

Received November 17, 2020, accepted November 30, 2020, date of publication December 4, 2020, date of current version December 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3042522

Client-Centric Access Device Selection for Heterogeneous QoS Requirements in Beyond 5G IoT Networks

MUHAMMAD ASAD¹, (Graduate Student Member, IEEE),
SAAD QAISAR^{1,2}, (Senior Member, IEEE), AND ABDUL BASIT¹

¹School of Electrical Engineering and Computer Science, National University of Sciences and Technology, Islamabad 44000, Pakistan

²Department of Electrical Engineering, University of Jeddah, Jeddah 23218, Saudi Arabia

Corresponding author: Muhammad Asad (14phdmasad@seecs.edu.pk)

ABSTRACT The expected manifolds increase in the traffic and the massive number of connected Internet of Things (IoT) devices would be a challenge to ensure a certain Quality of Service (QoS) in Beyond 5G (B5G) networks. Due to the increased availability of processing and energy resources and heterogeneous QoS requirements in modern IoT nodes, a client-centric access device selection approach for QoS provisioning in multiple Radio Access Technologies (RATs) scenario is proposed in this paper. The proposed algorithm provides the ability to specify node-specific QoS requirements at each node, a better access device selection, and improved network scalability. For experimental evaluation, a hybrid indoor network consisting of Wireless Fidelity (WiFi) and Light Fidelity (LiFi) RATs has been considered. Experimental results show that the proposed technique outperforms several conventional client-based access device selection approaches by up to 32.66% in network emulation experiments and up to 50% in hardware experimentation. The contribution of this paper includes the proposed algorithm, its complexity and game-theory based convergence analyses, evaluation of the proposed algorithm using network emulation and hardware-based setups, and LiFi channel analysis. The analyses and results imply that the proposed algorithm could be utilized in next-generation IoT networks as it performs better than conventionally used access device selection techniques for QoS provisioning.

INDEX TERMS Quality of service, beyond 5G IoT networks, IoT QoS provisioning, light fidelity, visible light communication.

I. INTRODUCTION

As 5G networks roll-out globally, researchers have started to lay out plans for Beyond 5G (B5G) networks. A few key requirements of B5G networks include universal connection, low latency, energy efficiency, reliability, and data rates of up to 100 terabits per second. Several new technologies are planned to be part of B5G networks, including terahertz communication, visible light communication, holographic radio, and large intelligent surface [1].

The IP traffic is predicted to increase to 396 exabytes by 2022, three times more than in 2017. By 2025, connected devices are expected to surpass 75.4 billion globally [2]. The Internet of Things (IoT) devices are expected to

reach hundreds of billions by the year 2030. B5G networks would face the challenge of a massive number of connected IoT devices and manifolds traffic increase. To provide Quality of Service (QoS) guarantees for ensuring certain network quality would be vital in such a scenario [3].

Current cellular networks are insufficient to accommodate massive IoT networks. The seamless connectivity for IoT devices in B5G networks would require solving various challenges, including QoS guarantees [4], dealing with excessive overheads [5], non-orthogonal multiple access techniques with improved spectral and bandwidth efficiencies [6], and improving energy efficiency [7].

B5G networks would comprise a range of IoT devices, including sensors, wearables, washing machines, health monitors, air conditioners, transportation systems, smartphones, and tablets, etc. [8]. The functionalities of these devices

The associate editor coordinating the review of this manuscript and approving it for publication was Chin-Feng Lai¹.

significantly vary from simple sensing to high processing devices. Where some IoT nodes would perform control and monitoring tasks, other nodes might have to stream high definition content [9]. Such a vast variation in the requirements of IoT nodes translates to significantly different QoS expectations for each node.

Conventionally, network-based approaches are used for QoS provisioning by associating a client to a specific access device or base station [10]. The QoS parameters used by such methods for access device selection are independent of the end-device and do not account for the node-specific QoS requirements. Besides, network devices lack an accurate last-hop view and are unaware of device and application-specific requirements at the node. On the other hand, there are client-based access device selection approaches presented in the literature, but not much consideration has been given to the massive number of devices expected in B5G IoT networks. Such client-based approaches also use the same QoS parameters for all end devices, ignoring the heterogeneous QoS requirements of different IoT nodes.

Another challenge for access device selection in B5G networks is posed by multiple Radio Access Technologies (RATs), including WiFi, 4G/LTE, 5G, and Bluetooth, etc. The end-devices need to not only opportunistically switch between access devices from the same access technology, e.g., between WiFi Access Points (APs), but also between access devices from multiple RATs. This challenge is further complicated by including new RATs in B5G networks such as Light Fidelity (LiFi) or Visible Light Communication (VLC) [11]. The terms LiFi and VLC are used interchangeably. Generally, VLC refers to the visible light spectrum, whereas LiFi refers to the complete light spectrum [12]. We will use the term LiFi for the rest of this paper. It is also desirable to allow multihoming, enabling a client to communicate over multiple RATs simultaneously.

To address the challenges mentioned above, we have proposed a client-centric access device selection algorithm to provision QoS considering heterogeneous requirements of IoT nodes, multiple RATs, and multihoming. Although the proposed algorithm in this paper would generally apply to any choice of RATs, a hybrid indoor WiFi and LiFi environment has been considered for experimental evaluation. This makes it necessary to discuss the motivation for selecting such an experimental setup.

B5G networks would be significantly dependent on WiFi. By 2022, WiFi hotspots are expected to increase to 549 million, along with 80% of mobile traffic occurring indoors [2]. Massive competition is expected in the limited frequency spectrum because of the dense deployment of WiFi APs. To address this challenge, LiFi technology [12] appears to be a promising candidate to complement WiFi because of an enormous 300 THz frequency spectrum and an expected 10 Gbps data rate [13]. Besides, LiFi offers better data security along with no RF emissions. However, LiFi technology is susceptible to obstructions and has limited coverage [14].

The rest of the paper is organized as follows. The background and related work are presented in Section II. The system model and the proposed algorithm are discussed in Section III. Section IV presents the LiFi channel analysis. As a proof of concept, network emulation based experimentation is performed in Section V. To validate the proof of concept, hardware experimentation is performed in Section VI. Section VII provides a discussion, whereas Section VIII concludes the paper.

II. BACKGROUND AND RELATED WORK

A. MOTIVATION AND PROBLEM STATEMENT

With the massive deployment of IoT devices in B5G networks, ensuring QoS requirements to provide a certain network quality would be essential. With the availability of multiple RATs in B5G networks, suitable access device selection to improve provided QoS is possible. Although network-based approaches [15]–[21] provide a more optimal access device allocation, using such approaches would result in a huge amount of signaling information being transmitted between network elements and IoT nodes. Besides, scalability issues of network-core would arise because of the expected massive number of IoT devices. A client-centric approach would address these concerns and benefit from the understanding of client-specific QoS requirements, power and application-specific requirements of the client, and a better knowledge of the client's access network state. Several client-centric approaches [22]–[25] have been proposed in the literature. However, none of these approaches consider heterogeneous QoS requirements of IoT nodes.

To the best of the authors' knowledge, there is no existing client-based QoS provisioning technique dealing with heterogeneous requirements of IoT nodes in B5G networks. Such a technique would allow each IoT node to choose QoS parameters different from other nodes to fit its QoS requirements. It is also believed that no other work has evaluated the performance of such a client-based technique using network emulation and hardware experiments considering the scenario of a hybrid WiFi and LiFi IoT network.

The following Research Questions (RQs) define this paper's research scope:

- RQ1: Is it possible to formulate a client-based access device selection algorithm considering client-specific QoS requirements for access device selection in B5G IoT networks?
- RQ2: Would such an algorithm work for multi-RAT and multihoming scenarios?
- RQ3: What would be the computational efficiency and convergence of the algorithm?
- RQ4: How do Signal-to-Noise Ratio (SNR) and data rate in a LiFi channel vary with respect to LiFi AP and receiver orientations, and by using light sources of different half-intensity radiation angles?

- RQ5: Would the proposed algorithm perform better than other access device selection techniques in a hybrid indoor WiFi and LiFi network?

B. RELATED WORK

IoT devices' massive deployment has allowed technology's creative intervention in many areas affecting human life, ranging from smart homes to smart industries and smart grids. The key objective of introducing smart devices in the human habitat revolves around collecting and analyzing usage patterns and providing effective resource utilization. Smart IoT devices have limited analytical capability to act autonomously on the sensor-generated data. The devices also tend to transmit the raw sensing data to its manufacturer for detailed data mining inside the remote data center [26]. Some other use cases entail explicit requirement of transmitting sensor data over the communication network, such as [27] for baby monitoring empowering smart cradle, and [28] for a pedestrian crossing control system. Assistance required from communication networks for few applications has hard QoS bounds to ensure the infrastructure's safety and security, such as mentioned in [29] for smart grids. Conversely, some other applications do not put such hard bounds and require IoT data transmission for awareness only, such as smart homes [30].

Conventionally, QoS provisioning in cellular networks has been the responsibility of network devices. In such a scenario, the association of a User Equipment (UE) to a specific Base Station or an eNodeB is controlled by the network [15]. Similarly, in computer networks, centralized decisions are made to select end-to-end paths by core-layer devices, including Layer-3 switches and routers. Software-Defined Networking controllers are recently deployed for QoS provisioning on behalf of core-network devices [16]. Centralized approaches help network operators to achieve certain overall network optimization related goals such as throughput maximization [17], load balancing [18], and user fairness enhancement [19]. Although centralized approaches achieve better network optimization, they usually result in significant communication overheads for exchanging information among different network entities [31].

The issue of excessive communication overheads can be addressed using distributed client-based RAT selection approaches [32]. Most of the distributive approaches are game-theory based [33]. These approaches are further classified into partially distributed and fully distributed. In partially distributed approaches, information from other users is also used, whereas, in fully distributed approaches, the RAT selection decision is made independently of other users [34]. For example, the decision to select between multiple WiFi APs in today's end-devices is fully distributed. A client usually selects among available WiFi APs based on a single Received Signal Strength Indicator (RSSI) metric [35]. A multi-RAT selection algorithm to select between cellular networks and WiFi APs also exists in today's end-devices. However, WiFi is mostly the preferred network in such a scenario, independent of the QoS provided by both networks.

Client-based RAT selection is considered a vital requirement in B5G networks. A cellular network selection mechanism has been proposed for a mobile device with simultaneous connectivity to two different cellular operators [22]. The factors considered for selecting a particular cellular network consist of handover rate, energy consumption, call quality, and financial cost. In [23], an approach to achieve higher data rates in 6G networks is proposed using data-fusion. Cooperation between user equipment and network devices is proposed, along with the use of machine-learning for network optimization. An approach using context-aware RAT selection is proposed in [24]. The presented experimental results indicate that context-aware RAT selection outperforms conventionally used RAT selection techniques, including signal strength, link throughput, and network delay. A multihoming-based multi-RAT selection technique for transmitting traffic of various classes over different RATs has also been presented in [25].

Recently, hybrid WiFi and LiFi networks have been researched in literature. In one such approach [20], fuzzy logic has been used to evaluate clients to be connected with a WiFi AP. The remaining clients are associated with the LiFi AP. An approach for load balancing is proposed in [36], with LiFi being the default network. Users with weak levels or no optical signals are shifted to the WiFi AP. For AP assignment, two different optimization algorithms are proposed, including a joint optimization algorithm and a separate optimization algorithm. Considering the effects of shadowing, a game-theoretic load balancing approach between WiFi and LiFi networks is proposed in [37]. Due to expected lower throughput at users experiencing shadowing and channel blockage in LiFi channel, such users are moved to a WiFi AP. Some work for developing a handover management protocol is presented in [38], considering factors such as link quality, interference, and call flow. The paper discusses using indoor networks and transportation systems' while considering a Femto AP and a LiFi AP. A hybrid WiFi and LiFi system for QoS provisioning using link selection is proposed with buffer overflow constraints and delay in [21]. Because of the smaller coverage regions of LiFi APs, a concept of handover skipping is proposed in [39] to enable handover between non-adjacent APs for hybrid WiFi and LiFi networks. A summary of access device selection literature is provided in Table 1.

