ever-increasing demands from bandwidth-intensive applications [6]. Different techniques, such as massive multiple-input–multiple-output (MIMO) and beamforming [7], spatial multiplexing [8], multiband transmission [9] [10], channel bonding and bandwidth aggregation [11], and prioritized processing [12] have been devised to keep up with the rapidly growing real-time bandwidth hungry applications. However, due to new applications' stringent performance requirement, MEC communication still struggles to provide adequate connectivity [13]. Therefore, to meet the service delay requirement of the MEC applications, we exploit the idea of simultaneous offloading over multiple radio access technologies (RATs) [14].

A. Multi-RAT

The concept of multiradio access technologies (Multi-RATs) exploits the fact that today's smartphones are equipped with multiple RATs. Starting with 4G/5G, our smartphones are connected to macro-base station (BS) to provide broad coverage. On the other hand, Wi-Fi, based on the IEEE 802.11 standard, provides ultra fast connectivity, but confined to a local area network. Similarly, Wi-Fi Direct is a peer-to-peer WiFi standard for device-to-device communication without involving intermediary central access point or router [15]. Wi-Fi

Manuscript received 28 March 2023; revised 31 May 2023 and 9 July 2023; accepted 16 July 2023. Date of publication 25 July 2023; date of current version 24 January 2024. This work was supported by the Science Foundation Ireland under Grant 18/CRT/6222. *(Corresponding author: Asad Ali.)*

Asad Ali is with the CONNECT Research Centre, Trinity College, The University of Dublin, Dublin 2, D02 PN40 Ireland (e-mail: alias@tcd.ie).

Kanza Ali is with the Graduate School of Information Sciences, Tohoku University, Sendai 980-8577, Japan (e-mail: kanza.ali@vision.is.tohoku.ac.jp).

Digital Object Identifier 10.1109/JIOT.2023.3298831

Direct has been shown to be a successful avenue for task offloading [16].

B. Why Multi-RAT?

Historically, network densification and channel bandwidth expansion have been the go-to solutions for enhancing network capacity and improving Quality of Service (QoS) [2]. This adds to the cost of the network and rather makes the situation worse by devouring the already scanty radio spectrum. Furthermore, 5G networks are optimized for high data rates over short distances [17]. Applications that demand high data rates over long distances have a huge impact on its performance. Therefore, it may be advisable to complement 5G with WiFi to improve QoS in areas where a single RAT may not be adequate. Additionally, in situations where a certain RAT underperforms, we have the flexibility to redistribute its traffic load to an alternative technology.

Taking into account these factors, we aim to use all three RATs concurrently to offload computationally intensive tasks to MEC servers. Moreover, while improving network capacity has received significant attention, cost reduction has been largely neglected. Given the cost associated with 5G and 6G, multi-RAT offloading mechanism will be a viable solution for delivering MEC services in low- and middle-income, addressing the challenge of affordable broadband connectivity while also enabling the provision of high-end MEC services in high-income economies.

C. Our Contribution

While Multi-RAT does offers flexibility in leveraging the unique characteristics of each RAT in relation to distance, interference, physical barriers, weather, and deployment environment, an ineffective scheduling scheme may prove counterproductive, drastically affecting throughput and adding to service delays as a result of out-of-order transmission and reception of data packets and underutilization of system capacity. To address packet reordering, the transmission control protocol (TCP) employs a receiver buffer to store and reorder out-of-order packets based on sequence numbers assigned by the sender. TCP allows a maximum packet reordering of two positions, beyond which it is considered as loss [18]. Consequently, transmission window reduction occurs due to perceived unfavorable channel conditions, leading to a decrease in transmission rate and the adoption of lower order modulation to compensate. Additionally, a disproportionate or equally distributed load among radios can result in underutilization of capacity. For instance, if the fast RAT completes its transmission while other radios are still active, the idle time of the faster RAT could have been utilized for additional data transmission. Moreover, the MEC server must wait for data from slower RATs before processing the received data from the faster RAT.

Using multiple RATs concurrently has garnered attention in recent studies. These investigations encompass a range of techniques and protocols with diverse objectives and approaches for modeling the computation offloading process. For instance, works, such as [14] and [19], distribute tasks across RATs. This can introduce delays due to packet reordering as tasks experience different delays on different RATs. Certain techniques, like the one proposed in [20], select the best radio-edge server pair instead of using multiple RATs simultaneously. Additionally, the approach presented in [21] partitions the tasks based on the processing capacities of user equipment (UE) and the MEC server. Nevertheless, this technique does not account for gaps between the RATs or queuing analysis that are leading causes of out-of-order packet arrivals.

Considering these factors, we propose an analytical model that aims to optimize scheduling, capacity utilization, and distribution in order to maximize system throughput and minimize end-to-end communication delay. We ensure that simultaneous arrival of packets is achieved without causing out-of-order packet reception. The main contributions of this article are as follows.

1) We formulate a continuous nonlinear program (CNLP) in order to optimally utilize the available capacities of the RATs. The proposed CNLP, solved through Lagrange's multiplier theorem for several constraints, avoid the reordering delay by equalizing communication delay across all the RATs equal thereby varying the load on the RATs according to their performances. Capacities and performance of the RATs are computed a priori.
2) We develop a technique that optimally distributes the capacity at the relay node among different users according to the incoming traffic load, so that MEC traffic is relayed without disruption.
3) We develop a packet scheduling technique that distributes the traffic among the RATs in such a way that packet order is maintained at source and destination thereby completely avoiding packet reordering delay to keep the throughput intact.

The remainder of this article is structured as follows. In Section II, we give a brief overview of the state of the art on computation offloading over multiple RATs. In Section III, we will discuss our computational offloading over multi-RAT model and describe our problem formulation. We solve the formulated problem in Section IV. In Section V, we incorporate spatial and temporal variation in channel. We discuss performance evaluation in Section VI. Finally, the conclusion and future work are given in Section VII.

II. Literature Survey

Computation offloading is one of the oldest topics of computing and probably the main motivation behind computer networks, as can be seen in the memo shared by Licklider in 1963 [22]. More recently, the increasing popularity of bandwidth hungry applications in conjunction with mobile devices brought this issue into limelight. Computation offloading to remote central cloud servers is often unsuitable for real-time applications, as the transmission distance and number of hops required to reach a central computing node typically incur latency of several tens of milliseconds, with comparably high jitter. MEC, on the contrary, outdoes traditional cloud computing by significantly enhancing the capabilities of capacity-limited mobile devices thereby remarkably reducing the service delay [23]. It is for this reason that MEC manifests itself as promising technology for extending the computation and storage capabilities of mobile devices.

We acknowledge that over the last few years there have been a large number of studies focusing on the technical aspects of the MEC [24], [25]. Most of the solutions are single RAT-based and are inadequate to incorporate several key characteristics and are often too simple to reflect real-world scenarios. In the following discussion, we shall divide our review of the literature into two parts. We shall review the shortcomings in the existing offloading techniques in general and then in the second part, we shall review the work done in the context of multi-RAT systems.

A. *Computation Offloading in General*

Computation offloading techniques and protocols differ in purpose and how they model the computation offloading process. A detailed review of computation offloading modeling is given in [26]. Most of the existing works have assumed constant values for several important parameters, such as signal-to-noise ratio (SNR), bitrate, received signal power, path loss, etc. [27], [28], [29], [30]. Similarly, [31] has considered constant values for transmission and processing delay. Assuming constant values for these important parameters is not realistic and leaves little room for improving the performance. Similarly, several techniques assume that static offloading where network haphazardry (i.e., the fact that networks are dynamic in nature) and spatiotemporal variation in the network is ignored [32], [33], [34]. With user mobility and nature of applications combined with variation in wireless channels, MEC-based wireless networks are highly dynamic. Therefore, using deterministic optimization models fall short in real-life scenarios.

In some recent works, to make efficient use of resources, [35] considers local computation by partially offloading the tasks to MEC servers. The authors consider both single as well as multiusers scenarios of MEC resource allocation for computation offloading, which are solved by branch-and-bound and iterative-based heuristics, respectively. Schemes, like partial offloading, require perfect user-MEC server-remote cloud coordination that leads to high signaling overhead. Moreover, these schemes assume that a task can be arbitrarily divided into subtasks which is an unreal assumption. To further ameliorate the resource allocation, Wu et al. [36] investigated the efficiency of deep reinforcement learning and developed solutions for joint resource allocation and energy minimization based on deep Q-networks (DQNs). The authors developed techniques based on DQN, convex optimization, and traditional Q-learning. However, offloading learning policy is for fixed topology and given the efficiency of DQN, they are not suitable for edge video processing. The goal of minimizing energy consumption and processing delay is carried forward in [37], where the authors have developed an evolutionary algorithm that jointly optimizes energy consumption and processing delay and attempts to find pareto-optimal point between energy consumption and processing delay.

Computation offloading is also investigated in vehicular edge networks (VECs) in [29], where the authors have worked on selecting least congested edge server with an aim to minimize cellular hand-offs to avoid obstruction in computation. Furthermore, [38] has proposed unmanned aerial vehicle (UAV) as MEC server. Assuming MEC servers in remote or disaster-hit areas are not deployed overnight; in conjunction with the cost-benefit reasons, UAV-enabled servers can be a plausible mechanism to make up for the infrastructure-based MEC servers. Qin et al. [38] have given time-varying priority to the reconnaissance task and total reconnaissance utility has been maximized through an optimization problem.

To summarize, given the dynamic nature of MEC applications and wireless networks, the assumption taken in most of the existing solutions is not at par with real world. The networking and processing models have several flaws, such as taking fixed delays, loads, and capacities. Moreover, the fact that BS serves as a relay node and that the transmission capacity and communication-related delays post- and pre-BS can be different is ignored. Furthermore, the existing solutions are not scalable enough to cope up with massive IoT and service requirements of future applications. Therefore, we need a solution that is scalable, flexible, and completely represents the actual networking and processing operation of the real world.

B. *Parallel Offloading Over Multi-RAT*

The co-existence of Wi-Fi and macro-cell networks, such as LTE, has been a widely studied research area [39]. However, in WiFi-LTE-integrated networks, only a portion of the capacity of the WiFi AP is used, and data are offloaded to Wi-Fi with the aim of improving the cost and throughput of a cell. Similarly, most of the studies investigate downlink performance [40]. We, on the contrary, investigate the synergy of WiFi, WiFi Direct, and cell network, and offload the data to a remote server and use any portion of capacity of any RAT depending upon the channel condition. Leveraging multi-RATs in the context of MEC offloading has been carried out in [14], [19], and [20]. Within this frame of reference, Braud et al. [14] offloaded data on the basis of the tasks in mobile augmented reality context. For instance, one task is sent over one RAT while another task is sent over another RAT. Distributing data on the basis of the computational tasks can lead to packet reordering delay as they can be of different sizes. Moreover, performance is measured on the basis of the transmission delay and ($Load/Bandwidth$) metric. Different important parameters, such as queuing delay, processing delay at the node, and congestion, are ignored. Offloading augmented reality requests have also been contemplated in [41] with a totally different direction. The authors have come up with an idea of generating new revenue streams for network service providers by reward maximization through task offloading in AR applications. However, the proposed algorithm is based on online learning which still has to cope up with the issue of high velocity of data with time-varying distribution.