C. CONTRIBUTION

As presented in Table 1, both network-centric and client-centric approaches have been proposed in the literature for access device selection. To the best of the authors' knowledge, this paper presents the first client-based access device selection algorithm for QoS provisioning in Beyond 5G IoT networks, while allowing each client to specify client-specific QoS parameters. The algorithm works in a multi-RAT environment and allows multihoming as well. Our previous work [40] presented an initial novel system model for such an access device selection. This paper expanded on the same work and made the following new contributions:

TABLE 1. Summary of access device selection techniques.

Referred Study	Network Type	QoS Parameters	Decision Control	Solution Method	Client-specific QoS Requirements	Multihoming Consideration	Multi-RAT Experimentation
[15]	5G Networks	Throughput	Network	Optimization Theory	No	No	No
[16]	Core Routing Networks	Throughput	Network	Optimization Theory	No	No	No
[17]	5G Networks	Throughput	Network	Optimization Theory	No	No	No
[18]	WiFi Networks	Throughput	Network	Optimization Theory	No	No	No
[19]	WiFi Networks	Throughput	Network	Optimization Theory	No	No	No
[20]	Hybrid WiFi and LiFi Networks	Throughput	Network	Fuzzy Logic	No	No	Yes
[21]	Hybrid WiFi and LiFi Networks	Buffer Overflow, Delay	Network	-	No	No	Yes
[22]	5G Networks	Energy Efficiency, Cost, Handover Rate, Call Quality	Client	Path Performance Calculation	No	No	No
[23]	6G Networks	Throughput	Client	Machine Learning	No	No	No
[24]	5G Networks	Throughput, Handover Rate, Packet Delivery Ratio	Client	Analytical Hierarchical Process	No	No	Yes
[25]	5G Networks	Traffic Class	Client	Optimization Theory	No	Yes	Yes
Proposed Algorithm	Beyond 5G Hybrid WiFi and LiFi IoT Networks	Any client-specific QoS parameter(s)	Client	Game-theory	Yes	Yes	Yes

- Improved the already proposed system model and problem formulation presented in [40]. The related work section is also improvised.
- A novel algorithm is proposed to solve the formulated problem and provide the above-discussed benefits.
- Complexity analysis and game-theory based convergence analysis are performed to evaluate the proposed algorithm.
- Since LiFi is envisioned to be an integral RAT of B5G networks, a LiFi channel analysis is performed.
- Evaluation of the proposed algorithm is performed using network emulation and hardware experimentations for a hybrid indoor WiFi and LiFi environment.

III. SYSTEM MODEL AND PROPOSED ALGORITHM

In this section, the network model and problem formulation are presented, followed by an example of a hybrid indoor network consisting of WiFi and LiFi access technologies. Afterward, the proposed algorithm is presented, along with its complexity and game-theoretic convergence analyses.

A. NETWORK MODEL

Consider a multi-RAT IoT environment, with one or more access devices belonging to each RAT. A set \mathcal{N} consists of all nodes in an IoT network, whereas the set \mathcal{D} contains all the access devices. The total number of nodes is N , and the

total number of access devices is D . A node $n \in \mathcal{N}$ may not be able to connect to all access devices in \mathcal{D} , but is only in the coverage area of a subset of access devices \mathcal{D}_n , where $\mathcal{D}_n \subseteq \mathcal{D}$. Let the number of access devices in \mathcal{D}_n be D_n .

There is a possibility that a node $n \in \mathcal{N}$ can simultaneously connect with multiple access devices, i.e., multihoming is possible. In such a case, a node can establish a connection with any number of access devices, up to the multihoming limit, defined as M_n . We define a combinations matrix \mathbf{C}_n containing all possible D_n bit binary combinations with number of ones less than or equal to M_n . Each c -th row of \mathbf{C}_n contains a D_n -bit binary combination, where each bit corresponds to a binary assignment variable x_{dc}^n . Here, $x_{dc}^n = 1$ indicates that for such a c -th binary combination, a connection exists between node $n \in \mathcal{N}$ and an access device $d \in \mathcal{D}_n$. Similarly, $x_{dc}^n = 0$ indicates otherwise.

As discussed in Section I, each IoT node could have a different QoS expectation from the network. These QoS requirements could be transformed into parameters classified into three categories:

- QoS parameter: Should satisfy a certain QoS threshold.
- Link quality parameter: Should be used in the calculation of link quality.
- Both QoS and link quality parameter.

The requirements of a node n for access device selection translates to a set of parameters \mathcal{P}_n , with P_n being

TABLE 2. Network model abbreviations.

Abbreviation	Description
\mathcal{N}	IoT nodes set
N	Total number of IoT nodes
\mathcal{D}	Access devices set
D	Total number of access devices
\mathcal{D}_n	Available access devices subset of node n
D_n	Access devices available to node n
\mathcal{P}_n	Parameters set of node n
P_n	Number of parameters of node n
x_{dc}^n	Binary variable indicating link between node n and access device d for c -th binary combination
\mathbf{C}_n	Combinations matrix of node n
\mathbf{Q}_n	QoS matrix of node n
\mathbf{L}_n	Link quality expectation matrix of node n
\mathbf{A}_n	Actual values matrix of node n
y_p	Selection variable of parameter p for link quality calculation
z_p	Selection variable of parameter p for QoS constraints
LQ_d^n	Link quality value between node n and access device d
OQ^n	Overall network quality value available to node n
M_n	Multihoming limit for node n
w_d^n	Workload/data transmitted in bits between node n and access device d
r_d^n	Link rate between node n and access device d
l_d^n	Link latency between node n and access device d
β_d^n	Link bandwidth between node n and access device d
SNR_d^n	Signal-to-Noise ratio of link between node n and access device d
R_{pd}	Receiver photodetector responsivity
κ	Power conversion coefficient (optical to electrical)
N_{LiFi}	Power spectral density of noise at LiFi receiver
$H_{LiFi_d}^n$	Impulse response of LiFi channel between node n and access device d
B_{LiFi}	Bandwidth of LiFi AP
P_{opt}	Transmitted optical power
N_{WiFi}	Noise power spectral density at WiFi receiver
$G_{WiFi_d}^n$	WiFi channel gain between node n and access device d
B_{WiFi}	Bandwidth of WiFi AP
P_{WiFi}	Transmit power of WiFi AP

the number of these parameters. Each parameter in \mathcal{P}_n is classified into one of the three above mentioned categories. A QoS matrix \mathbf{Q}_n for node n consisting of QoS requirements of all P_n parameters is defined as:

$$\mathbf{Q}_n = [q_1 \quad q_2 \quad \dots \quad q_{P_n}] \quad (1)$$

The corresponding value for a certain parameter in \mathbf{Q}_n with no QoS requirement (category no. 2) is set to ∞ . The link quality expectation matrix \mathbf{L}_n containing parameter values for the link quality calculation at node n is defined as:

$$\mathbf{L}_n = [l_1 \quad l_2 \quad \dots \quad l_{P_n}] \quad (2)$$

The corresponding values in \mathbf{L}_n for parameters not considered for link quality calculation (category no. 1) would be set to zero. Specifying a higher value in \mathbf{L}_n will result in a lower contribution made by the corresponding parameter in link quality calculation. The actual values matrix \mathbf{A}_n containing the values of P_n parameters for links from D_n access devices

at node n can be given as:

$$\mathbf{A}_n = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1P_n} \\ a_{21} & a_{22} & \dots & a_{2P_n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{D_n1} & a_{D_n2} & \dots & a_{D_nP_n} \end{bmatrix} \quad (3)$$

Each row of \mathbf{A}_n represents the perceived values of P_n parameters from each access device. For example, a_{11}, a_{12} , and a_{1P_n} could represent delay, jitter, and throughput from the first access device. Similarly, a_{D_n1}, a_{D_n2} , and $a_{D_nP_n}$ would represent delay, jitter, and throughput from the D_n -th access device.

Two selection variables $y_p \in \{-1, 0, 1\}$ and $z_p \in \{0, 1\}$ are defined. If p -th parameter is ignored in link quality calculation, the value of y_p is set to 0 for such p -th parameter. For a parameter with a lower desired value, e.g., delay or jitter, y_p equals 1, whereas, y_p is set to -1, if the desired value of the p -th parameter is high, e.g., bandwidth or SNR. For a p -th parameter with a lower desired value, QoS constraints selection variable z_p equals 1, and for a p -th parameter with a higher desired value, z_p equals 0.

The Link Quality (LQ) between node n and access device d can be defined as:

$$LQ_d^n = \sum_{p=1}^{P_n} y_p \frac{l_p - a_{dp}}{a_{dp}}, \quad a_{dp} \neq 0 \quad (4)$$

If OQ^n is defined as the overall network quality available to node n , the optimal access device selection problem is a maximization function computed separately at each node n :

$$OQ^n = \max (OQ^n, \sum_{d \in \mathcal{D}_n} x_{dc}^n LQ_d^n), \quad \forall c \in \mathbf{C}_n \quad (5)$$

subject to the following constraints:

$$C1: \begin{cases} a_{dp} \geq q_p & \text{if } z_p = 0 \\ a_{dp} \leq q_p & \text{if } z_p = 1 \end{cases}, \quad \forall d \in \mathcal{D}_n, p \in \mathcal{P}_n \quad (6)$$

$$C2: \sum_{d \in \mathcal{D}_n} x_{dc}^n \leq M_n, \quad \forall c \in \mathbf{C}_n \quad (7)$$

The constraint C1 ensures that the QoS requirements mentioned in \mathbf{Q}_n are satisfied. C2 is the multihoming constraint, allowing node n to connect to a maximum of M_n access devices simultaneously.

B. EXAMPLE: A HYBRID WiFi AND LiFi NETWORK

As an example, a hybrid WiFi and LiFi network is modeled in this sub-section with an objective to minimize link delay, with throughput and SNR QoS constraints. If the workload, a function of bits, between node n and access device d is defined as w_d^n , and r_d^n is the data rate available to node n from access device d , the communication link latency between them can be given as:

$$l_d^n = \frac{w_d^n}{r_d^n} \quad (8)$$

The data rate r_d^n between node n and access device d is dependent on the available bandwidth β_d^n and signal-to-noise ratio SNR_d^n . The data rate can be calculated for a WiFi network using Shannon–Hartley theorem. However, for the LiFi network, a gap exists between the upper bound calculated using Shannon–Hartley equation and the actual capacity of a LiFi link [41]:

$$r_d^n = \begin{cases} \frac{\beta_d^n}{2} \log_2 \left(1 + \frac{e}{2\pi} SNR_d^n \right), & \text{for LiFi channel} \\ \beta_d^n \log_2 (1 + SNR_d^n), & \text{for WiFi channel} \end{cases} \quad (9)$$

Using the channel model in [20], the SNR of the link between node n and access device d can be given as:

$$SNR_d^n = \begin{cases} \frac{R_{pd} H_{LiFi_d^n} P_{opt} \kappa}{N_{LiFi} B_{LiFi}}, & \text{for LiFi channel} \\ \frac{G_{WiFi_d^n} P_{WiFi}}{N_{WiFi} B_{WiFi}}, & \text{for WiFi channel} \end{cases} \quad (10)$$

In the above equation, R_{pd} is the photodetector's responsivity, $H_{LiFi_d^n}$ represents the LiFi channel between node n and access device d , P_{opt} represents the transmitted optical power, and κ is the optical to electrical power conversion coefficient. The Power Spectral Density (PSD) of noise at the node is given as N_{LiFi} , whereas B_{LiFi} shows the bandwidth provided by the LiFi AP. In a WiFi channel, the SNR between node n and access device d is calculated from the channel gain $G_{WiFi_d^n}$, WiFi AP transmitted power P_{WiFi} , channel bandwidth B_{WiFi} provided by WiFi AP, and noise PSD at the WiFi receiver N_{WiFi} .