Multi-RAT offloading is further carried forward in [20] where the proposed algorithm requires the end node to send

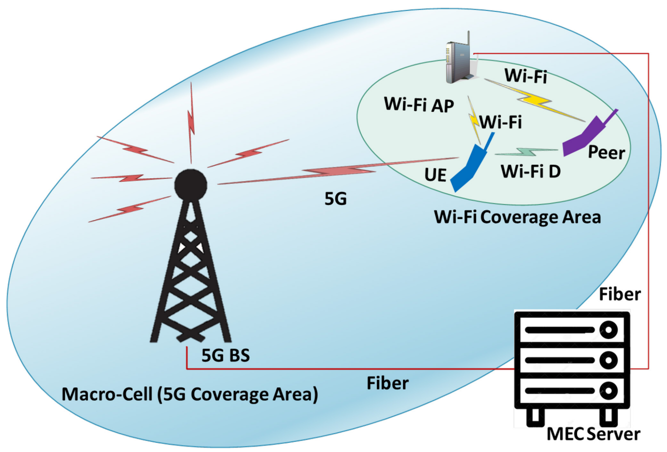

Fig. 1. Simple illustration of multi-RAT multipath computation offloading.

all the information to the relay node, such as required latency, data rate, average packet length, average packet arrival rate, required computing power, and so on. We believe, relying on real-time values of scores of parameters for real-time applications will make the job cumbersome and will defy the real purpose of task offloading in real world. Moreover, the radio-access technologies are not used simultaneously, rather the choice is made for best radio-edge pair. Finally, a detailed analytical framework of the presented work is also missing. Similarly, [19] distributes data flows on the basis of tasks which is subject to packet reordering delay.

We argue that load, when offloading, must be distributed among RATs according to the data size rather than tasks be sent over different RATs. This distribution must be done according to the channel health and performance of the link which must be duly computed. In addition, the system should maintain the order of the data packets.

III. Computational Offloading Over Multi-RATs

In this section, we first introduce the multi-RAT computation offloading MEC model considered in our work, followed by a description of our proposed computation offloading. After that, we will formulate the problem for multi-RAT simultaneous computation offloading.

A. Assumed System Model

Our assumed system model is summarized in Fig. 1. Starting from the end node, we have a smartphone as UE that acts as a source node. The UE is equipped with multiple RATs, such as 5G Transceiver, Wi-Fi, and Wi-Fi Direct. We assume that it is in range of a 5G BS, Wi-Fi access point, and occasionally, a peer device comes in its transmission range. Therefore, it can transmit through 5G, Wi-Fi, and Wi-Fi Direct simultaneously. In the figure, a peer is any device that serves as a relay node and has same features as the end node itself, that is, any device capable of transmitting over 5G, Wi-Fi, and Wi-Fi Direct. In the figure, the end node is mobile in nature whereas the Wi-Fi access point and 5G BS are fixed. We also assume that Wi-Fi access point and 5G BS can serve multiple users simultaneously. Both the Wi-Fi AP and 5G BS are connected to a single MEC server by optical fiber connection.

TABLE I
Summary of Notations

Notation	Description
D_l	Delay when computation performed locally
$D_v^{''}$	Delay of the slowest radio when task are offloaded over v RATs
Δ	Packet re-ordering delay
δ	Packet retransmission delay
Θ	Propagation Delay
λ_U	Traffic Load on a particular link u
d_u	Delay of the link u
D_r	End-to-End Delay on RAT r
ζ_i	Capacity of RAT i
μ	Packet length
R_u	Data rate of link u
λ	Packet Arrival Rate
λ_u	Load on link (u)
L_i	Load on link i
h_r	Load share ratio of an arbitrarily chosen radio r
T_{sd}	Traffic from node s to node d
τ_{ij}	Delay of the link u
m	Upper limit for the number of hops

Suppose a task has been generated by the application, the end node has two options, either execute the task locally or offload the computation to MEC server. Thus, service delay can be mathematically expressed as follows:

$$D = \begin{cases} 1, \; D_l \\ 2, \; D_v^{''} + \Delta + \delta \end{cases} \tag{1}$$

here D_l is the service delay when a task is executed locally. Service delay in this case includes computation delay and queuing delay, that is, the time the packet waits in the queue to get the processor. Similarly, $D_v^{''}$ is the service delay of the slowest RAT when a task is offloaded over v number of RATs. Service delay in this case also includes transmission-related delays in addition to packet reordering delay (Δ) and retransmission delay (δ). Packet reordering delay is incurred when packets arrive out-of-order at the MEC server. Packet retransmission delay is incurred when because of the gap between the received packets, receiver is unable to order the packets as per the sequence number and asks the sender to retransmit the packets. Further details about packet reordering and retransmission are given in Section III-C. Finally, the upper bound for v depends upon the number of RATs available as we assume not all RATs are available all the time, for example, a peer node may be unavailable, channel conditions on certain RAT may not be favorable for transmission or a remote area where macro-cellular services are accessible. For better understanding, key notations are summarized in Table I.

If a task is executed locally, the service delay will be the sum of processing time and the time it waits in the queue to get the processor. Assuming Poisson processes with rate λ, if ζ is the processing capacity of the device, processing time will be $1/\mu\zeta$, where μ is the size of the process. Similarly, the queuing time will be $\lambda/((\mu\zeta)(\mu\zeta - \lambda))$. Therefore, the service delay if the task is executed locally will be

$$D_l = \frac{1}{\mu\zeta} + \frac{\lambda}{\mu\zeta(\mu\zeta - \lambda)}. \tag{2}$$

We will formulate the service delay for the offloaded task in the next section. Next, suppose the task is computationally intensive and cannot be executed locally or local execution time is very large from what would have been if the task was executed locally, that is

$$D_l >> D_v^{''} + \Delta + \delta. \tag{3}$$

It has been shown that multiaccess edge servers are capable of achieving better computation performance than local computation schemes in the number of studies [36]. Therefore, let us assume the end node opt to offload the task to the MEC server to speed up the processing. Now, the questions arise how much traffic load should each RAT get, and how to schedule the traffic among the RATs to avoid packet reordering delay at the receiver's end. Therefore, once the system decides to offload the task, the goal is to minimize the delay while keeping in view these considerations. We would also like to mention that following the general notations trend, μ will be used as the packet length and λ will be used as data load.

B. Delay When Computation Offloaded

In this section, we provide the mathematical model for computation offloading and formulate the objective function for our proposed CNLP.

When computation offloading is decided, other than the processing delay and queuing delay mentioned above, we will have transmission delay and slot-synchronization delay for wireless network data transmission. The data in wireless networks is governed mainly by four different types of delays, namely, queuing delay, slot synchronization delay, transmission delay, and propagation delay [42]. When a data packet arrives at certain node, it is kept in the queue before it gets its turn for processing or transmission. This is the time the packet spends in routing queues and is called queuing delay. Queuing delay depends upon the capacity of the transmitter and packet arrival rate. Denote μ as the packet length, ζ_u as the capacity of the link (u), and λ_u as the load on link (u): the average queuing delay for a single link (u) can be obtained as $\lambda_u/(\mu\zeta_u)(\mu\zeta_u - \lambda_u)$ [43]. Similarly, assuming a time-division multiple access (TDMA)-based transmission where synchronization among the nodes is important, slot-synchronization delay will incur when the node has to synchronize its operation with the neighboring wireless nodes. The packet will wait for getting its designated time slot before it is transmitted. Average slot-synchronization delay can be obtained as $1/2\mu\zeta_u$ [42]

After getting its designated slot, the packet is transmitted into the link. The associated delay is given by $1/\mu\zeta_u$. Finally, time taken by the signals to propagate from source to destination is referred to as propagation delay, which depends upon the propagation distance of the signal [44]. We can see that these delays will keep adding as the packet traverse relay nodes. Combining the four quantities, we get packet delay of the link (u) as follows:

$$d_u = \frac{1}{2\mu\zeta_u} + \frac{1}{\mu\zeta_u - \lambda_u} + \Theta \tag{4}$$

here Θ is the propagation delay. Let the total number of hops from source to destination be m; for any arbitrarily chosen RAT r, our goal is to minimize the following:

$$D_r = \sum_{u=1}^{m}\left(\frac{1}{2\mu\zeta_u} + \frac{1}{\mu\zeta_u - \lambda_u}\right) + \sum_{u=1}^{m}\Theta_u \tag{5}$$

where ζ_u and λ_u, respectively, are the capacity and load of link u. For the same packet size, (5) shows that delay is a function of the capacity of the link and the load. Therefore, in order to minimize the delay, we must optimize the load on the RATs in order to optimally utilize the obtained capacities. Since the propagation delay is independent of capacity and load, and only dependent on distance, its value is added at the end of the computation.

C. Continuous Nonlinear Program Formulation

We assume that the source node is mobile, and its transmission capacity is driven by its SNR which is primarily a function of its distance from the relay node. Assuming both WiFi and cellular networks are equipped with scheduled access [45], Nonorthogonal multiple access (NOMA) and beamforming capabilities [6], [46], both the technologies have interference suppression. Therefore, we ignore the interference and take the SNR obtained as a result of distance and operating environment only, for computing the transmission capacity of both the RATs. Moreover, we develop a technique to handle any change in channel condition due to SNR or other factors in Section V. Furthermore, for computation purposes, instead of relying on transmission capacity, we take a more practical approach by considering the number of bits per second received successfully which is essentially synonymous to system throughput. If BER_u is the bit error rate and R_u is the data rate of link u, we can write the capacity as follows:

$$\zeta_u = (1 - \text{BER}_u)^l \times R_u \tag{6}$$

where l is the packet length. The bit error rate (BER_u) of the channel taken here is after applying a low-density parity check (LDPC) correction code. To optimally utilize the available capacity, the load can be contrived in a way that makes maximum use of the available capacity as disproportionate or equally shared load will lead to underutilization of capacity as discussed in Section I-C. Here, it may be noted that with equally distributed load, the channel and the time slot of the faster RAT, once done with transmission of its load share, can be employed by other nodes in the network. However, the radio of the UE remains idle despite the fact that there exist data load which the UE has allocated to other RATs.

Moreover, the MEC server cannot take action on the transmitted data as it is waiting to receive the remaining data. Therefore, the processing at the MEC server is hampered by the slow RATs. Additionally, if the order of the packets at the receiver is different from the order of the same packets at the sender, the processing will be further hindered by packet reordering. In case of out-of-order reception, packets are cached in the receiver's buffer and reordered according to the sender's sequence number. Consequently, the transmission window is reduced, as these losses are attributed to unfavorable channel conditions. As a consequence, the sender drops the transmission rate, that is, using a lower order modulation, in order to make up for the change in channel condition [18].