Consider a hybrid WiFi and LiFi network with latency minimization objective, and rate and SNR QoS constraints. If latency with any arbitrary value l_1 is the only link quality parameter, the link quality expectation matrix can be defined as:

$$\mathbf{L}_n = [l_1 \quad 0 \quad 0] \quad (11)$$

Similarly, considering data rate and SNR as QoS constraints with thresholds of ϕ_R and ϕ_{SNR} , respectively, the QoS matrix \mathbf{Q}_n can be given as:

$$\mathbf{Q}_n = [\infty \quad \phi_R \quad \phi_{SNR}] \quad (12)$$

Considering hybrid network with a single WiFi AP and a single LiFi AP, the matrix of actual values \mathbf{A}_n would be given as:

$$\mathbf{A}_n = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \quad (13)$$

In the above matrix, the parameters of delay, rate, and SNR received from WiFi AP at node n are represented by a_{11} , a_{12} , and a_{13} , respectively. The same parameters are represented by a_{21} , a_{22} , and a_{23} for the LiFi AP.

With the three matrices \mathbf{Q}_n , \mathbf{L}_n , and \mathbf{A}_n defined, the overall quality maximization problem can be formulated using (4) and (5), subject to the following constraints:

$$C1 - a: \quad a_{d1} \leq \infty, \quad d \in \{1, 2\} \quad (14)$$

$$C1 - b: \quad a_{d2} \geq \phi_R, \quad d \in \{1, 2\} \quad (15)$$

$$C1 - c: \quad a_{d3} \geq \phi_{SNR}, \quad d \in \{1, 2\} \quad (16)$$

$$C2: \quad \sum_{d \in \mathcal{D}_n} x_{dc}^n \leq 1, \quad \forall c \in \mathbf{C}_n \quad (17)$$

Here, C1 is the QoS provisioning constraints. The constraint C1-a indicates that the delay from the selected AP should be less than ∞ , which would always be the case. The constraints C1-b and C1-c ensure that the available data rates and SNR from the selected AP are greater than ϕ_R and ϕ_{SNR} , respectively. The constraint C2 shows that multihoming is not allowed, and the maximum number of APs a node can connect is 1.

C. PROPOSED CLIENT-CENTRIC ALGORITHM

The proposed client-centric algorithm is shown in Algorithm 1. In this algorithm, the objective function shown in (5) is maximized individually at each client, subject to the constraints mentioned in (6) and (7). Various client-specific and radio parameters are input to the algorithm, along with the value of current Overall Quality (OQ) and the list of currently connected Access Devices (AD) at time instant t . The algorithm's output is the updated OQ and AD for the next time instant $t + 1$.

At the start of the algorithm, the values of AD and OQ for time $t + 1$ are initialized to the respective value at time t (Line 1 and Line 2). This ensures that if no better selection of available access devices could improve the overall quality, the same access devices would be used at the next time instant $t + 1$. As D_n access devices cover node n , it can connect to any of these access devices, subject to the maximum number of simultaneous connections limited by constraint C2 mentioned in (7). All the possible binary combinations that satisfy constraint C2 are stored in \mathbf{C}_n (Line 3).

The outer for-loop (Line 4 to 36) iterates over all these possible combinations of \mathbf{C}_n and compute the LQ and AD for each c -th combination at time $t + 1$. LQ is initialized to a null value (Line 5), and AD is initialized to an empty set (Line 6) for c -th combination at time instant $t + 1$. The length of each c -th combination is D_n bits.

The middle for-loop (Line 7 to 19) iterates over each d -th bit of the c -th combination. The existence of a connection between node n and device d for combination c , i.e., x_{dc}^n equals 1, is checked (Line 8). If a connection exists, the device d is appended to the AD set of c -th combination at time $t + 1$ (Line 9), and the inner for-loop (Line 10 to 17) is executed, iterating over all link quality parameters. If the QoS constraint (constraint C1) is satisfied for a specific parameter (Line 11), the LQ of c -th combination at time $t + 1$ is updated (Line 12). Otherwise, if the constraint C1 is not satisfied

Algorithm 1 Access Device Selection Algorithm for Node n

Input: $A_n, L_n, Q_n, Y_n, Z_n, M_n, D_n, P_n, \eta, T, h_n, s_n, OQ(t), AD(t)$
Output: $OQ(t+1), AD(t+1), h_n, s_n$

- 1: $AD(t+1) \leftarrow AD(t)$
- 2: $OQ(t+1) \leftarrow OQ(t)$
- 3: $C_n \leftarrow$ all possible binary combinations of length D_n satisfying constraint C2
- 4: **for** c in C_n **do**
- 5: $LQ(c, t+1) \leftarrow 0$
- 6: $AD(c, t+1) \leftarrow \{\}$
- 7: **for** $d \leftarrow 1$ to D_n **do**
- 8: **if** $x_{dc}^n = 1$ **then**
- 9: $AD(c, t+1) \leftarrow AD(c, t+1) || d$
- 10: **for** $p \leftarrow 1$ to P_n **do**
- 11: **if** $(a_{dp} \geq q_p \text{ AND } z_p = 0)$ OR $(a_{dp} \leq q_p \text{ AND } z_p = 1)$ **then**
- 12: $LQ(c, t+1) \leftarrow LQ(c, t+1) + y_p * ((l_p - a_{dp})/a_{dp})$
- 13: **else**
- 14: $LQ(c, t+1) \leftarrow 0$
- 15: Go to Line 20
- 16: **end if**
- 17: **end for**
- 18: **end if**
- 19: **end for**
- 20: **if** $AD(c, t+1) \neq AD(t)$ **then**
- 21: **if** $\frac{LQ(c, t+1)}{OQ(t)} > \eta, \forall t = t - T + 1, \dots, t$ **then**
- 22: **if** $LQ(c, t+1) > h_n$ **then**
- 23: **if** $rand \leq p^{s_n+1}$ **then**
- 24: $AD(t+1) \leftarrow AD(c, t+1)$
- 25: $OQ(t+1) \leftarrow LQ(c, t+1)$
- 26: Update h_n
- 27: **if** concurrent move **then**
- 28: $s_n \leftarrow s_n + 1$
- 29: **else**
- 30: $s_n \leftarrow 0$
- 31: **end if**
- 32: **end if**
- 33: **end if**
- 34: **end if**
- 35: **end if**
- 36: **end for**

(Line 13), $LQ(c, t+1)$ is set to zero (Line 14), and inner and middle for-loops are immediately exited (Line 15). At the end of the middle for-loop, the final value of $AD(c, t+1)$ and $LQ(c, t+1)$ for c -th combination at time $t+1$ is available.

The value of $AD(t+1)$ is updated to $AD(c, t+1)$ (Line 24), and $OQ(t+1)$ is replaced with $LQ(c, t+1)$ (Line 25), if four if-conditions are satisfied. The first if-conditions checks if the list of access devices in $AD(c, t+1)$ is different from $AD(t)$ (Line 20). The expected gain defined as $\frac{LQ(c, t+1)}{OQ(t)}$ should

exceed threshold η for the past T thresholds in the second if-condition (Line 21). Here T represents the frequency of measurement. Since nodes can switch between access devices to selfishly increase their link quality, there is a possibility that some nodes will keep on switching without reaching an equilibrium. A hysteresis parameter h_n is introduced for achieving convergence by dampening oscillations. This parameter h_n denotes the dependence of the access device switching to the history of past switches made by the node n . The value of h_n is updated according to some hysteresis policy (Line 26). For example, if we have access devices classified into multiple classes, e.g., throughput-fair, time-fair, proportional-fair, etc., the value of h_n for a specific class is the achieved link quality prior to leaving that class. For node n to re-associate to an access device in this class, the value of $LQ(c, t+1)$ should be greater than h_n (Line 22). If the access devices are not categorized into multiple classes, we can assume that this condition is always satisfied.

There is a possibility that the nodes concurrently switch to the same access device. This will result in lower link quality as expected. To minimize concurrent switches, nodes switch probabilistically with probability $p < 1$ (Line 23). The parameter p is dependent on network congestion and acts in the same manner as the 802.11 contention window mechanism. When nodes switch simultaneously to the same access device (Line 27), the value of switching parameter s_n is incremented by 1 (Line 28). Otherwise (Line 29), the value of s_n is set to a null value (Line 30).

D. ALGORITHMIC COMPLEXITY

The algorithmic complexity of Algorithm 1 is computed in two different ways in this sub-section; floating-point operations (flops) and Big-O notation.

1) FLOP COUNT

Each real floating-point operation is defined as a flop. For computing the number of flops consumed by Algorithm 1, one flop count is consumed for each operation, including arithmetic operations (addition, subtraction, multiplication, division), assignment operation, logical comparison, and addition or removal of an element from a set [42].

In Line 1, the set $AD(t+1)$ is replaced with $AD(t)$. This would require popping already stored elements in $AD(t+1)$ and inserting elements from $AD(t)$ to $AD(t+1)$. In the worst case, the already store elements in $AD(t+1)$ would be D_n , and the elements to be inserted into $AD(t+1)$ would also be D_n . This means that Line 1 will take a maximum of $2D_n$ flops. Since $OQ(t+1)$ is a single floating-point number and is replaced by $OQ(t)$, Line 2 will consume only one flop. All possible binary combinations of length D_n that satisfy constraint C2 are computed and assigned to C_n . In the worst case, the combinations of interest would be all 2^{D_n} combinations, except the combination with all zeros. Line 3 will, therefore, utilize $2^{D_n} - 1$ flops to compute all possible binary combinations and $2^{D_n} - 1$ flops to assign these combinations to C_n .

There are three nested for-loops in Algorithm 1. The inner for-loop (Line 10 to Line 17) uses two flops in Line 10 for initializing p (for the first iteration) or incrementing p (for remaining iterations), and checking if $p \leq P_n$. Seven flops are used in Line 11 to perform seven logical comparison operations. In Line 12, five flops are used to perform an assignment operation and four arithmetic operations. As the else-condition from Line 13 to Line 15 consumes fewer flops than the if-condition from Line 11 to Line 12, the number of flops consumed by the else-condition is ignored. Since the inner for-loop is repeated P_n times, a total of $14P_n$ flops are consumed by this loop.

Similarly, the middle for-loop (Line 7 to Line 19) uses two flops for Line 7. One flop is used each by Line 8 and Line 9 for logical comparison and adding an element to set $AD(c, t + 1)$, respectively. As the middle for-loop also consists of the inner for-loop and iterates D_n times, the total number of flops consumed by the middle for-loop would be $(4 + 14P_n)D_n$.