As a result, we see a sharp decline in the system throughput. The decrease in transmission rate as a result of reduction in transmission is clearly underutilization of the available capacity. This situation can be made up for if the delays of all the RATs are equal. Therefore, the first objective of our system is to make the delays of all the RATs equal, that is

$$D_r = D_t = D_v. \tag{7}$$

In (7) D_r, D_t, and D_v are the delays of the three arbitrarily chosen RAT r, t, and v. For (7) to hold, it is necessary for the participating RATs to always assume some load during the transmission

$$h_r > 0 \quad \forall\ \lambda > 0 \tag{8}$$

here h_r is the load share ratio of an arbitrarily chosen RAT r and λ is the total load. It follows that sum of load share ratios of all three RATs cannot exceed 1, that is

$$\sum_{r=0}^{v} h_r = 1. \tag{9}$$

Equation (9) ensures that sum of loads on individual RATs cannot exceed total incoming load, that is

$$\sum_{r=1}^{v} \lambda_r = \lambda. \tag{10}$$

In (10), λ_r is the load on RAT r while λ is total load generated by the device. Moreover, load on the RATs cannot be negative. Therefore, we have to make sure that load share ratios of all the RATs are always positive

$$h_r \geq 0. \tag{11}$$

Finally, the load on a RAT cannot exceed its capacity

$$\sum_{r=1}^{v} \lambda_r \leq \zeta_r. \tag{12}$$

Equations (7)–(12) ensure optimal capacity utilization and in-order delivery of packets to the destination. To summarize, the control variables in (7) ensure that delays of all the RATs are equal. In (8), if a RAT is participating, it should always carry some load. Equation (9) make sure that if is load expressed in terms of ratio, the sum of ratios of individual RATs do not exceed 1 which is supported by (11) to make sure that ratio of a load share cannot be negative. Equation (10) ascertains that sum of loads that individual RATs carry do not exceed the total load. Equation (12) is to confirm that load cannot exceed the capacity of a channel. Based on the discussion above, we formulate a CNLP where our objective functions are as follows:

$$\begin{aligned}&\text{minimize}\ \ D_r\\ &\qquad \text{s.t.}\ \ (3),\ \ (6)-(12)\end{aligned}$$

where D_r is the delay of arbitrarily chosen RAT r. Minimizing delay of one RAT will ensure that the delay of all the RATs is minimized as given in (7).

The objective function of the formulated problem exhibits nonlinearity, with all parameters being continuous in nature. In the context of linear problems, a key criterion for achieving optimal capacity utilization and minimal delay in all scenarios is the establishment of equal residual capacity to total capacity ratios across all RATs [47]. However, the applicability of such a solution is not guaranteed for nonlinear formulated problems. To ensure the correctness of any locally derived solution as the global solution, it is imperative to establish the convexity of our formulated problem. A comprehensive proof outlining the convexity of the problem is provided in Appendix A.

IV. Capacity Optimization

In this section, we solve our formulated CNLP to optimize capacity utilization at the source node. After that, we optimize the capacity at the relay node where it is shared among multiple receivers connected to it.

A. Optimizing Capacity Utilization at Source Node

In this section, we develop a solution for the proposed CNLP. We use Lagrange's multiplier theorem for several constraints. The goal here is to find the optimal loads share λ_i for all the RATs for which the delay is minimum. Using the Lagrange multiplier theorem, we rewrite our problem as follows:

$$\begin{aligned}G = \left(\frac{1}{2\mu\zeta_u} + \frac{1}{\mu\zeta_u - \lambda_u}\right) - K_1\left(\lambda - \sum_{r=1}^{n}\lambda_r\right)\\ - K_2(C_1) - K_3(C_2) - \cdots \end{aligned} \tag{13}$$

In (13), K_1, K_2, ... are Lagrange multipliers, λ is the total load, λ_r is the load RAT r will get, and C_1, C_2, ... are the constraints defined in (3) and (6)–(12). Taking partial derivative of (13) with respect to every variable and equaling to 0, we get

$$\begin{aligned}\frac{2}{\mu\zeta_r - \lambda_r} = -\frac{1}{\mu\zeta_r}\Bigg(&\frac{-\frac{2}{\zeta_r} + \frac{1}{\zeta_t} + \frac{1}{\zeta_v}}{2\mu}\\ &+ \frac{1}{\mu\zeta_t - \lambda_t} + \frac{1}{\mu\zeta_v - \lambda + \lambda_r + \lambda_t}\Bigg).\end{aligned} \tag{14}$$

From (7), we have

$$\frac{1}{2\mu\zeta_r} + \frac{1}{\mu\zeta_r - \lambda_r} = \frac{1}{2\mu\zeta_t} + \frac{1}{\mu\zeta_t - \lambda_t}. \tag{15}$$

Solving (15) for λ_t, we get

$$\lambda_t = \frac{\mu\zeta_t(\lambda_r(3\zeta_r - \zeta_t) - 3\mu\zeta_r(\zeta_r - \zeta_t))}{\lambda_r(\zeta_r - \zeta_t) - \mu\zeta_r(\zeta_r - 3\zeta_t)}. \tag{16}$$

Finally, λ_v for RAT v can be obtained by subtracting λ_r and λ_t from the total load, that is, $\lambda_v = \lambda - \lambda_r - \lambda_t$. The proof that λ_r, λ_t, and λ_t are optimal load shares that optimally utilize the available capacity is given in Appendix B. It is important to note that the channel condition is subject to time variation, with changes in physical conditions leading to fluctuations in channel quality. Consequently, the aforementioned load allocations may no longer remain valid in such instances. Thus, it becomes necessary to account for the changing channel condition when determining the load allocations. The procedure for incorporating this change is outlined in Section V.

Algorithm 1: Multi-RAT Traffic Offloading

```
Input: $\lambda$ and $\zeta_r$, $\zeta_t$, $\zeta_u$ of the three RATs $r, t, v$
Output: Communication Delay
  while true do
  • RAT performance computation
  1. Compute end-to-end performance of every RAT using (5).
  • Optimal capacity Uuilization at source link
  1. Compute the three load shares using (14) and (15).
  2. Assign the obtained load shares to the RATs in such a way that minimizes the delay.
  • Capacity optimization at relay nodes
  1. Determine incoming MEC traffic and its outgoing link.
  2. Assign the capacity on its outgoing link according to (26).
  end while
```

B. Optimizing Capacity Distribution at Relay Nodes

According to our system model, the peer node, the Wi-Fi access point, and the 5G BS are serving as relay nodes. Capacity utilization at the relay node is different from that at the source node. Unlike the source node, capacity at the relay node is shared among multiple receivers connected to it. If ζ_t is the total capacity of the relay node and k users are connected to it, mathematically, we can write

$$\zeta_t = \sum_{u=1}^{k} \zeta_u. \tag{17}$$

Relay nodes will be a major bottleneck if packets are not relayed smoothly as a result of dwindling capacity. We overcome this situation by optimizing the distribution of the total capacity ζ_t, such that ζ_u for link u is optimal according to the load λ_u on it.

Suppose a packet travels from source s to destination d, let the traffic from source to destination be T_{sd} and the traffic in other direction be T_{ds}. Also, let there be N sources and M destinations in the network. Therefore, total traffic (T) in the network will be

$$T = \sum_{s}^{N} \sum_{d}^{M} (T_{sd} + T_{ds}). \tag{18}$$

Next, consider two nodes i and j. Let the link between i and j be u and the load on the link u be λ_u. Also, let T_n be traffic load of another node n passing through link u. If there are N nodes in the network whose traffic load passes through link u, total load on link u is given by

$$\lambda_u = \sum_{n=1}^{N} T_n. \tag{19}$$

We know that each link carries a fraction of total traffic load of the network. Assuming the number of links from source to destination is essentially the number of hops and if $\bar{n}$ is the average number of hops that data take from source to destination, mathematically we can express the fraction of traffic load per hop as follows:

$$\bar{n} = \frac{\sum_{n=1}^{N} T_n}{\sum_{s}^{N} \sum_{d}^{M} (T_{sd} + T_{ds})}. \tag{20}$$

Let τ_{ij} be the delay of the link u. We can exploit Little's law to get system delay (Γ) as follows:

$$\Gamma = \frac{\sum_{n=1}^{N} T_n}{\sum_{s}^{N} \sum_{d}^{M} (T_{sd} + T_{ds})} \cdot \sum^{N} \tau_{ij}. \tag{21}$$

Assuming M/M/1 queuing system with capacity ζ_{ij} and Poisson arrival with an average of λ_{ij} packets and average service time of $1/\mu\zeta_{ij}$, τ_{ij} can be obtained as follows [43]:

$$\tau_{ij} = \frac{1}{\mu\zeta_{ij} - \lambda_{ij}} \tag{22}$$

here μ is the average packet length. Using the value of τ_{ij} in (21), we get

$$\Gamma = \frac{\sum_{n=1}^{N} T_n}{\sum_{s}^{N} \sum_{d}^{M} (T_{sd} + T_{ds})} \cdot \sum^{N} \frac{1}{\mu\zeta_{ij} - \lambda_{ij}}. \tag{23}$$

Using (18) and (19), and replacing ij with u, we can rewrite (23) as follows:

$$\Gamma = \frac{1}{T} \cdot \sum^{n} \frac{\lambda_u}{\mu\zeta_u - \lambda_u}. \tag{24}$$

Equation (24) shows the significance of capacity for system delay. In order to minimize the system delay, we must optimize the capacity. We again use the Lagrange multiplier theorem [48] and rewrite our capacity optimization problem as follows:

$$W = \frac{1}{T} \sum_{u=1}^{k} \frac{\lambda_u}{\mu\zeta_u - \lambda_u} - K\left(\sum_{u=1}^{k} \zeta_u - \zeta_t\right) \tag{25}$$

here K is the Lagrange multiplier and $(\sum_{u=1}^{k} \zeta_u - \zeta_t)$ is the capacity conservation constraint as shown in (17). Taking $(\partial W/\partial \zeta_u)$ and equaling to 0, we get

$$\zeta_u = \frac{\lambda_u}{\mu} + \frac{\left(\zeta_t - \sum_{u=1}^{k} \frac{\lambda_u}{\mu}\right) \cdot \sqrt{\lambda_u}}{\sum_{k=1}^{n} \sqrt{\lambda_u}}. \tag{26}$$

Equation (26) shows the capacity that a link u will get according to its load λ_u. Having obtained optimal load shares and capacity at relay nodes optimized, we briefly highlight our proposed MEC offloading technique in Algorithm 1.

While it is true that the service delay of local processing is larger than offloading the tasks to the MEC server, the proposed technique will compute end-to-end delay of all the RATs as described in Section III-B. Based on the performances of RATs, optimal load shares are computed as shown in Section IV-A. Next, we allocate the obtained load to RATs. The load allocation mechanism and the choice of relay nodes are described in Appendix D. Having transmitted the traffic at the source node, we make sure that MEC traffic is relayed smoothly at the relay node. Therefore, capacity is optimized at the relay node as explained in Section IV-B.

Traditionally, macro-cellular technologies employ proportional fair scheduling, that is, capacity is allocated according to the weight of the traffic load while Wi-Fi employees a throughput-based fairness model, that is, capacity is shared in way to give all the nodes equal throughput [49]. Therefore, the throughput of macro-cell and Wi-Fi is, respectively, given by

$$t_m = \frac{w_i \zeta_i}{\sum_n w_i} \tag{27}$$

$$t_{wf} = \frac{u}{\sum_n \frac{u w_i}{\zeta_i}} \tag{28}$$

here w_i is the weight of the user i. We, on the contrary, argue that capacity must be shared according to (26).

V. Managing Channel Variation

In Section III-B, we computed RAT performance in terms of delay from source node to destination and in Section IV, we showed optimal load distribution according to the obtained performance. However, as a result of change in channel condition, performance estimates obtained may become soon outdated and as a result, the load distribution and capacity optimization effectuated may not hold and the constraints may be violated. Confronted with such a situation, we have to allocate the load in such a way that the impact of the change in the performance is averted. Furthermore, we have to identify how frequent the RAT performance must be updated in order to reap the correct performance.

A. Frequency of RAT Performance Update

Given the temporal variation in a wireless channel, it is important to identify a suitable interval and frequency of RAT performance update, that is, how frequent should the RAT performance be updated to get the optimal performance? With larger interval between two consecutive performance updates, there is a possibility of decreasing performance due to stale information. Likewise, smaller intervals will result in sacrificing the bandwidth for network updates and making the task cumbersome. Performance of a RAT is subject to user mobility and network load in addition to the small-scale channel fading. Optimizing RAT update interval with respect to both instantaneous position and network load simultaneously is NP-hard and beyond the scope of this article. However, we strive to find a suitable interval that satisfies the performance of the network.

We propose a dynamic performance update interval as shown in Algorithm 2. The interval between two consecutive performance update varies according to the variation in the performance and will keep increasing as long as the variation (increase as well as decrease) in the performance is within the acceptable limit. Moreover, for data transmission at a particular instance, the performance of a RAT is estimated by taking the weighted moving average (WMA) of performance of last n seconds from the time the performance was last updated. Whenever the variation in the performance of a RAT is greater than certain threshold, the current data are transmitted using technique shown in Section V-B and performance is updated immediately. It may be apropos to mention that for WMA, we need at least two samples. Therefore, performance statistics are taken for next 2 s. Consequently, the minimum value of the loop counter n is 2.

Algorithm 2: RAT Performance Update

```
Input  : WMA, Time ($n$) and Performance
Output: Performance Update Interval
1 Update performance for $n = 2$ seconds.
2 Transmit current data.
3 Take WMA of last $n$ seconds.
4 while true do
5     $n \leftarrow n + 1$;
6     Go to Step 2
7 end
8 Go to Step 1
```

B. Managing Change in RAT Performance

Suppose there is a change in the performance of the RATs. Such a situation will lead to a violation of the constraints defined above unless the performance of RATs is updated. For example, with the change in channel condition, the delay of RATs will be different, thus violating the constraint defined in (7). Here, we attempt to temporarily reinstate the constraint before the performance of the RATs is updated in the next interval. This is carried out by adding certain amount of load to the faster RATs. Adding load will increase the delay of the faster RATs, thereby bringing them at par with slower RATs. This process is performed in three steps. In the first step, we determine how fast the faster RATS are with respect to the slowest RAT. Denote D_v as the delay of the fastest RAT, followed by D_t and D_r being the slowest among all the three. With this information given, the following holds true:

$$D_r = D_t - A = D_v - B. \tag{29}$$

Assuming D_r is the slowest RAT, D_t is faster than D_r by A μs (hence, A μs subtracted from it) while D_v is faster than D_r by B μs. With simple manipulation of (7), A and B can be obtained as follows:

$$A = \frac{1}{2\mu}\left(\frac{\zeta_t - \zeta_r}{\zeta_r \zeta_t}\right) + \left(\frac{\mu\zeta_t - \lambda_t - \mu\zeta_r + \lambda_r}{(\mu\zeta_t - \lambda_t)(\mu\zeta_r - \lambda_r)}\right) \tag{30}$$

$$B = \frac{1}{2\mu}\left(\frac{\zeta_v - \zeta_r}{\zeta_r \zeta_v}\right) + \left(\frac{\mu\zeta_v - \lambda_v - \mu\zeta_r + \lambda_r}{(\mu\zeta_v - \lambda_v)(\mu\zeta_r - \lambda_r)}\right). \tag{31}$$

In the second step, using (30), we derive (32) to get the equivalent load of A μs, that is, λ_A

$$\lambda_A = \frac{3\mu\zeta_A - 2A\mu^2\zeta_A^2}{1 - 2A\mu\zeta_A} \tag{32}$$

here ζ_A is the capacity of the RAT used with A which happens to be t as per (29). Similarly, we can derive expression for λ_B

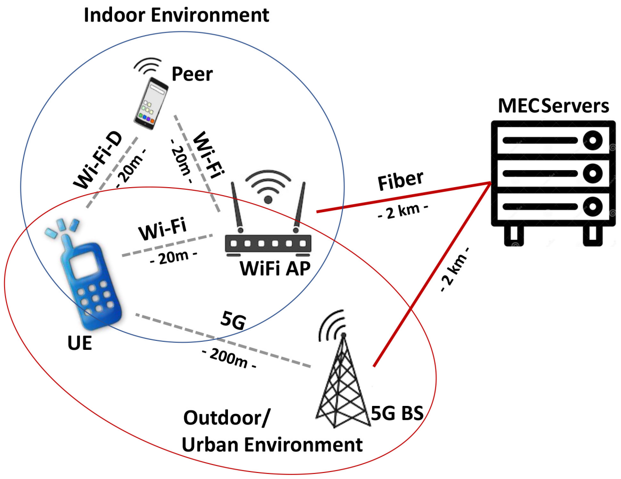

Fig. 2. Assumed topology where an end user inside a building is served by a peer device, Wi-Fi access point, and 5G macro-cell technology.

using (31) or simply by replacing A with B and capacity of RAT t with capacity of RAT v in (32).

Assuming traffic load is continuously being generated by the user, in the third step, we add λ_A and λ_B amount of load to their respective RATs, t and v, respectively, in this case, to make the delays equal. Adding loads λ_A and λ_B to RAT t and v, respectively, (29) will become

$$\begin{aligned} D_r &= D_t + \left\{\sum_{u=1}^{m}\left(\frac{1}{2\mu\zeta_u}+\frac{1}{\mu\zeta_u-\lambda_A}\right)+\sum_{u=1}^{m}\Theta_u\right\} \\ &= D_v + \left\{\sum_{u=1}^{m}\left(\frac{1}{2\mu\zeta_u}+\frac{1}{\mu\zeta_u-\lambda_B}\right)+\sum_{u=1}^{m}\Theta_u\right\}. \quad (33) \end{aligned}$$

The new delays of the three RATs are now equal. The change in RAT performance is incorporated.

VI. Performance Evaluation

In this section, we provide mathematical comparative analysis results to show the performance of our proposed scheme. We call our proposed scheme "multi-RAT parallel offloading (MPO)" and compare our performance with 5G, Wi-Fi, and Wi-Fi Direct and schemes that distribute the load on the basis of the tasks, such as [14] and [19]. For elaboration purposes, we refer to these schemes as atomic load distribution (ALD) schemes. We show how different RATs take different loads for their corresponding performance and compare their delay. We then compare the performance of our proposed scheme with Wi-Fi, 5G, and ALD. We also consider impact on the services of a newly arrived traffic when the node is busy serving the existing traffic in its queue. Finally, we show data outage probability comparison to verify service-level agreement (SLA) of MPO with Wi-Fi and 5G.

A. Environment Setting and Parameters

We consider the scenario shown in Fig. 2 where an end user is assumed to be based inside a multistory building. A Wi-Fi access point and a peer device are assumed to be inside the building while 5G macro-cell BS is assumed to be at a distance of 200 m in an urban environment. The end user is assumed to be simultaneously connected to a peer, Wi-Fi AP, and 5G BS. For Wi-Fi, we have used a frequency band of 5 GHz whereas for 5G, we have used 3.4-GHz band from Frequency Range 1 [50]. Similarly, EIRP for Wi-Fi is 30 and 43 dBm for 5G. Next, we describe how to compute different parameters in order to get performance measures of different RATs.

For WiFi, SNR is computed on the basis of indoor path-loss model as described in [51]. For ease of reference, we write the path-loss formula here

$$\mathrm{PL}_{wi} = \mathrm{PL}_{wi}(d_o) + 10\alpha\log\left(\frac{d_{wi}}{d_{wi,o}}\right) + \beta d \quad (34)$$

where PL_{wi} is the indoor path loss for Wi-Fi in dB, α is the path-loss exponent, β is specific attenuation, $\mathrm{PL}(d_{wi,o})$ is path loss at a reference distance which is taken to be 1 m. The values of both α and β are taken to be 2.

Similarly, for 5G, *SNR* is computed on the basis of the path-loss model given in [52] where macro-cell path loss is divided into two parts, that is, outdoor propagation loss and the building penetration loss. The outdoor propagation loss is given by

$$\mathrm{PL}_{mo} = 54 + 40\log d_{mo} - 30\log hb + 21\log f \quad (35)$$

where PL_{mo} is the outdoor path loss for macro-cell in *dB*, d_{mo} is the distance of user from macro-cell BS in meters, hb is the height of BS, and f is the frequency. The corresponding building penetration loss is given by [52]

$$\mathrm{PL}_{mi} = 0.6d_{mi} - 0.6h + 10 \quad (36)$$

where PL_{mi} is the loss in *dB* when the signal from the macro-cell BS penetrates the building, d_{mi} is the indoor distance of the user from the wall, and h is the height of the floor. The BER obtained on the basis of computed SNR is considered after LDPC code correction.

Data rate for Wi-Fi $R(w)$ is calculated as follows:

$$R(w) = M \cdot S \cdot R_c \cdot \frac{1}{T_s} \quad (37)$$

where M is the modulation scheme used, S is the number of subcarriers, R_c is the coding rate, and T_s is the symbol duration for Wi-Fi. Similarly, 5G data rate computation is based on the 3GPP TS 38.306 standard [53] and is given by

$$\begin{aligned} R(m) = 10^{-6} \cdot \sum_{j=1}^{J}\Bigg(& v_L^{(j)} \cdot Q_m^{(j)} \\ & \cdot f^{(j)} \cdot R_{\max} \cdot \frac{N_{\mathrm{PRB}}^{\mathrm{BW}(j),\psi} \cdot 12}{T_s^{\psi}} \cdot (1-\mathrm{OH})^{(j)}\Bigg) \quad (38) \end{aligned}$$

here J is the aggregated carrier component. In our case, we have not used carrier aggregation, therefore, its value is 1. $v_L^{(j)}$ is the number of streams. Again our computation is based on single-input–single-output signal, therefore, its value is 1. $Q_m^{(j)}$ is the modulation scheme used, $f^{(j)}$ is the scaling factor whose value we have taken to be 1. $R_{\max}$ is the coding rate, ψ is the numerology which defines the guard interval. Its value is 0 to 4 that corresponds to 15 kHz, 30 kHz, and so on up to

TABLE II
PARAMETERS SETTING

Technology	Wi-Fi (802.11ax)	5G
Distance	20 m	200 m
Bandwidth	80 MHz	100 MHz
Capacity	SNR Driven	
EIRP	30 dBm	43 dBm
Modulation	SNR Driven	
Code Rate	SNR Driven	
Frequency	5 GHz	3.4 GHz (FR-1)
α	2	-
β	2	-
Height of 5G Base Station	-	45 m
Height of Floor	-	10 m
Aggregated Carrier	1	1
Number of Streams	1	1
5G Numerology	-	1

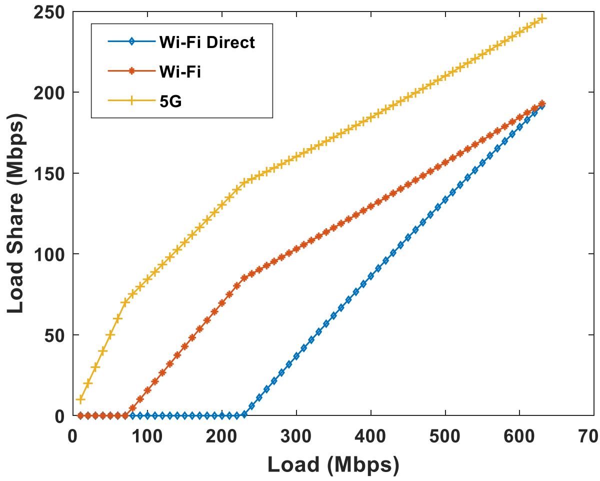

Fig. 3. Load shares assumed by different RATs as a result of increase in the incoming load.