The outer for-loop (Line 4 to Line 36) uses two flops in Line 4 to increment and compare if loop index c is still in C_n . One flop each is used by Line 5 and Line 6 to initialize $LQ(c, t + 1)$ with a null value and $AD(c, t + 1)$ with an empty set, respectively. A comparison of two sets $AD(c, t + 1)$ and $AD(t)$ is performed in Line 20. Since this is an element-by-element comparison, the worst-case scenario will result in D_n element-by-element comparisons, thus, utilizing D_n flops. Two flops are used in Line 21 for an arithmetic operation and a comparison operation. One flop each is used by Line 22 and Line 23 to perform a comparison operation. To update the value of $AD(t + 1)$ in Line 24, at maximum D_n elements will be removed, and D_n elements will be added. This will result in $2D_n$ flops being used. One flop is used to update the value of $LQ(t + 1)$ in Line 25. Although hysteresis and concurrent move are discussed in Line 26 and Line 27, the hysteresis policy could vary, as well as the strategy to detect concurrent move. A detailed discussion of these policies is not carried out in this paper. For complexity analysis, we assume that Line 26 uses α flops, and Line 27 uses β flops. The body of if-condition in Line 28, as well as the body of else-condition in Line 30, each uses one flop. Therefore, we will count only one flop being utilized by both of these lines, as only one of these lines would be executed in an iteration. As discussed earlier, the length of C_n could be at most $2^{D_n} - 1$, the outer for-loop will also iterate a maximum of $2^{D_n} - 1$ times. The number of flops thus consumed by outer for-loop would then be $(10 + \alpha + \beta + 7D_n + 14P_nD_n)(2^{D_n} - 1)$. Adding to it the $2D_n + 1 + 2(2^{D_n} - 1)$ flops consumed by Lines 1 to 3, the total number of flops used by Algorithm 1, F_{A1} , are:

$$F_{A1} = (10 + \alpha + \beta + 7D_n + 14P_nD_n)(2^{D_n} - 1) + 2D_n + 1 + 2(2^{D_n} - 1) \quad (18)$$

$$F_{A1} \approx (\alpha + \beta + 7D_n + 14P_nD_n)(2^{D_n} - 1) \quad (19)$$

In Line 3 of Algorithm 1, the binary combinations of interest are those with the number of ones less than or equal

to M_n . The total number of these combinations of interest, represented as Z , can be computed as $Z = \sum_{k=1}^{M_n} \frac{D_n!}{(k!(D_n - k)!)}$, and F_{A1} can then be written as:

$$F_{A1} \approx (\alpha + \beta + 7D_n + 14P_nD_n)(Z) \quad (20)$$

Equations (19) and (20) indicate that flops consumed by Algorithm 1 increase exponentially as the number of access devices, i.e., D_n increases. Exponential algorithms do not generally work very well for larger inputs. For the case where multihoming is not allowed, nodes can connect only to one access device, resulting in $Z = D_n$. This would ensure that the flop count does not increase exponentially, but cubically. For such a case, the flop count F_{A1} can be stated as:

$$F_{A1} \approx (\alpha + \beta + 7D_n + 14P_nD_n)(D_n) \quad (21)$$

For the scenario of hybrid WiFi and LiFi indoor network, where nodes have access to only two access devices and can connect to a maximum of one access device at a time, i.e., $M_n = 1$, the flops consumed becomes linear. With no hysteresis policy and no detection of concurrently moving nodes to the same access device, F_{A1} would be given as:

$$F_{A1} \approx (14 + 28P_n)(2) \quad (22)$$

$$F_{LiFi-WiFi} \approx 56P_n \quad (23)$$

2) BIG-O NOTATION

We have three nested for-loops in Algorithm 1. The outer for-loop runs from 1 to $2^{D_n} - 1$, the middle for-loop runs from 1 to D_n , and the inner for-loop runs from 1 to P_n . In Big O notation, the algorithmic complexity is exponential, i.e., $O(2^{D_n} \times D_n \times P_n)$. For the case of no multihoming, the outer for-loop runs from 1 to D_n , and the complexity reduces to cubic, i.e., $O(D_n^2 \times P_n)$. For the case of hybrid WiFi and LiFi network with no multihoming ($D_n = 2$), the complexity is linear, i.e., $O(P_n)$.

E. GAME-THEORETIC ANALYSIS

In our proposed algorithm, a client will always select access devices with maximum overall link quality. Since there is only one-hop, we can assume that the proposed algorithm will always select the most optimal access devices and is optimal from the client's point of view. From the network's point of view, the allocation might be sub-optimal, but there is no feedback from the network to the client, and we cannot evaluate optimality from the network's point of view.

Another approach is to perform a game-theoretic analysis of such distributed algorithms. A scenario where individual nodes locally decide to maximize link quality in a fully distributed fashion, and without any intervention from other entities, could be modeled as a non-cooperative game. For such a game, the set of players is the set of nodes, and the set of strategies is the set of access devices. Such a non-cooperative game converges to Nash Equilibrium if each player considers its selected strategy to be optimal given the choices of all

other players, that is, it cannot unilaterally improve its payoff by changing strategies.

Algorithm 1 is guaranteed to converge to Nash Equilibrium for any combination of access devices (Theorems 1 - 3, [43]). In the case of time-fair access devices, the average Pareto-efficiency gain (average throughput improvement per client) between a non-Pareto-optimal Nash profile from Algorithm 1 and a Pareto-dominant profile is bounded by (Theorem 6, [43]):

$$\begin{cases} 2, & \text{if } N \leq D \\ \frac{N+D}{N}, & \text{if } N \geq D \end{cases} \quad (24)$$

For proportional-fair access devices, the same average Pareto-efficiency gap is bounded by (Theorem 7, [43]):

$$\begin{cases} 2 \cdot (1 + \ln(N)), & \text{if } N \leq D \\ \frac{N+D}{N} \cdot (1 + \ln(N)), & \text{if } N \geq D \end{cases} \quad (25)$$

IV. LiFi CHANNEL ANALYSIS

Before performing experimentation, we analyze the LiFi channel in this section for developing a detailed understanding of such a channel at different heights, distances, and orientations. We consider a LiFi channel model, as shown in Figure 1. We assume no interference from other light sources in such a scenario and no reflecting objects. The signal received at the node from LiFi AP is made up only of the Line-of-Sight (LoS) component.

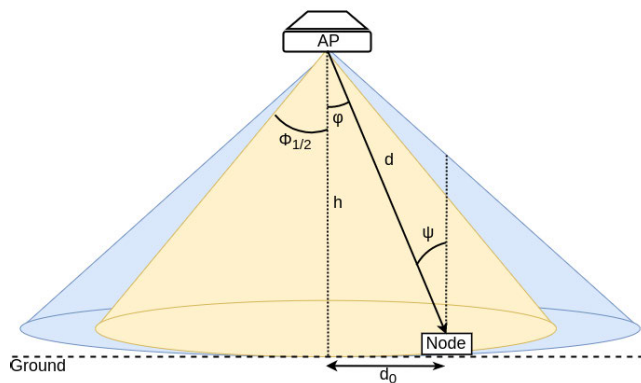


FIGURE 1. LiFi channel model.

We consider a similar channel model and channel parameters for our analysis, as proposed in [20]. We consider an LoS signal received from LiFi AP at the photodetector on the node lying on the ground. A field angle defines the area where the lamp could radiate 10% of its maximum luminous intensity, shown with a blue circle. The beam angle defines the area where the lamp radiates at least half (50%) of its maximum luminous intensity, shown with a yellow circle. We define this half-intensity radiation angle as $\Phi_{1/2}$.

We further define two other angles: the angle of irradiance (ϕ) and the angle of incidence (ψ). The angle of irradiance is the angle with respect to the perpendicular axis

of AP, and the angle of incidence is the angle with respect to the perpendicular axis of the node. We also define the height between the AP and the ground as height h , and the distance between the AP and the node as d . We calculate the distance d by using Pythagoras' theorem on height h and the horizontal distance d_0 between the center point of the light source and the photodetector at the node.

The Lambertian emission order m is calculated as $m = -\ln 2 / \ln(\cos\Phi_{1/2})$. We define optical filter gain as g_f , n as the refractive index, A_{pd} as the physical area of the photodetector, and Ψ_{max} as the semi-angle of the field of view of the photodetector. The optical concentrator gain $g_c(\Psi)$ is then given as:

$$g_c(\Psi) = \begin{cases} \frac{n^2}{\sin^2(\Psi_{max})} & 0 \leq \Psi \leq \Psi_{max} \\ 0 & \Psi > \Psi_{max} \end{cases} \quad (26)$$

The LiFi channel in (10) can be defined as:

$$H_{LiFi} = \frac{(m+1)A_{pd}}{2\pi d^2} \cos^m(\phi) g_f g_c(\psi) \cos(\psi) \quad (27)$$

For LiFi channel analysis, we consider the parameters as defined in Table 3. We analyze the LiFi channel by evaluating the channel SNR and data rate per node by varying angle of irradiance (Figure 2), angle of incidence (Figure 3), and using different light sources with different half-intensity radiation angles (Figure 4). In all three analyses, we also vary the distance d between the AP and the node, and observe that SNR decreases with increasing distances. This is obvious because signal power reduces over distance, ultimately reducing the SNR. As the data rate is directly proportional to SNR, the data rate per node also reduces. We use (9) to calculate the bandwidth from the SNR, assuming that the LiFi AP uses Time Division Multiple Access (TDMA), and the bandwidth is equally divided among eight users.

TABLE 3. LiFi channel analysis parameters.

Parameter	Value
Physical Area of photodetector, A_{pd}	10^{-4} m^2
Responsivity of photodetector, R_{pd}	0.53 A/W
Optical filter gain, g_f	1
Refractive Index, n	1.5
Optical to electrical power conversion coefficient, κ	3
Half-intensity radiation angle, $\Phi_{1/2}$	60°
Photodetector semi-angle of field of view, Ψ_{max}	90°
AP transmit power, P_{opt}	3 W
AP bandwidth, B_{LiFi}	40 MHz
LiFi noise PSD, N_{LiFi}	$10^{-21} \text{ A}^2/\text{Hz}$

Figure 2 shows that the SNR and data rate decreases with an increasing angle of irradiance ϕ . The increasing value of ϕ corresponds to moving the node away from the center of the light source. The angle of incidence ψ is kept 0° , which means the receiver on the node is directly facing the light source. The light source considered for this result has a half-intensity radiation angle $\Phi_{1/2}$ of 60° .

In the next analysis in Figure 3, we consider the same light source with $\Phi_{1/2} = 60^\circ$, with the node lying on the center

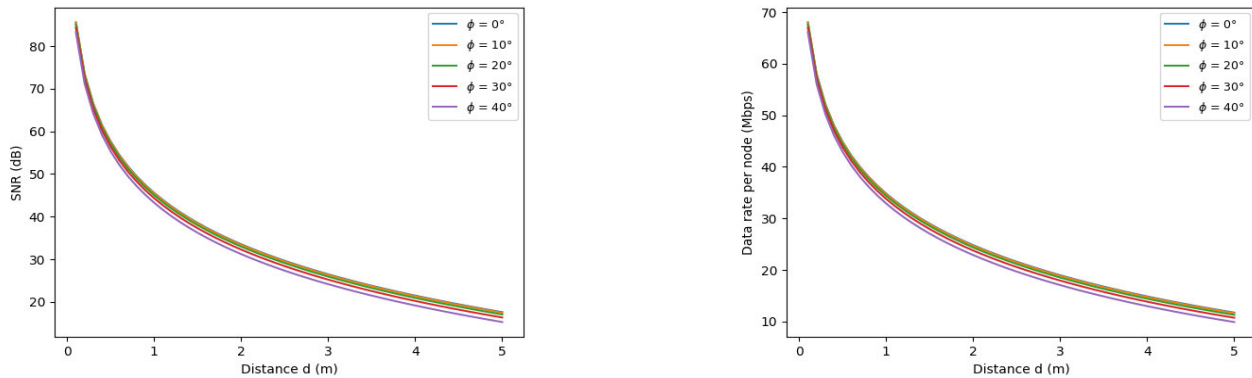


FIGURE 2. Calculated SNR and data rate per node by varying ϕ and d , with $\psi = 0^\circ$ and $\Phi_{1/2} = 60^\circ$.

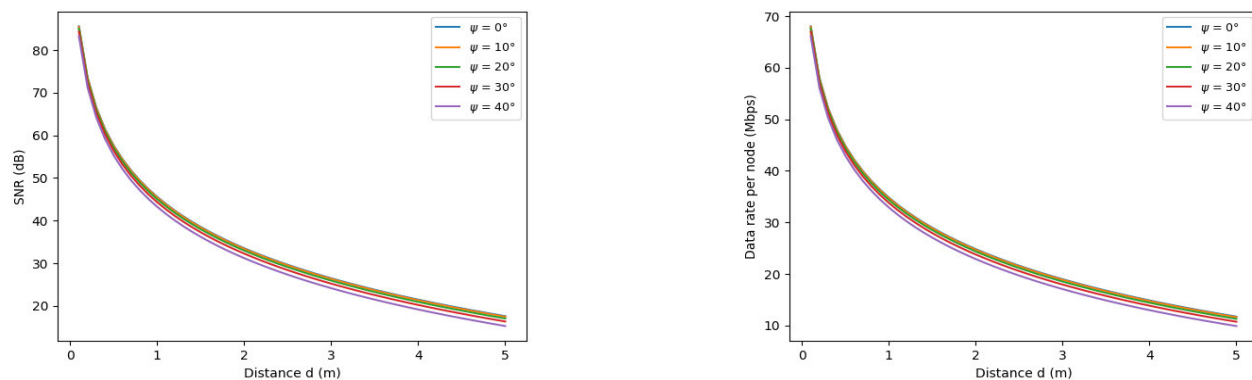


FIGURE 3. Calculated SNR and data rate per node by varying ψ and d , with $\phi = 0^\circ$ and $\Phi_{1/2} = 60^\circ$.

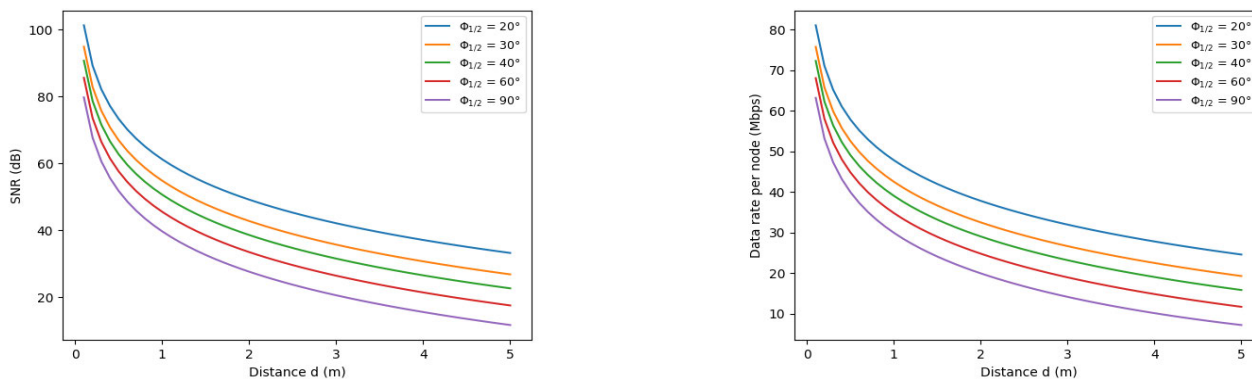


FIGURE 4. Calculated SNR and data rate per node by varying $\Phi_{1/2}$ and d , with $\phi = 0^\circ$ and $\psi = 0^\circ$.

spot of the coverage area of the light source, i.e., $\phi = 0^\circ$. Here, we vary the angle ψ , which corresponds to rotating the receiver at the node, and observe that the signal SNR and data rate decreases with increasing values of ψ .

In the last analysis shown in Figure 4, both ϕ and ψ equals 0° . This means the receiver at the node is directly facing the light source and is placed at the center of the light coverage area. However, we use different light sources, having different $\Phi_{1/2}$ values. With light sources having smaller coverage angles, the light is more concentrated in the

coverage area, providing better SNR and data rates. As the light coverage area increases, i.e., $\Phi_{1/2}$ becomes larger, both SNR and data rate reduces.

V. PROOF OF CONCEPT: NETWORK EMULATION BASED EXPERIMENTATION

This section presents some network emulation based experimentation for proof of concept of the proposed algorithm. The network emulation based setup and obtained results are explained in the following sub-sections.

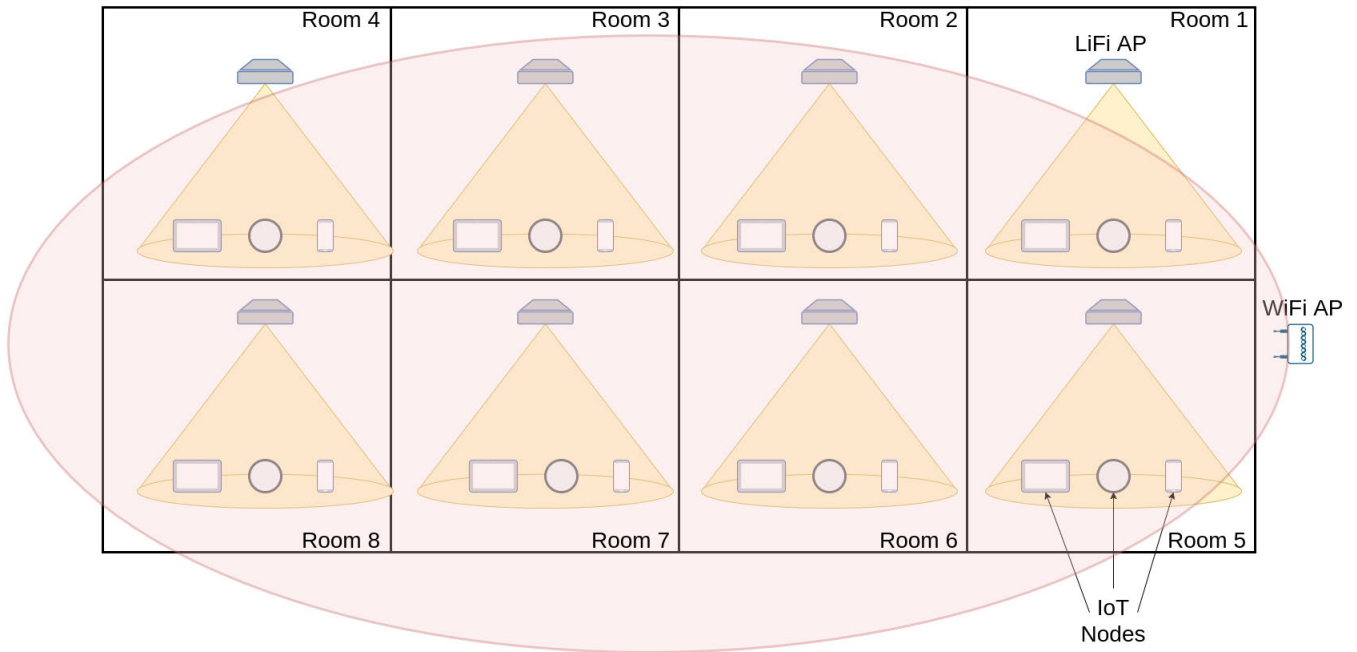


FIGURE 5. Hybrid WiFi and LiFi network emulation experimentation setup.

A. SETUP

The network emulation setup considered is shown in Figure 5. The setup has a single WiFi AP and eight LiFi APs, with one LiFi AP placed in each room. The yellow cone shows the coverage area of a LiFi AP, whereas the pink oval shows the WiFi AP’s coverage area. Three nodes are placed under each LiFi AP, whereas a single WiFi AP covers all nodes. Such a network emulation setup is considered for a typical office corridor with eight rooms. For simplicity, we have assumed that all the nodes are stationary. The network emulation setup parameters are summarized in Table 4.

TABLE 4. Network emulation setup parameters.

Parameter	Value
Number of WiFi AP	1
Number of LiFi APs	8
Number of IoT Nodes	24
WiFi AP data rate	54 Mbps
WiFi link delay	20 - 40 msec
LiFi link data rate	20 Mbps
LiFi link delay	40 msec
UDP flow size	1 - 13 Mbps

To emulate such a network, Mininet [44] network emulator has been used. Mininet provides virtual spaces for nodes using Linux kernel API and configures various link parameters to emulate the described setup. For proof-of-concept, both WiFi and LiFi links are modeled using TCLink class in Mininet. The LiFi link bandwidth available at each node is considered to be constant at 20 Mbps, with a 40 msec link delay. The link bandwidth and delay from the WiFi AP are assumed to be a function of distance. The IoT nodes

in Room 1 and Room 5 receive a WiFi link bandwidth of 54 Mbps with 20 msec delay to the WiFi AP, nodes in Room 2 and Room 6 receive a WiFi link bandwidth of 36 Mbps with 30 msec delay to the WiFi AP, whereas nodes in Room 3 and Room 7 receive a WiFi link bandwidth of 18 Mbps with 40 msec delay to the WiFi AP. For nodes in Room 4 and Room 8, the WiFi connection is assumed to be very bad, with SNR less than the decodability threshold ϕ_{SNR} .

To evaluate QoS-based network performance, User Datagram Protocol (UDP) flows of sizes 1 to 13 Mbps have been admitted at each node. For results, each experiment is repeated ten times, where the plotted data-point represents the mean value, and the error bar indicates the standard deviation. To calculate delays between APs and nodes, Internet Control Message Protocol (ICMP) is used, whereas iperf utility is used for generating the UDP traffic.

B. RESULTS

We have compared the results of our Proposed Methodology (PM), based on Algorithm 1, with AP techniques used conventionally. These techniques include Best SNR (B-SNR) and Maximum Bandwidth (M-BW). The PM aims to minimize the delay with throughput and SNR as QoS constraints. The B-SNR technique opts for the AP, offering the best SNR. On the other hand, the M-BW technique chooses AP offering the maximum link bandwidth.

For the first analysis, the throughput QoS constraint is evaluated. If the flow admitted at a node is received with the same throughput at the AP, the node QoS is marked as satisfied, otherwise, the QoS is marked as unsatisfied.

Figure 6 evaluates the percentage of nodes whose QoS is satisfied with increasing flow size admitted at each node. With the increasing sizes of flows being admitted, network congestion also increases, making it challenging to satisfy each node's throughput QoS requirement. Such a behavior can be seen in Figure 6, where the percentage of nodes with QoS satisfied decreases for all techniques as the flow size increases. The behavior for all techniques stays the same till the flow size reaches 3 Mbps, with all techniques satisfying QoS of 100% of nodes. As the PM selects AP based on throughput QoS constraint, it outperforms B-SNR and M-BW techniques for flow sizes greater than or equal to 7 Mbps. The M-BW technique performs the worst because of the congestion caused by all nodes selecting the WiFi AP, and not utilizing the LiFi link at all.

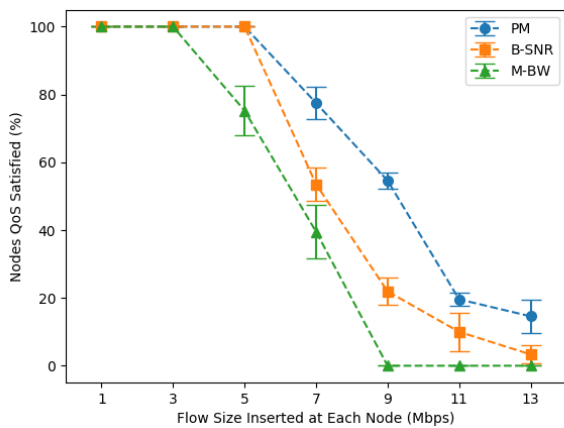


FIGURE 6. Percentage of QoS satisfied nodes.