120 kHz, respectively. We are using a bandwidth of 100 MHz for which the recommended guard interval is 30 kHz, therefore, its value will be 1 as 0 is not supported for 100 MHz according to the standard. T_s^{ψ} is the average symbol duration and is given by $(10^{-3}/14 \cdot 2^{\psi})$. The data rates obtained here are fed to (6) to get the capacities of the RATs. Based on the obtained capacities, loads distribution is carried out as described in Section IV-A.

The parameters used in computations are summarized in Table II.

B. Results

We begin with load distribution and system delay analysis of the proposed scheme where system delay is essentially network-wide packet delay. Fig. 3 shows the load each RAT will get for different load generated by the end user. 5G, for having higher bandwidth, has highest capacity among all the RATs and as a result has least delay as per (5), therefore the load share taken by 5G is the highest. On the other hand, the increase in load share with the increase in generated traffic for Wi-Fi Direct is highest. This is because the more the load is taken by a RAT, the sooner it will reach its saturation point. Therefore, to avoid saturation, more traffic is transferred to the RAT that has the lowest traffic load, which in this case happens to be Wi-Fi Direct.

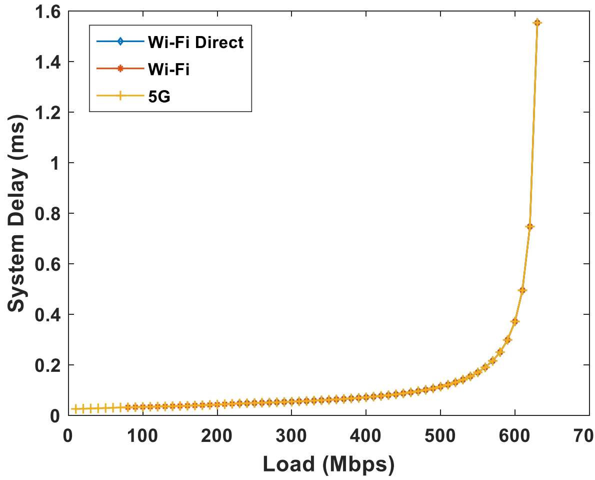

Fig. 4. Delay for different RATs as a result of increase in the incoming load.

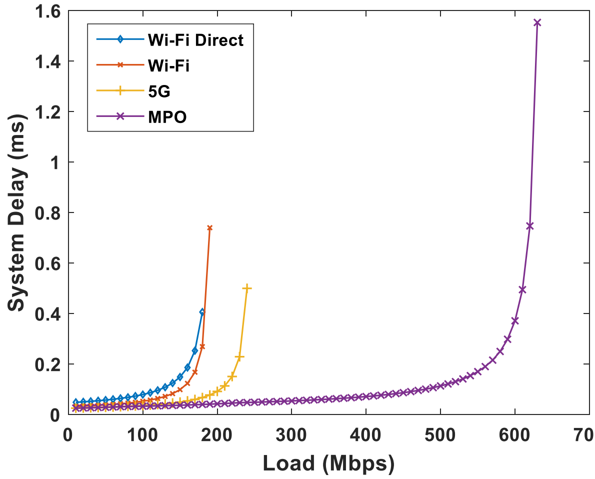

Fig. 5. Delay comparison of the proposed MPO schemes when data are offloaded through Wi-Fi alone, Wi-Fi Direct alone, and 5G alone.

We then analyze the delay for the corresponding load assumed by these RATs in Fig. 4. There are three curves in the figure which appear to be one single curve. The load shares assumed by different RATs are different as shown in Fig. 3, their delay, however, is equal. This is very important outcome of our proposal. We argued that packet reordering delay in multipath multi-RAT packet routing impedes the throughput significantly and is a major reason of real-time transmission missing the delay deadlines. However, with all the data packet reaching simultaneously, there will be no packet reordering delay. Another important outcome of the proposed scheme is the significantly high data that it can handle. The delay for up to 600 Mb is less than 0.1 ms, after which point it jumps to saturation point.

We also compare delay performance when data are offloaded through Wi-Fi Direct alone, Wi-Fi alone, and 5G alone, with MPO. As can be seen in Fig. 5, MPO outperforms Wi-Fi and 5G offloading in terms of the amount of data that they can carry. Both Wi-Fi Direct and Wi-Fi reach saturation before 200 Mb/s, 5G reaches saturation at slightly beyond

200 Mb/s whereas MPO, on the contrary, performs well all the way till 600 Mb/s, and the delay remains less than 0.1 ms for up to 600 Mb/s. This is a gain of about 70% as compared to Wi-Fi Direct and Wi-Fi, and about 63% as compared to 5G.

Next, considering the prioritized processing of MEC traffic, it is inevitable that the conventional traffic will experience repercussions. Likewise, when a node's capacity is already extensively utilized, the QoS for incoming traffic will inevitably be affected. Therefore, to quantify the impact on the QoS of the newly arrived data traffic, we propose a novel metric termed Relative QoS (RQoS), which aims to quantify the impact of QoS on the newly arrived data traffic while the nodes are concurrently processing existing data in their queues. RQoS is a relative metric that measures the impact relative to the prior load on the node. Specifically, we assess the RQoS (I) provided to the newly arrived data in relation to the prior load on the node.

Let $\zeta_{r,cr}$ be the current capacity of a certain RAT r. Suppose the newly arrived normalized load at time t requires the capacity $\zeta_{r,req}$ for time Δt seconds. The RQoS I of the newly arrived data will be

$$I = \sum_{r=1}^{v} \left(\frac{\int_t^{\Delta t} \zeta_{r,req} - \int_t^{\Delta t} \zeta_{r,cr}}{\int_t^{\Delta t} \zeta_{r,req}} \right). \tag{39}$$

In Fig. 11(b), the RQoS of newly arrived data packets is depicted in relation to the existing data at the node. The results are obtained for an average incoming load of 150 Mb/s over a duration of 10 ms. The vertical axis represents the variation in RQoS with respect to the current or existing load at the node, as indicated on the x-axis. A value of 1 signifies no impact on the service, while a value of 0 implies that the new packets will not receive any service. Therefore, a higher RQoS value indicates a lesser impact on the services. Observing the graph, it can be observed that up to 50 Mb/s of the existing load, no RAT experiences any impact. However, beyond this point, Wi-Fi shows a decline in service quality as the existing load increases, while 5G demonstrates a decline after reaching 65 Mb/s of existing data. On the other hand, Mobile MPO consistently maintains a satisfactory level of service until reaching a load of 130 Mb/s, beyond which there is a decline in quality. Notably, MPO exhibits a 61% improvement compared to Wi-Fi and a 50% improvement compared to 5G. At a load of 150 Mb/s, Wi-Fi experiences a 55% impact on service, resulting in a 45% decline in quality. Similarly, 5G shows a 70% impact, corresponding to a 30% decline in quality. In contrast, MPO maintains a service level of over 90%. The QoS for newly arrived data packets is minimally affected by MPO, highlighting its superior performance.

We also compare the packet outage probability of the three schemes. The knowledge of packet outage probability is useful for verifying SLA compliance. Packet outage probability is linked to the probability of load getting greater than a given threshold, as for certain load, $\lambda > \zeta$, there will be outages in packets. When outages are greater than certain threshold, the SLA terms will be violated. For Poisson packet arrival, the probability of load getting greater than capacity is given by

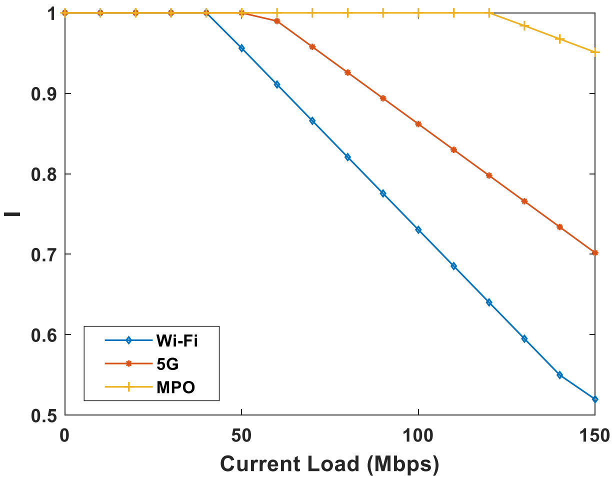

Fig. 6. Impact on the services of different RATs.

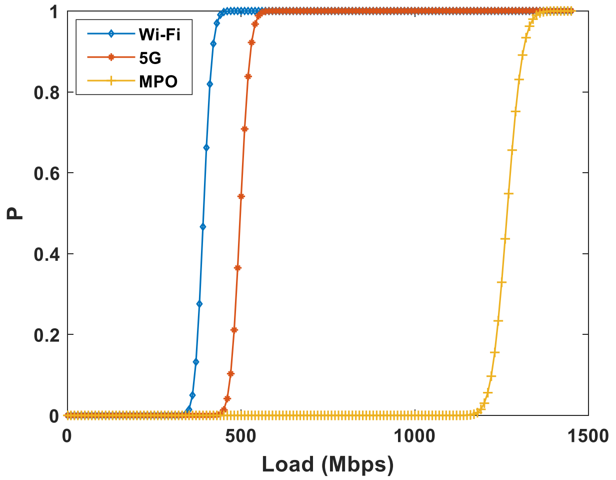

Fig. 7. Outage probability comparison of different schemes to show SLA conformity.

$$P(\lambda > \zeta) = 1 - \left(e^{-\rho} \sum_{i=0}^{k} \frac{\rho^i}{i!} \right) \tag{40}$$

where $(e^{-\rho} \sum_{i=0}^{k} \frac{\rho^i}{i!})$ is the cumulative distribution function (CDF) of the Poisson distribution. In Fig. 7, we present a comparison of the probability of load exceeding capacity for Wi-Fi, 5G, and MPO to assess the occurrence of packet outage. The vertical axis represents the probability of load surpassing capacity, while the x-axis represents the incoming traffic load. The three schemes exhibit full compliance with the SLA, ensuring no packet loss, up to approximately 350 Mb/s. At this point, the probability for Wi-Fi starts to decline and reaches 0 at around 450 Mb/s. For 5G, the probability is affected after approximately 450 Mb/s and also reaches 0 at 580 Mb/s. Conversely, MPO maintains a consistent probability of 1 until approximately 1100 Mb/s. This corresponds to a significant gain of 67% compared to Wi-Fi and a 58% gain compared to 5G in terms of packet outage probability.

So far we showed the gain in performance by using multiple radio resources. Next, we compare performance of our proposed MLO scheme with ALD that distributes the load

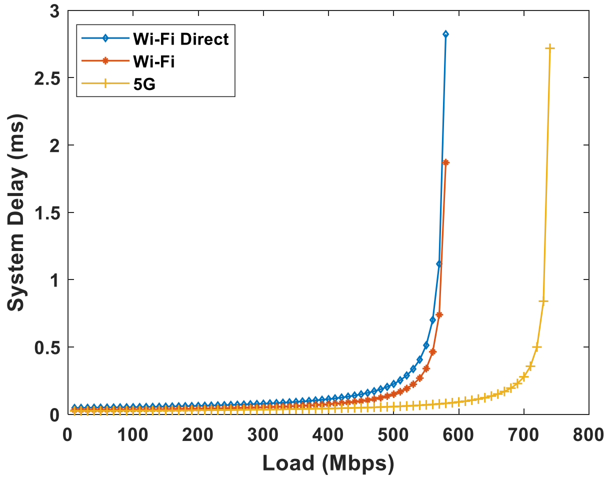

Fig. 8. System delay of different RATs for ALD.

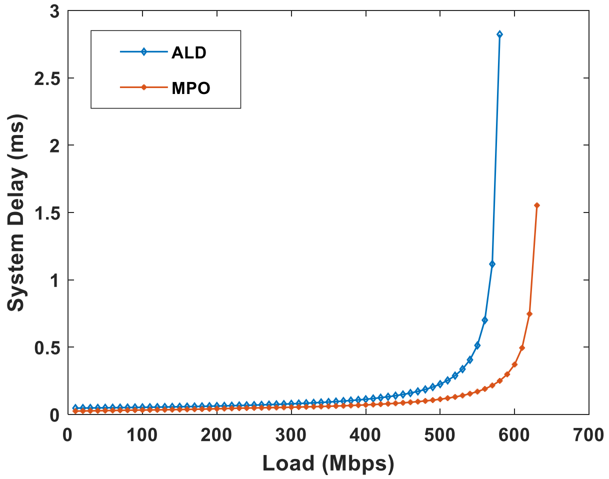

Fig. 9. System delay comparison of MPO with ALD when capacity distribution at relay is optimized for both the schemes.

on the basis of the task rather than the load itself. Considering a task size of 800 kB, the delay of different RATs for ALD is shown in Fig. 8. As can be seen, different RATs have different delays. Now, this is a major bottleneck as applications depend on the reception of all three tasks in order to provide seamless services to the end users. Therefore, all data packets must arrive at the transport protocol layer in sequence whereas data arrived out-of-order is either kept in buffer or totally discarded depending upon the magnitude of latency of slower RATs.

In Fig. 9, we thus compare the system delay of the proposed MPO with ALD. Here, we can see that the performance of ALD is limited by a slower RAT whereas the proposed MPO scheme apportions loads according to the performance of the RATs by virtue of which a slower RAT receives a lower load and thus its effect on performance is minimized. As can be seen, for the given scenario, the proposed MPO scheme carry approximately 80 Mb/s more load in comparison with ALD. ALD is saturated at the offered load of about 550 Mb/s whereas MPO can carry a load up to 630 Mb/s. Similarly, MPO has consistently lesser system delay in comparison to ALD. The higher system delay of ALD is contributed by higher load share allocated to slower RAT which happens to be Wi-Fi Direct in this case.

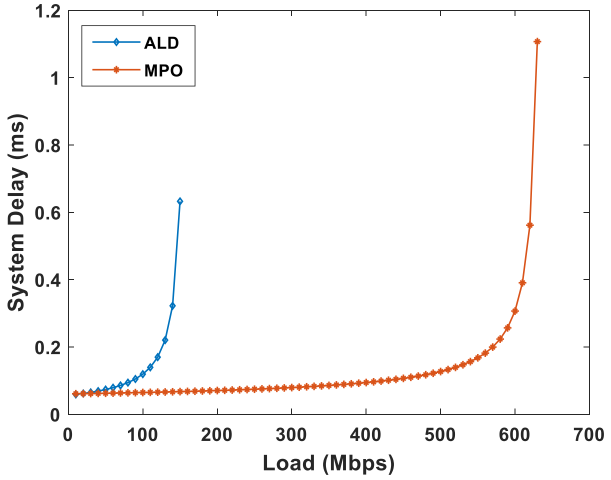

Fig. 10. System delay comparison of MPO with ALD when capacity distribution at relay is optimized for MPO only.

We also analyze the performance of our proposed capacity optimization technique at relay node. Until now, we have compared ALD and MPO with both schemes having capacity optimized according to the load, because the goal was to show system performance in terms of load distribution under the same parameters. Here, we compare the performance of the proposed optimal capacity distribution scheme against conventional technique where data at relay node is forwarded with even capacity or nonoptimal capacity distribution among the links.

In Fig. 10, we plot the system delay for four users with 40% MEC traffic. Thus, the load on the x-axis indicates 60% conventional and 40% MEC traffic for four users operating simultaneously. It is clear from Fig. 10 that proposed MPO with optimized capacity distribution can support nearly four times the maximum load that ALD can while giving lower system delay. With the same total capacity, ALD reaches its saturation point at about 150 Mb/s while MPO, intelligently distributing the capacity according to the load, maintains a stable delay until 600 Mb/s.

The delay and the load shared by RATs against the incoming load are not linear. If there is existing load at the node, the incoming traffic will incur more delay and accordingly the load shares will be different as discussed in Section III. Therefore, we have shown the impact of current or existing delay at node on delay and load share in Fig. 11. In Fig. 11(a) and (b), we have compared load share and delay for different RATs for MPO. We have also compared the delay of the proposed MPO with ALD in Fig. 11(c), that linearly distributes the load on the basis of the tasks. The figure shows that the proposed MPO has consistently lower delay than ALD.

VII. Conclusion and Future Work

In this work, we developed a technique that optimally utilizes the capacity at source node and optimally distributes the available capacity among the links at relay node. We

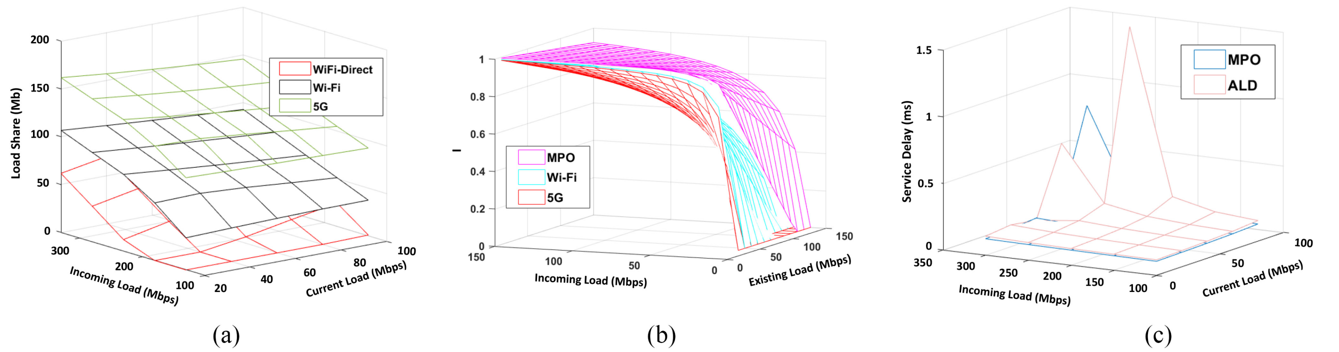

Fig. 11. Impact of current load on different parameters. (a) Impact on load share. (b) Impact on services on incoming data. (c) Impact on delay.

considered the performance of all the RATs and distributed the traffic among the RATs in such a way that delay for all the RATS is equalized, thereby avoiding the packet reordering delay at the destination node. As a proof of concept, we showed that to minimize system delay and maximize throughput, QoS, and SLA compliance, we must optimize capacity utilization at the source node and capacity distribution on the outgoing links at relay nodes. Our numerical results demonstrated that our proposed technique fares better than contemporary techniques that distribute the data on the basis of the number of tasks. We believe that our proposed technique for simultaneous offloading over multiple RATs will not only improve MEC performance for future applications but is also a plausible mechanism to make up for the debilitated telecom infrastructure in low- and middle-income countries.

The utilization of multiple RATs results in increased energy consumption in the system. In order to mitigate this issue, we plan to introduce an energy consumption model by analyzing and optimizing the energy consumption with respect to the load in future work. Additionally, a more comprehensive analysis of SLA compliance will be conducted, encompassing additional parameters like service delay and QoS. Moreover, we aim to enhance the accuracy of estimating the instantaneous capacity and integrate the UE mobility model, which affects the SNR and channel condition. Finally, the use of multiple RATs can augment the attack surface of the UE, rendering it more susceptible to security breaches. Thus, research can also be conducted to address the security challenges of a multi-RAT system.

Appendix A
Proof of Convexity of the Formulated Problem

Considering that the objective function for delay minimization problem is nonlinear, we need to verify that any solution we find is a correct global minimum solution. Therefore, in this section, we attempt to proof convexity of the delay minimization problem to confirm that the local optima is also the global optima.

Theorem 1: The delay minimization function given in (5) is a convex function.

Corollary 1: If $f(x)$, where $x \in \mathbb{R}$ is a convex function, $f(x) + w$ is also a convex function, where w is a positive real number.

Corollary 2: If $d_u = ([1/2\mu\zeta_u] + [1/\mu\zeta_u - \lambda_u] + \Theta)$ is true for one link u, it is true for all $u = 1$ to n links.