The average throughput achieved per node is compared for all three techniques in Figure 7. With PM performing the best for satisfying throughput QoS constraint, the technique is also expected to achieve the highest average throughput per node. Considering an ideal network with no limitation of capacity, the average throughput per node would always be equal to the admitted flow size. This is shown with a red dotted line

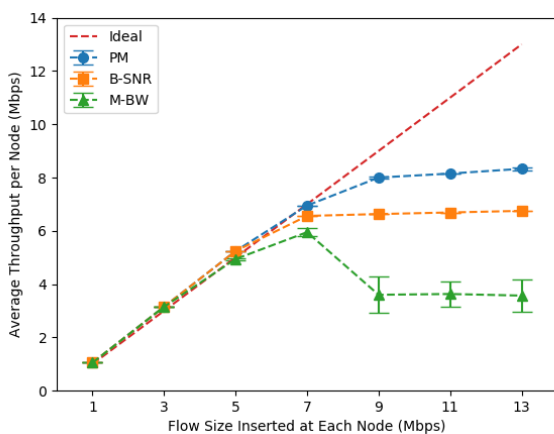


FIGURE 7. Average throughput per node.

in Figure 7. As the networks with infinite capacities do not exist in reality, the average throughput per node decreases as network congestion increases. A trend similar to Figure 6 can be seen in Figure 7, with all techniques providing almost an ideal average throughput per node till the admitted flow size per node is less than or equal to 5 Mbps. When the flow size increases beyond 5 Mbps, the PM achieves a higher average throughput per node compared with B-SNR and M-BW. Network congestion for PM occurs for flow sizes greater than or equal to 9 Mbps, as shown by the PM curve following the ideal curve till 7 Mbps. For B-SNR and M-BW, this congestion occurs earlier at 7 Mbps and 5 Mbps, respectively. After congestion occurs in the network, the achieved average throughput per node curve starts to flatten. A drop in throughput is observed for M-BW beyond 7 Mbps. This behavior indicates the presence of severe network congestion. As the size of admitted flows in a congested network increases, the network becomes further congested, resulting in packet losses. The result is lesser packets received at the AP and a lower throughput between the node and the AP. This behavior is not yet visible in PM and B-SNR. However, if experiments would be conducted beyond 13 Mbps, congestion will occur for these two techniques as well at a certain data rate.

Considering the SNR and throughput, the objective of the PM is to minimize link delay. The average delay per node is evaluated and compared in Figure 8. It can be seen that the PM offers the lowest delay and is very close to B-SNR. The reason for such a behavior is lower link delays being considered for links offering better SNR. Although in other network topologies, it is expected that the PM will outperform B-SNR. As M-BW does not give any consideration to the link delay, it has performed the worst. As the flow size increases beyond 9 Mbps, severe network congestion in M-BW starts to drop all the ICMP packets, and the evaluation of delay is no more possible. This is represented by the M-BW curve extending beyond the visible y-axis in Figure 8.

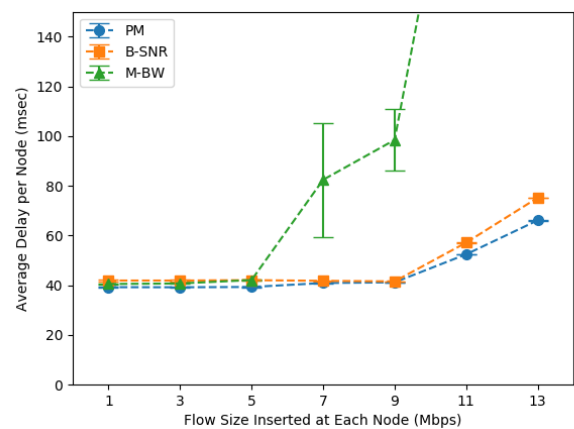


FIGURE 8. Average delay per node.

The number of nodes connected to WiFi and LiFi APs is compared in Figure 9. As determined by Algorithm 1, the nodes distribution among WiFi and LiFi APs varies

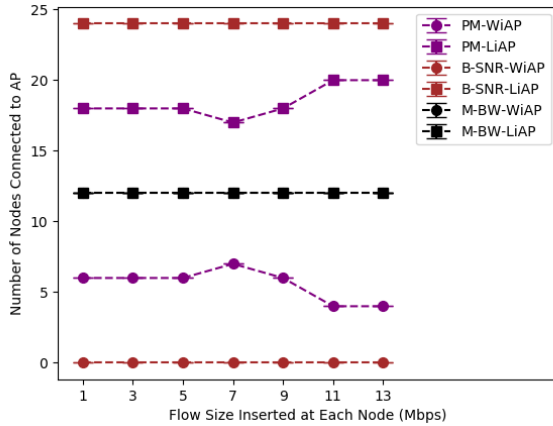


FIGURE 9. Nodes distribution among WiFi and LiFi APs.

in PM with different admitted flow sizes. As there is only a single WiFi AP, the WiFi link would quickly become congested with increasing flow sizes. Therefore, more nodes would associate with LiFi APs at higher admitted flow sizes in PM. Around six or more nodes connect to WiFi AP at lower admitted flow sizes, whereas only four nodes connect to WiFi AP at flow sizes greater than 9 Mbps in PM. With the assumption of nodes directly under the LiFi AP, and considering no blocking and shadowing, nodes always experience better SNR from the respective LiFi AP, as compared to the WiFi AP. Therefore, all nodes are associated with LiFi APs in B-SNR. In the case of M-BW, all nodes receive bandwidth of 20 Mbps from their respective LiFi APs. Nodes in Room 1 and Room 5 receive a bandwidth of 54 Mbps over the WiFi link, whereas the nodes in Room 2 and Room 6 receive a bandwidth of 36 Mbps. The bandwidth received over the WiFi link is less than 20 Mbps for the remaining nodes. Therefore, the twelve nodes in Room 1, Room 2, Room 5, and Room 6 connect to the WiFi AP. At the same time, the other twelve nodes use the LiFi link. As the same number of nodes connect to both LiFi and WiFi APs, the black-lines in Figure 9 indicating the number of nodes connected to LiFi AP and WiFi AP are overlapping.

VI. HARDWARE EXPERIMENTATION

The proof of concept using network emulation based experimentation in Section V has shown that our proposed algorithm outperforms other conventionally used access device selection mechanisms in today's end devices. In this section, similar hardware experimentation is performed to evaluate if the PM would still outperform the other techniques. The setup and results for such a hardware experiment are explained in this section.

A. SETUP

Due to hardware limitations, the setup is not as extensive as the network emulation setup. In this setup, we consider a network of four IoT nodes, two LiFi APs, and one WiFi AP. Raspberry Pi 4 and 3B+ are being used as IoT nodes, with a

built-in WiFi module and an external USB based LiFi module. As shown in Figure 10 setup at *aeroLiFi*,¹ two IoT nodes are placed under each LiFi AP, with a distance of 130 cm between them. The LiFi setup used is the LiFi-XC generation released by PureLiFi in 2018. Each LiFi-XC system consists of an access point and a USB dongle receiver, supporting a full-duplex link rate of 43 Mbps, and the ability to support eight users at a time, using time-division multiple access [45]. However, the received average data rate from experimentation in our setup is observed to be around 25 Mbps, which is further divided among two users. The hardware experimentation setup parameters are summarized in Table 5.

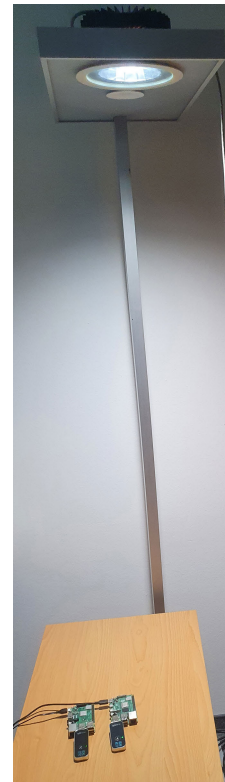


FIGURE 10. Placement of IoT nodes under LiFi AP.

TABLE 5. Hardware experimentation setup parameters.

Parameter	Value
Number of WiFi AP	1
Number of LiFi APs	2
Number of IoT Nodes	4
WiFi AP data rate	54 Mbps
LiFi AP data rate	43 Mbps
WiFi AP bandwidth	20 MHz
LiFi AP maximum transmit power	8 W
LiFi AP coverage area	2.8 - 3.5 m
LiFi receiver maximum transmit power	2.5 W
Distance between LiFi AP and receiver	130 cm
UDP flow size	1 - 13 Mbps

The experimental setup, shown in Figure 11, is split into two rooms. The first room has a WiFi AP and a LiFi AP,

¹[Online]. Available: <http://aerolifi.com/>, accessed in December 2020

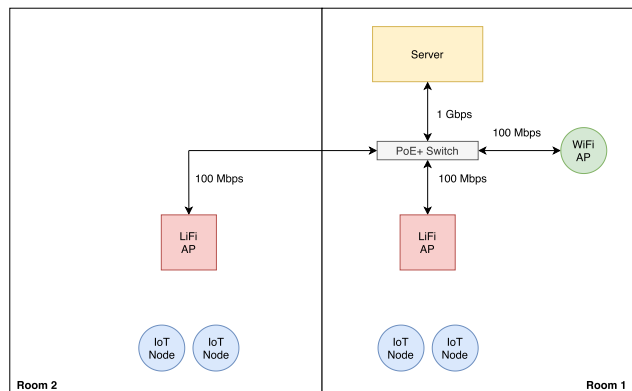


FIGURE 11. Hybrid WiFi and LiFi hardware experimentation setup.

whereas there is only a LiFi AP in the second room. All the APs used support Power-over-Ethernet. Therefore, a PoE+ switch is used to power the access points. A PoE+ switch is used instead of a PoE switch because LiFi APs require more power than provided by the PoE standard. We have also used another Raspberry Pi to act as an iperf server and a DHCP server. Both the WiFi and LiFi interfaces of the IoT nodes are in the same subnet.

We use Balena² IoT cloud platform to deploy Linux-based containerized applications on Raspberry Pi devices. For the Raspberry Pi server application, we deployed two Docker³ containers, one acting as a DHCP server, and the other as an iperf server. The Raspberry Pi client application also consists of two Docker containers, one for loading the kernel module of PureLiFi LiFi-XC drivers, and the other container is used as an iperf client. Although the selected TP-Link EAP115-Wall WiFi AP provides up to 300 Mbps with 802.11n mode, we used the 802.11 b/g mixed mode with a single channel of 20 MHz bandwidth to provide a maximum data rate of 54 Mbps, similar to our network emulation setup.

Similar to the network emulation setup, flow sizes of 1 to 13 Mbps are admitted to evaluate the network performance. Like network emulation analysis, each data point shows the mean of repetition of ten experiments, and the error bar represents the standard deviation. Similarly, iperf utility and ICMP packets are used to generate UDP traffic and measure the delay between the IoT node and the server, respectively.

B. RESULTS

We perform a similar network evaluation in hardware experimentation as in our network emulation setup. Due to different network size, topology, and link parameters, we do not expect to have exactly similar results, but similar trends. The PM tries to satisfy the QoS constraints of data rate and SNR while minimizing the delay. In B-SNR, nodes connect to the

AP offering the best SNR, while in M-BW, nodes connect based on the maximum link bandwidth.