Proof: Following Corollary 1, we ignore propagation delay and draw Hessian matrix for all links, all RATs

$$H(f) = \begin{pmatrix} \frac{\partial^2 f}{\partial D_1^2} & \frac{\partial^2 f}{\partial D_1 \partial D_2} & \cdots & \frac{\partial^2 f}{\partial D_1 \partial D_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial D_n \partial D_1} & \frac{\partial^2 f}{\partial D_n \partial D_2} & \cdots & \frac{\partial^2 f}{\partial D_n^2} \end{pmatrix}. \tag{42}$$

To make the computation simple, let us draw Hessian matrix for single link only, without loss of generality, as allowed by Corollary 2. The resultant matrix is given in (41), shown at the bottom of the page. We began with calculating Eigenvalues of the matrix and then putting the smallest and largest possible values for all the variables in the Eigenvalues. The results were positive for both the minimum and maximum values, indicating the function being convex. However, we duly prove its convexity through principle minor technique. The resultant matrix given in (41) is a 3×3 matrix which implies that there will be three orders of principal minors where the first-order leading principal minor P_1 is obtained by deleting the last two rows and columns of the matrix, that is

$$P_1 = \frac{1}{\mu^3\zeta} + \frac{2\zeta^2}{(\mu\zeta - \lambda)^3}. \tag{43}$$

Examining (43), we see that none of the terms is negative here. Therefore, P_1 is greater than 0. Note that $(\mu\zeta - \lambda)$ is a very large positive number. Similarly, we find second-order leading principal minor P_2. P_2 will be the determinant

$$H(d_u) = \begin{pmatrix} \frac{1}{\mu^3\zeta} + \frac{2\zeta^2}{(\mu\zeta-\lambda)^3} & \frac{1}{2\mu^2\zeta^2} + \frac{2\mu\zeta}{(\mu\zeta-\lambda)^3} - \frac{1}{(\mu\zeta-\lambda)^2} & -\frac{2\zeta}{(\mu\zeta-\lambda)^3} \\ \frac{1}{2\mu^2\zeta^2} + \frac{2\mu\zeta}{(\mu\zeta-\lambda)^3} - \frac{1}{(\mu\zeta-\lambda)^2} & \frac{2\mu^2}{(\mu\zeta-\lambda)^3} + \frac{1}{\mu\zeta^3} & -\frac{2\mu}{(\mu\zeta-\lambda)^3} \\ -\frac{2\zeta}{(\mu\zeta-\lambda)^3} & -\frac{2\mu}{(\mu\zeta-\lambda)^3} & \frac{2}{(\mu\zeta-\lambda)^3} \end{pmatrix} \tag{41}$$

of matrix obtained by deleting last row and column from $H(d_u)$

$$\begin{aligned}P_2 = &-\frac{1}{4\mu^4\zeta^4} + \frac{1}{\mu^4\zeta^4} - \frac{4\mu\zeta^2}{(\mu\zeta-\lambda)^6} + \frac{4\mu^2\zeta^2}{(\mu\zeta-\lambda)^6}\\ &+ \frac{4\mu\zeta}{(\mu\zeta-\lambda)^5} - \frac{1}{(\mu\zeta-\lambda)^4}\\ &- \frac{2\mu\zeta}{\mu^2\zeta^2(\mu\zeta-\lambda)^3} + \frac{2}{\mu\zeta(\mu\zeta-\lambda)^3}\\ &+ \frac{2\zeta^2}{\mu\zeta^3(\mu\zeta-\lambda)^3} + \frac{1}{\mu^2\zeta^2(\mu\zeta-\lambda)^2}.\end{aligned} \tag{44}$$

Examining (44), there are ten terms. The result of the first two terms will be positive as second terms is greater than the first one. The result of the 3rd and 4th terms is also positive as 4th terms is greater than 3rd. The result of 5th and 6th terms will be again positive as the 5th term is greater than 6th. Finally, 7th terms is smaller than 8th + 9th + 10th. Therefore, the net result of these four terms will be positive which implies that overall P_2 is positive. Next, we move to third-order principal minor P_3 which is the determinant of the Hessian Matrix itself and is given by

$$P_3 = \frac{\frac{3(\lambda-\mu\zeta)^4}{\mu^4\zeta^4} + \frac{4(\lambda-\mu\zeta)^2}{\mu^2\zeta^2} - 4}{2(\mu\zeta-\lambda)^7}. \tag{45}$$

Again, in (45), the only negative term here is 4. However, the first two terms in numerator are greater than 4 due to which net result of numerator will be positive. Therefore, it is safe to say P_3 is also positive. From the net results of P_1, P_2, and P_3, we can say that first-, second-, and third-order leading principal minors are all positive. Therefore, we can say that the resultant Hessian matrix of the function is positive definite. From this, we conclude that the delay minimization function is convex. ■

Appendix B
Proof of Obtained Load Shares Being Optimal

Here, we prove λ_r, λ_t, and λ_t to be optimal loads that utilize available capacity optimally.

Theorem 2: λ_r, λ_t, and λ_t are the optimal load shares.

Proof: Using proof by contradiction, let us assume λ_r, λ_t, and λ_v are not optimum and instead x, y, and z are the optimal load shares. Therefore, we attempt to optimize these quantities by extending the Nash Bargaining theorem [54] to three players as follows:

$$\text{maximize } J = (\lambda_r - x)(\lambda_t - y)(\lambda_v - z). \tag{46}$$

Taking $(\partial J/\partial \lambda_i)$ with respect to $\lambda_i = \lambda_r$, λ_t, and λ_v and equaling to 0, we get

$$0 = \lambda_t\lambda_v - z\lambda_t - y\lambda_v + yz \tag{47}$$

$$0 = \lambda_r\lambda_v - z\lambda_r - x\lambda_v + xz \tag{48}$$

$$0 = \lambda_r\lambda_t - y\lambda_r - x\lambda_t + xy. \tag{49}$$

Solving (47)–(49) for λ_r, λ_t, and λ_v, the quantities remain unchanged, substantiating the fact that the quantities are optimum. This contradicts our assumption and hence proves the theorem. ■

Appendix C
Proof of Capacity at Relay Node Being Optimal

In this section, we prove that the obtained capacity in (26), on the basis of the Lagrange theorem defined in (25), is optimal. We argue that ζ_u for link u is optimal for certain $\zeta_t = \sum_{u=1}^{n} \zeta_u$.

We used the Lagrange multiplier theorem [48], on that account let the original function be $f(x, y)$ and let for the sake of simplicity $g(x, y) = \sum_{u=1}^{n} \zeta_u - \zeta_t$ and $g(x, y) = 0$, but $\nabla g \neq 0$, without loss of generality $(\partial g/\partial y) \neq 0$. Writing (25) in its standard form, we get

$$W = f(x, y, k) = f(x, y) - k(g(x, y)) \tag{50}$$

where $W = f(x, y, k)$ is the new function obtained as a result of incorporating multiplier k.

The Lagrange multiplier theorem is based on the implicit function theorem (IFT). Therefore, by *IFT*, we can assume that there is a function $y = y(x)$ such that $g(x, y(x)) = 0$ which follows that $f(x, y(x))$. Furthermore, using the same theorem, we have

$$y'(x) = -\frac{g_x}{g_y}. \tag{51}$$

Since $f(x, y(x))$ is assumed to be optimal, its derivative has to be 0. Using the chain rule, we have

$$f_x + f_y \cdot y'(x) = 0. \tag{52}$$

Equation (52) shows an optimal value. Next, we have to show that this optimal value is equal to the value of the original equation (50).

Using (51), we get

$$f_x - f_y \cdot \frac{g_x}{g_y} = 0. \tag{53}$$

Let $-k$ denote f_y/g_y

$$f_y + kg_y = 0. \tag{54}$$

Using (53), we get

$$f_x + kg_x = 0. \tag{55}$$

Equation (55) shows that the gradient of $(f + kg)$ at points defined by constraint is 0. Also, following (52), (55) shows that original value defined by function in (50) is optimal, hence our capacity is optimal.

Appendix D
Assigning Load Shares to RATs

We formulate an integer linear program to assign load shares to the RATs. Let delay for the three load shares λ_r, λ_t, and λ_u over RAT r be $d_{r,r}$, $d_{t,r}$, and $d_{u,r}$, delay for the same load shares over RAT t be $d_{r,t}$, $d_{t,t}$, and $d_{u,t}$. Similarly, delay for these load share over RAT u be $d_{r,u}$, $d_{t,u}$, and $d_{u,u}$, as shown in Table III.

Before formulating the integer linear program, let us define a binary variable $x_{i,j}$ such that

$$x_{i,j} = \begin{cases} 1, & \text{if load } i \text{ is assigned to RAT } j \\ 0, & \text{otherwise.} \end{cases}$$

TABLE III
DELAY OF THE OBTAINED LOAD SHARES ON DIFFERENT RATS

	RAT_r	RAT_t	RAT_u
λ_r	$d_{r,r}$	$d_{v,r}$	$d_{t,r}$
λ_t	$d_{r,t}$	$d_{t,t}$	$d_{v,t}$
λ_v	$d_{r,v}$	$d_{t,v}$	$d_{v,v}$

Similarly for load share λ_r over RAT u, the assignment variable will be $x_{r,u}$ and its value will be 0 or 1 depending upon whether or not λ_r is assigned to u. Next, we formulate our integer linear program as follows:

$$\text{minimize} \begin{pmatrix} d_{r,r} \cdot x_{r,r} + d_{v,r} \cdot x_{r,t} + d_{t,r} \cdot x_{r,v} + \\ d_{r,t} \cdot x_{t,r} + d_{t,t} \cdot x_{t,t} + d_{v,t} \cdot x_{t,v} + \\ d_{r,v} \cdot x_{v,r} + d_{t,v} \cdot x_{v,t} + d_{v,v} \cdot x_{v,v} \end{pmatrix}$$

Subject to

$$x_{r,r} + x_{r,t} + x_{r,v} = 1 \tag{56}$$

$$x_{t,r} + x_{t,t} + x_{t,v} = 1 \tag{57}$$

$$x_{v,r} + x_{v,t} + x_{v,v} = 1 \tag{58}$$

$$x_{r,r} + x_{t,r} + x_{v,r} = 1 \tag{59}$$

$$x_{r,t} + x_{t,t} + x_{v,t} = 1 \tag{60}$$

$$x_{r,v} + x_{t,v} + x_{v,v} = 1 \tag{61}$$

$$x_{i,j} \geq 0. \tag{62}$$

The objective functions say to minimize the total delay of the three RATs when loads are assigned to them. The decision variable 0s indicate that if a load is not assigned, its value will be zero, that is, the corresponding load will not go to the RAT where the value is 0. Constraints (56)–(61) show that loads are assigned to one single RAT only and one RAT will get one share of load only. No two loads can go to a single RAT conversely, no RAT can be assigned more than one load share. Finally, (62) is the positivity constraint. We solved the integer linear program with the simplex method.

REFERENCES

[1] X. Jiang, F. R. Yu, T. Song, and V. C. M. Leung, "A survey on multi-access edge computing applied to video streaming: Some research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 871–903, 2nd Quart., 2021.

[2] M. Z. Chowdhury, M. Shahjalal, S. Ahmed, and Y. M. Jang, "6G wireless communication systems: Applications, requirements, technologies, challenges, and research directions," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 957–975, 2020.

[3] C. D. Alwis et al., "Survey on 6G frontiers: Trends, applications, requirements, technologies and future research," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 836–886, 2021.

[4] S. Dang, O. Amin, B. Shihada, and M.-S. Alouini, "What should 6G be?" *Nat. Electron.*, vol. 3, no. 1, pp. 20–29, 2020.

[5] L. Zhao, G. Zhou, G. Zheng, I. Chih-Lin, X. You, and L. Hanzo, "Open-source multi-access edge computing for 6G: Opportunities and challenges," *IEEE Access*, vol. 9, pp. 158426–158439, 2021.