First, the percentage of nodes that satisfy the QoS constraint of throughput with increasing flow size inserted per node is evaluated. Till 7 Mbps flow size admitted per node, no congestion occurs for any of the techniques. This is shown in Figure 12, with all techniques satisfying QoS requirements for 100% of nodes until 7 Mbps. The congestion occurs for all techniques as flow size increases beyond 7 Mbps. However, AP selection based on throughput QoS constraint in PM outperforms B-SNR and M-BW by better satisfying QoS requirements for all admitted flow sizes beyond 7 Mbps. The M-BW and B-SNR techniques perform nearly the same and are unable to satisfy the QoS requirements of any node at 11 Mbps and beyond.

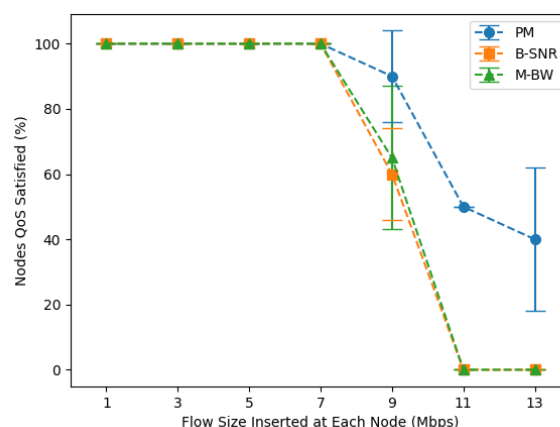


FIGURE 12. Percentage of QoS satisfied nodes.

As the QoS satisfied is based on the throughput constraint, the average throughput achieved per node would have a similar trend, as shown in Figure 13. The performance of an ideal network with infinite capacity and no congestion is shown with a red-line. As also seen in Figure 12, no congestion occurs until 9 Mbps for any of the techniques. The PM outperforms other techniques by providing a higher average throughput per node beyond 9 Mbps. B-SNR and M-BW perform similarly at flow sizes greater than 9 Mbps.

The average delay experienced per node is shown in Figure 14. Our PM outperforms other techniques at lower flow sizes as the network is not congested, and both LiFi and WiFi APs can satisfy QoS constraints. Therefore, PM selects the AP with the minimum delay. However, as the network congestion increases, the delay of PM becomes greater than M-BW because PM prioritizes the QoS constraint of throughput over the delay. The AP, which satisfies the QoS constraint is selected by PM, even if it has a higher delay than the other AP. For M-BW, the nodes in Room 1 connect to the WiFi AP, and the nodes in Room 2 connect to the LiFi AP, the average delay experienced at the nodes is the least because the nodes are always connected to the AP in the same room, i.e., smaller distance. Finally, for B-SNR, the delay experienced is the most because all nodes connect to their respective LiFi APs,

²[Online]. Available: <https://www.balena.io/>, accessed in December 2020

³[Online]. Available: <https://www.docker.com/why-docker>, accessed in December 2020

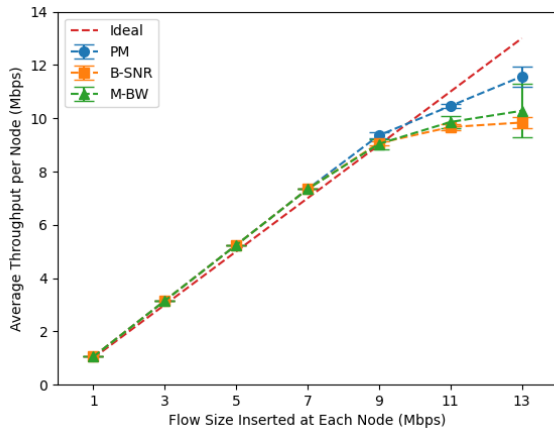


FIGURE 13. Average throughput per node.

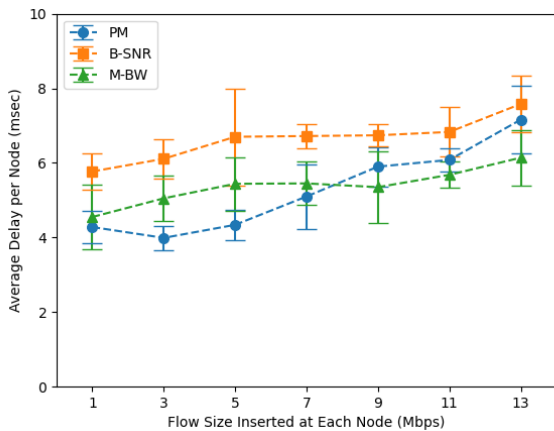


FIGURE 14. Average delay per node.

without utilizing the WiFi link, thus causing more congestion in the LiFi network. This congestion of the LiFi network and non-utilization of the WiFi network results in the highest average delay per node for B-SNR.

The nodes connected to LiFi and WiFi APs for B-SNR and M-BW are independent of the flow sizes, as shown in Figure 15. As the nodes are lying directly under the LiFi AP with no blockage, shadowing, and mobility, the B-SNR nodes would always connect to the LiFi AP, as the SNR received over the LiFi channel would always be better than the WiFi channel. For M-BW, it has been observed from the experimental evaluation that the WiFi link bandwidth is greater than LiFi AP for nodes in Room 1. These nodes are always connected to the WiFi AP. Since nodes in Room 2 have a greater distance from the WiFi AP, they perceive a lower data rate from WiFi AP than LiFi AP due to different modulation and coding scheme (MCS), code rate selection, and signal strengths. Therefore, in M-BW, the IoT nodes in Room 2 will always connect to the LiFi AP. Since two nodes connect to both LiFi AP and WiFi AP in M-BW, the black-lines indicating the number of nodes connected to each AP are overlapping. In the PM, the nodes connected to LiFi and WiFi AP vary as the flow size changes. The nodes are dynamically allocated to the AP best satisfying the

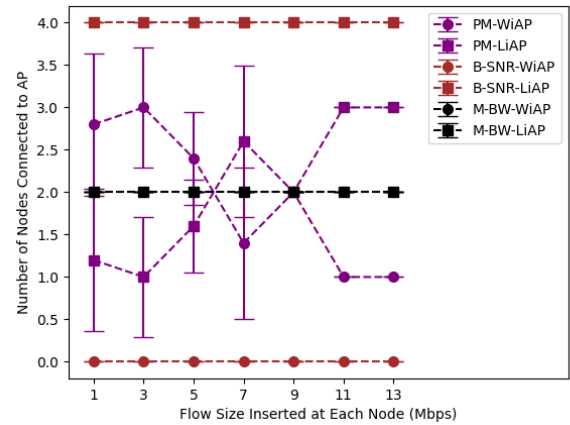


FIGURE 15. Nodes distribution among WiFi and LiFi APs.

QoS constraint of throughput, while minimizing delay. As flow size increases, the single WiFi link becomes congested, resulting in more nodes connecting to LiFi APs at higher flow sizes. On average, more nodes connect to the WiFi AP till flow size of 9 Mbps, with just one node connecting on average to the WiFi AP beyond 9 Mbps.

VII. DISCUSSION

The outcome of the five research questions, defined in Section II-A, are discussed in this section. This is followed by a comparison of results obtained from network emulation and hardware experiments in Section V-B and Section VI-B, respectively.

RQ1 aimed to formulate a client-based access device selection algorithm considering client-specific QoS requirements for access device selection in B5G IoT networks. Such an algorithm is formulated in Algorithm 1. As Algorithm 1 works for multi-RAT and multihoming scenarios, RQ2 is also fulfilled.

RQ3 intended to evaluate the computational efficiency and convergence of the algorithm. The algorithmic complexity of the proposed algorithm is evaluated to be exponential. However, the complexity reduces to cubic in the absence of multihoming and reduces to linear when only two access devices are considered. Regarding convergence, the algorithm converges to Nash Equilibrium with the average Pareto-efficiency gap bound defined by (24) and (25).

RQ4 aimed to analyze the LiFi channel data rate and SNR for different configurations. The outcome of this analysis shows a decrease in SNR and data rate with increasing distance between the AP and the receiver. LiFi channel is analyzed by varying the angle of irradiance (Figure 2), angle of incidence (Figure 3), and half-intensity radiation angle (Figure 4).

The RQ5 targeted evaluating the performance of the proposed algorithm compared to other access device selection techniques. The proposed algorithm is compared with two other access device selection techniques; B-SNR and M-BW. Using network-emulation and hardware experimentation setups, it is shown that the proposed algorithm outperforms

the other two techniques by providing a better QoS satisfied ratio, a higher average throughput, and a lower average delay. The proposed algorithm improves the QoS satisfied ratio by up to 32.66% in network emulation and up to 50% in hardware experimentations.

Comparing the results of network emulation and hardware experimentations, it can be seen that PM outperforms B-SNR and M-BW for QoS provisioning as well as provides a higher average throughput per node in both setups. However, network congestion and a decrease in QoS satisfied ratio occur earlier at 7 Mbps in network emulation than 9 Mbps in hardware experimentation. This is because of the difference in total system bandwidth between network emulation and hardware setups. Network emulation setup has a total system bandwidth of 534 Mbps for 24 nodes (22.25 Mbps per node), whereas hardware setup has a total system bandwidth of 140 Mbps for four nodes (35 Mbps per node). As a lower per node bandwidth is available in network emulation setup, results show earlier congestion. The PM provides the lowest per node delay in network emulation results for all flow sizes. The delay per node for PM is the lowest till 7 Mbps for hardware experimentation, but becomes greater than M-BW at 9 Mbps and beyond. However, since this difference in the delay between PM and M-BW is insignificant and is less than 1 msec, the difference can be neglected. The magnitude of delay of around 40 msec in network emulation and around 4 msec in hardware experimentation is incomparable because of different network topology and different link delays in both setups. An exactly same percentage of nodes associate with WiFi AP and LiFi AP for B-SNR and M-BW in both network emulation and hardware experimentation setups, as the AP selection in these techniques is independent of flow size and network size. In PM, when the flow size increases beyond 9 Mbps, four out of 24 nodes (16.67%) and one out of four nodes (25%) connect to WiFi AP in network emulation and hardware experimentations, respectively. This shows a consistent behavior of less number of nodes connecting to WiFi AP at higher flow sizes. However, the exact percentages should not be compared due to different network sizes and WiFi link load in both setups. The same 54 Mbps WiFi link is shared between 24 nodes in the network emulation setup, and among only four nodes in the hardware experimentation setup.

VIII. CONCLUSION

In this paper, we presented a client-based access device selection mechanism for QoS provisioning in Beyond 5G IoT networks while considering multiple-RATs and heterogeneous QoS requirements of IoT nodes. In contrast to conventionally available access device selection approaches, the proposed technique allowed IoT nodes to specify their own QoS requirements. A novel algorithm is proposed for such an access device selection mechanism with cubic complexity in the absence of multihoming and linear-complexity when only two access devices are used. The game-theoretic analysis showed that such a non-cooperative game is guaranteed to

converge to Nash equilibrium. The proposed algorithm's performance is evaluated in a hybrid WiFi and LiFi IoT network scenario using Mininet based network emulation and hardware experimental evaluations. The comparison with other conventional access device selection techniques, including B-SNR and M-BW, shows that our proposed algorithm outperforms these techniques.

As future work, experiments with multiple WiFi APs could be considered, in addition to multiple LiFi APs. The AP selection for mobile nodes with various mobility models would also be an interesting topic to investigate. As energy efficiency is a concern for IoT nodes, another possible future work direction could be AP selection in hybrid WiFi and LiFi networks while jointly maximizing energy efficiency and QoS.

ACKNOWLEDGMENT

The authors would like to thank the technical assistance, software and hardware support, and valuable input provided by aeroLiFi GmbH for the completion of this work.