[6] J. R. Bhat and S. A. Al-Qahtani, "6G ecosystem: Current status and future perspective," *IEEE Access*, vol. 9, pp. 1–34, 2021.

[7] N. Shlezinger, G. C. Alexandropoulos, M. F. Imani, Y. C. Eldar, and D. R. Smith, "Dynamic metasurface antennas for 6G extreme massive MIMO communications," *IEEE Wireless Commun.*, vol. 28, no. 2, pp. 106–113, Apr. 2021.

[8] D. Pinchera, M. D. Migliore, and F. Schettino, "Optimizing antenna arrays for spatial multiplexing: Towards 6G systems," *IEEE Access*, vol. 9, pp. 53276–53291, 2021.

[9] A. Ali and F. A. Khan, "Condition and location-aware channel switching scheme for multi-hop multi-band WLANs," *Comput. Netw.*, vol. 168, Feb. 2020, Art. no. 107048.

[10] A. Ali, F. Hussain, R. Hussain, A. M. Khan, and A. Ferworn, "Multiband multi-hop WLANs for disaster relief and public safety applications," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops (WCNCW)*, 2020, pp. 1–6.

[11] S. Lee, T. Kim, S. Lee, K. Kim, Y. H. Kim, and N. Golmie, "Dynamic channel bonding algorithm for densely deployed 802.11ac networks," *IEEE Trans. Commun.*, vol. 67, no. 12, pp. 8517–8531, Dec. 2019.

[12] A. Ali and K. Ali, "Delay sensitive routing algorithm," *J. High Speed Netw.*, vol. 20, no. 4, pp. 253–262, 2014.

[13] J. Xia et al., "Opportunistic access point selection for mobile edge computing networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 695–709, Jan. 2021.

[14] T. Braud, P. Zhou, J. Kangasharju, and P. Hui, "Multipath computation offloading for mobile augmented reality," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, 2020, pp. 1–10.

[15] G. S. Gaba, G. Kumar, T.-H. Kim, H. Monga, and P. Kumar, "Secure device-to-device communications for 5G enabled Internet of Things applications," *Comput. Commun.*, vol. 169, pp. 114–128, Mar. 2021.

[16] H. Ko and S. Pack, "Distributed device-to-device offloading system: Design and performance optimization," *IEEE Trans. Mobile Comput.*, vol. 20, no. 10, pp. 2949–2960, Oct. 2021.

[17] S. Forge and K. Vu, "Forming a 5G strategy for developing countries: A note for policy makers," *Telecommun. Policy*, vol. 44, no. 7, 2020, Art. no. 101975.

[18] S. Song, J. Jung, M. Choi, C. Lee, J. Sun, and J. Chung, "Multipath based adaptive concurrent transfer for real-time video streaming over 5G multi-RAT systems," *IEEE Access*, vol. 7, pp. 146470–146479, 2019.

[19] Z. Jing, Q. Yang, M. Qin, J. Li, and K. S. Kwak, "Long-term max-min fairness guarantee mechanism for integrated multi-RAT and MEC networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2478–2492, Mar. 2021.

[20] K. C.-J. Lin, H.-C. Wang, Y.-C. Lai, and Y.-D. Lin, "Communication and computation offloading for multi-RAT mobile edge computing," *IEEE Wireless Commun.*, vol. 26, no. 6, pp. 180–186, Dec. 2019.

[21] M. Qin et al., "Service-oriented energy-latency tradeoff for IoT task partial offloading in MEC-enhanced multi-RAT networks," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1896–1907, Feb. 2021.

[22] J. Licklider, *Memorandum for: Members and Affiliates of the Intergalactic Computer Network*, 1963. Accessed: Jun. 25, 2021. [Online]. Available: http://shannon.usu.edu.ru/Papers/Lick/

[23] A. Mukhopadhyay and M. Ruffini, "Learning automata for multi-access edge computing server allocation with minimal service migration," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2020, pp. 1–6.

[24] P. Porambage, J. Okwuibe, M. Liyanage, M. Ylianttila, and T. Taleb, "Survey on multi-access edge computing for Internet of Things realization," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2961–2991, 4th Quart., 2018.

[25] Q. Pham et al., "A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art," *IEEE Access*, vol. 8, pp. 116974–117017, 2020.

[26] H. Lin, S. Zeadally, Z. Chen, H. Labiod, and L. Wang, "A survey on computation offloading modeling for edge computing," *J. Netw. Comput. Appl.*, vol. 169, Nov. 2020, Art. no. 102781.

[27] A. Samanta and Y. Li, "Latency-oblivious incentive service offloading in mobile edge computing," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, 2018, pp. 351–353.

[28] D. Zhao, T. Yang, Y. Jin, and Y. Xu, "A service migration strategy based on multiple attribute decision in mobile edge computing," in *Proc. IEEE 17th Int. Conf. Commun. Technol. (ICCT)*, 2017, pp. 986–990.

[29] A. Bozorgchenani, S. Maghsudi, D. Tarchi, and E. Hossain, "Computation offloading in heterogeneous vehicular edge networks: Online and off-policy bandit solutions," *IEEE Trans. Mobile Comput.*, vol. 21, no. 12, pp. 4233–4248, Dec. 2022.

[30] T. Alfakih, 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.

[31] 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. 2022.

[32] T. X. Tran and D. Pompili, "Joint task offloading and resource allocation for multi-server mobile-edge computing networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 856–868, Jan. 2019.

[33] Z. Yang, B. Liang, and W. Ji, "An intelligent end–edge–cloud architecture for visual IoT assisted healthcare systems," *IEEE Internet Things J.*, vol. 8, no. 23, pp. 16779–16786, Dec. 2021.

[34] X. Cao, F. Wang, J. Xu, R. Zhang, and S. Cui, "Joint computation and communication cooperation for energy-efficient mobile edge computing," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4188–4200, Jun. 2019.

[35] Z. Ning, P. Dong, X. Kong, and F. Xia, "A cooperative partial computation offloading scheme for mobile edge computing enabled Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4804–4814, Jun. 2019.

[36] Y.-C. Wu, T. Q. Dinh, Y. Fu, C. Lin, and T. Q. S. Quek, "A hybrid DQN and optimization approach for strategy and resource allocation in MEC networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 7, pp. 4282–4295, Jul. 2021.

[37] A. Bozorgchenani, F. Mashhadi, D. Tarchi, and S. A. S. Monroy, "Multi-objective computation sharing in energy and delay constrained mobile edge computing environments," *IEEE Trans. Mobile Comput.*, vol. 20, no. 10, pp. 2992–3005, Oct. 2021.

[38] Z. Qin et al., "Task selection and scheduling in UAV-enabled MEC for reconnaissance with time-varying priorities," *IEEE Internet Things J.*, vol. 8, no. 24, pp. 17290–17307, Dec. 2021.

[39] R. Bajracharya, R. Shrestha, Y. B. Zikria, and S. W. Kim, "LTE in the unlicensed spectrum: A survey," *IETE Tech. Rev.*, vol. 35, no. 1, pp. 78–90, 2018.

[40] F. Tian, Y. Yu, X. Yuan, B. Lyu, and G. Gui, "Predicted decoupling for coexistence between WiFi and LTE in unlicensed band," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4130–4141, Apr. 2020.

[41] Z. Xu et al., "Online learning algorithms for offloading augmented reality requests with uncertain demands in MECs," in *Proc. IEEE 41st Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2021, pp. 1064–1074.

[42] A. Ali, K. Ali, and A. A. Shaikh, "Energy and delay aware routing algorithm for fiber-wireless networks," *Wireless Netw.*, vol. 20, no. 6, pp. 1313–1320, Aug. 2014.

[43] L. Kleinrock, *Queuing systems*. New York, NY, USA: Wiley, 1975. [Online]. Available: https://cds.cern.ch/record/103535

[44] A. Ali, I. Ullah, T. Taqueer, and S. M. H. Zaidi, "Performance enhancement of WLANs," in *Proc. Intl. Conf. High Capacity Opt. Netw. Emerg. Technol.*, Dec. 2011, pp. 148–152.

[45] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi, "A tutorial on IEEE 802.11ax high efficiency WLANs," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 197–216, 1st Quart., 2019.

[46] W.-J. Lee, W. Shin, J. A. Ruiz-de Azua, L. F. Capon, H. Park, and J.-H. Kim, "NOMA-based uplink OFDMA collision reduction in 802.11ax networks," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, 2021, pp. 212–214.

[47] A. Ali, "Combine and conquer: Simultaneous transmission over multiband multi-hop WLAN systems," *IEEE Access*, vol. 9, pp. 27496–27509, 2021.

[48] E. J. McShane, "The Lagrange multiplier rule," *Amer. Math. Month.*, vol. 80, no. 8, pp. 922–925, 1973.

[49] R. Mahindra, H. Viswanathan, K. Sundaresan, M. Y. Arslan, and S. Rangarajan, "A practical traffic management system for integrated LTE-WiFi networks," in *Proc. MobiCom*, New York, NY, USA, 2014, pp. 189–200.

[50] "Global updates on spectrum for 4G/5G." Qualcomm. 2020. Accessed: Jun. 25, 2021. [Online]. Available: https://www.qualcomm.com/media/documents/files/spectrum-for-4g-and-5g.pdf

[51] V. Degli-Esposti, G. Falciasecca, F. Fuschini, and E. M. Vitucci, "A meaningful indoor path-loss formula," *IEEE Antennas Wireless Propag. Lett.*, vol. 12, pp. 872–875, 2013.

[52] H. Okamoto, K. Kitao, and S. Ichitsubo, "Outdoor-to-indoor propagation loss prediction in 800-MHz to 8-GHz band for an urban area," *IEEE Trans. Veh. Technol.*, vol. 58, no. 3, pp. 1059–1067, Mar. 2009.

[53] *Technical Specification Group Radio Access Network; User Equipment (UE) Radio Access Capabilities*, 3GPP Standard 38.306, Dec. 2017. Accessed: Jun. 25, 2021. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3193

[54] J. F. Nash Jr., "The bargaining problem," *Econometrica*, vol. 18, no. 2, pp. 155–162, Apr. 1950.

Asad Ali received the M.S. degree in applied information sciences from the Graduate School of Information Sciences, Tohoku University, Sendai, Japan, in 2019.

He is currently an ADVANCE-CRT Scholar with the CONNECT Research Centre, School of Computer Science and Statistics, Trinity College, The University of Dublin, Dublin, Ireland, where his studies are being sponsored by the Science Foundation Ireland. He has over 30 research papers published in international journals of repute and refereed conferences. His research interests are in multiradio transmission and network optimization.

Mr. Ali was awarded the prestigious MEXT Scholarship by the Government of Japan to support his studies in the country. He has received number of awards, research, and travel grants for his research work.

Kanza Ali received the M.S. degree in system information sciences from the Graduate School of Information Sciences, Tohoku University, Sendai, Japan, in 2019, where she is currently pursuing the Ph.D. degree supported by the prestigious MEXT Scholarship provided by the Government of Japan.

Her research endeavors primarily focus on the applications of deep learning in computer vision and computer networks.

Ms. Ali has been honored with multiple awards and research grants in recognition of her exceptional academic achievements.