REFERENCES

- [1] Y. Yuan, Y. Zhao, B. Zong, and S. Parolari, "Potential key technologies for 6G mobile communications," *Sci. China Inf. Sci.*, vol. 63, no. 8, pp. 1–19, Aug. 2020.
- [2] G. Forecast, "Cisco visual networking index: Global mobile data traffic forecast update, 2017–2022," *Update*, vol. 2017, p. 2022, Feb. 2019.
- [3] R. A. Aljiznawi, N. H. Alkhazaali, S. Q. Jabbar, and D. J. Kadhim, "Quality of service (QoS) for 5G networks," *Int. J. Future Comput. Commun.*, vol. 6, no. 1, p. 27, 2017.
- [4] M. Asad, A. Basit, S. Qaisar, and M. Ali, "Beyond 5G: Hybrid end-to-end quality of service provisioning in heterogeneous IoT networks," *IEEE Access*, vol. 8, pp. 192320–192338, 2020.
- [5] Y. Xiao, M. Krunz, and T. Shu, "Multi-operator network sharing for massive IoT," *IEEE Commun. Mag.*, vol. 57, no. 4, pp. 96–101, Apr. 2019.
- [6] X. Chen, D. Wing Kwan Ng, W. Yu, E. G. Larsson, N. Al-Dhahir, and R. Schober, "Massive access for 5G and beyond," 2020, *arXiv:2002.03491*. [Online]. Available: <http://arxiv.org/abs/2002.03491>
- [7] B. Liu, C. Liu, and M. Peng, "Resource allocation for energy-efficient MEC in NOMA-enabled massive IoT networks," *IEEE J. Sel. Areas Commun.*, early access, Aug. 24, 2020, doi: 10.1109/JSAC.2020.3018809.
- [8] G. A. Akpakwu, B. J. Silva, G. P. Hancke, and A. M. Abu-Mahfouz, "A survey on 5G networks for the Internet of Things: Communication technologies and challenges," *IEEE Access*, vol. 6, pp. 3619–3647, 2018.
- [9] M. R. Palattella, M. Dohler, A. Grieco, G. Rizzo, J. Torsner, T. Engel, and L. Ladid, "Internet of Things in the 5G era: Enablers, architecture, and business models," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 3, pp. 510–527, Mar. 2016.
- [10] M. Wang, J. Chen, E. Aryafar, and M. Chiang, "A survey of client-controlled HetNets for 5G," *IEEE Access*, vol. 5, pp. 2842–2854, 2017.
- [11] S. Idris, U. Mohammed, J. Sanusi, and S. Thomas, "Visible light communication: A potential 5G and beyond communication technology," in *Proc. 15th Int. Conf. Electron., Comput. Commun. (ICECCO)*, Dec. 2019, pp. 1–6.
- [12] H. Haas, L. Yin, Y. Wang, and C. Chen, "What is LiFi?" *J. Lightw. Technol.*, vol. 34, no. 6, pp. 1533–1544, Mar. 15, 2016.
- [13] M. S. Islim, R. X. Ferreira, X. He, E. Xie, S. Videv, S. Viola, S. Watson, N. Bamiedakis, R. V. Penty, I. H. White, A. E. Kelly, E. Gu, H. Haas, and M. D. Dawson, "Towards 10 Gb/s orthogonal frequency division multiplexing-based visible light communication using a GaN violet micro-LED," *Photon. Res.*, vol. 5, no. 2, p. A35, 2017.
- [14] X. Wu, M. Dehghani Soltani, L. Zhou, M. Safari, and H. Haas, "Hybrid LiFi and WiFi networks: A survey," 2020, *arXiv:2001.04840*. [Online]. Available: <http://arxiv.org/abs/2001.04840>
- [15] D. Bethanabhotla, O. Y. Bursalioglu, H. C. Papadopoulos, and G. Caire, "Optimal user-cell association for massive MIMO wireless networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 1835–1850, Mar. 2016.

- [16] A. Basit, S. Qaisar, M. Ali, M. Naeem, M. Bruyere, and J. J. P. C. Rodrigues, "Interconnecting networks with optimized service provisioning," *Telecommun. Syst.*, vol. 73, no. 2, pp. 223–239, Feb. 2020.
- [17] M. S. Al-Abiad, A. Douik, S. Sorour, and J. Hossain, "Throughput maximization in cloud-radio access networks using cross-layer network coding," *IEEE Trans. Mobile Comput.*, early access, Jul. 29, 2020, doi: 10.1109/TMC.2020.3012935.
- [18] H. S. Oh, D. G. Jeong, and W. S. Jeon, "Joint radio resource management of channel-assignment and user-association for load balancing in dense WLAN environment," *IEEE Access*, vol. 8, pp. 69615–69628, 2020.
- [19] W. Li, S. Wang, Y. Cui, X. Cheng, R. Xin, M. A. Al-Rodhaan, and A. Al-Dhelaan, "AP association for proportional fairness in multirate WLANs," *IEEE/ACM Trans. Netw.*, vol. 22, no. 1, pp. 191–202, Feb. 2014.
- [20] X. Wu, M. Safari, and H. Haas, "Access point selection for hybrid Li-Fi and Wi-Fi networks," *IEEE Trans. Commun.*, vol. 65, no. 12, pp. 5375–5385, Dec. 2017.
- [21] M. Hammouda, S. Akin, A. M. Vegni, H. Haas, and J. Peissig, "Link selection in hybrid RF/VLC systems under statistical queueing constraints," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2738–2754, Apr. 2018.
- [22] J.-P. Huang, C.-C. Huang-Fu, and P.-Y. Cheng, "Dual sim dual standby user equipment rat selection," U.S. Patent 10 681 604, Jun. 9, 2020.
- [23] B. Sliwa, R. Falkenberg, and C. Wietfeld, "Towards cooperative data rate prediction for future mobile and vehicular 6G networks," in *Proc. 2nd 6G Wireless Summit (6G SUMMIT)*, Mar. 2020, pp. 1–5.
- [24] A. Habbal, S. I. Goudar, and S. Hassan, "A context-aware radio access technology selection mechanism in 5G mobile network for smart city applications," *J. Netw. Comput. Appl.*, vol. 135, pp. 97–107, Jun. 2019.
- [25] V. Passas, V. Miliotis, N. Makris, and T. Korakis, "Dynamic RAT selection and pricing for efficient traffic allocation in 5G HetNets," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.
- [26] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Internet of Things applications: A systematic review," *Comput. Netw.*, vol. 148, pp. 241–261, Jan. 2019.
- [27] W. A. Jabbar, H. K. Shang, S. N. I. S. Hamid, A. A. Almohammed, R. M. Ramli, and M. A. H. Ali, "IoT-BBMS: Internet of Things-based baby monitoring system for smart cradle," *IEEE Access*, vol. 7, pp. 93791–93805, 2019.
- [28] W. K. Saad, Y. Hashim, and W. A. Jabbar, "Design and implementation of portable smart wireless pedestrian crossing control system," *IEEE Access*, vol. 8, pp. 106109–106120, 2020.
- [29] M. Abujubbeh, F. Al-Turjman, and M. Fahrioglu, "Software-defined wireless sensor networks in smart grids: An overview," *Sustain. Cities Soc.*, vol. 51, Nov. 2019, Art. no. 101754.
- [30] W. A. Jabbar, T. K. Kian, R. M. Ramli, S. N. Zubir, N. S. M. Zamrizaman, M. Balfaqih, V. Shepelev, and S. Alharbi, "Design and fabrication of smart home with Internet of Things enabled automation system," *IEEE Access*, vol. 7, pp. 144059–144074, 2019.
- [31] G. Dandachi, S. Eddine Elayoubi, T. Chahed, and N. Chendeb, "Network-centric versus user-centric multihoming strategies in LTE/WiFi networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 5, pp. 4188–4199, May 2017.
- [32] D. D. Nguyen, H. X. Nguyen, and L. B. White, "Evaluating performance of RAT selection algorithms for 5G hetnets," *IEEE Access*, vol. 6, pp. 61212–61222, 2018.
- [33] Z. Han, D. Niyato, and W. Saad, *Game Theory in Wireless and Communication Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2011.
- [34] D. Niyato and E. Hossain, "Dynamics of network selection in heterogeneous wireless networks: An evolutionary game approach," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 2008–2017, May 2009.
- [35] B. Wu, Z. Ma, S. Poslad, and W. Zhang, "An efficient wireless access point selection algorithm for location determination based on RSSI interval overlap degree determination," in *Proc. Wireless Telecommun. Symp. (WTS)*, Apr. 2018, pp. 1–8.
- [36] Y. Wang, D. A. Basnayaka, X. Wu, and H. Haas, "Optimization of load balancing in hybrid LiFi/RF networks," *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1708–1720, Apr. 2017.
- [37] Y. Wang, X. Wu, and H. Haas, "Load balancing game with shadowing effect for indoor hybrid LiFi/RF networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 4, pp. 2366–2378, Apr. 2017.
- [38] M. Z. Chowdhury, M. T. Hossain, M. K. Hasan, and Y. M. Jang, "Integrated RF/optical wireless networks for improving QoS in indoor and transportation applications," *Wireless Pers. Commun.*, vol. 107, no. 3, pp. 1401–1430, Aug. 2019.
- [39] X. Wu, D. C. O'Brien, X. Deng, and J.-P.-M. G. Linnartz, "Smart handover for hybrid LiFi and WiFi networks," *IEEE Trans. Wireless Commun.*, early access, Sep. 4, 2020, doi: 10.1109/TWC.2020.3020160.
- [40] M. Asad, S. Qaisar, and A. Basit, "Client based access layer QoS provisioning in beyond 5G IoT networks," in *Proc. 3rd Int. Conf. Adv. Commun. Technol. Netw. (CommNet)*, Sep. 2020, pp. 1–8.
- [41] X. Wu and H. Haas, "Load balancing for hybrid LiFi and WiFi networks: To tackle user mobility and light-path blockage," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1675–1683, Mar. 2020.
- [42] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD, USA: The Johns Hopkins Univ. Press, 1996.
- [43] E. Aryafar, A. Keshavarz-Haddad, M. Wang, and M. Chiang, "RAT selection games in HetNets," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 998–1006.
- [44] R. L. S. de Oliveira, C. M. Schweitzer, A. A. Shinoda, and L. Rodrigues Prete, "Using mininet for emulation and prototyping software-defined networks," in *Proc. IEEE Colombian Conf. Commun. Comput. (COLCOM)*, Jun. 2014, pp. 1–6.
- [45] L. I. Albraheem, L. H. Alhudaithy, A. A. Aljaser, M. R. Aldhafian, and G. M. Bahliwah, "Toward designing a Li-Fi-based hierarchical IoT architecture," *IEEE Access*, vol. 6, pp. 40811–40825, 2018.



MUHAMMAD ASAD (Graduate Student Member, IEEE) received the bachelor's degree in telecommunication engineering from the University of Engineering and Technology Taxila, Pakistan, in 2011, and the master's degree in communications engineering from the Technical University of Munich, Germany, in 2013. He is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Computer Science, National University of Science and Technology (NUST). His research interests include software-defined 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, USA, in 2005 and 2009, respectively. He is currently serving as an Assistant Professor with the School of Electrical Engineering Computer Science (SEECS), National University of Sciences Technology (NUST), Pakistan. He is also the Lead Researcher and the Founding Director of the CoN-NekT Lab: Research Laboratory of Communications, Networks and Multimedia, NUST. As of September 2011, he is 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 has published over 45 articles at reputed international venues with a vast amount of work in the pipeline.



ABDUL BASIT received the bachelor's degree in computer science from the University of Management and Technology, Lahore, in 2003, the M.S. degree in software engineering from the National University of Sciences and Technology, in 2005, and the Ph.D. degree from the Department of Computing, School of Electrical Engineering and Computer Sciences. He has also served in various positions and organizations in academia and industry for more than seven years. His research interests include programmable networking, network modeling and optimization, and industrial applications of geographical information systems.

...