Bidirectional LiFi Attocell Access Point Slicing Scheme

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

Bidirectional LiFi Attocell Access Point Slicing Scheme

Hamada Alshaer and Harald Haas

***Abstract*—LiFi attocell access networks will be deployed everywhere to support diverse applications and service provisioning to various end-users. The LiFi infrastructure providers will need to offer LiFi access points (APs) resources as a service. This, however, requires a research challenge to be solved to dynamically and effectively allocate resources among service providers (SPs) , while guaranteeing performance isolation among them and their respective users. This paper introduces an autonomic resource slicing (virtualization) scheme, which realizes autonomic management and configuration of virtual APs, in a LiFi attocell access network, based on SPs and their users service requirements. The developed scheme comprises of traffic analysis and classification, a local AP controller, downlink and uplink slice resources manager, traffic measurement and information collection modules. It also contains a hybrid medium access protocol and an extended token bucket fair queueing algorithm to support uplink access virtualization and spectrum slicing. The proposed resource slicing scheme collects and analyzes the traffic statistics of the different applications supported on the slices defined in each LiFi AP and distributes the available resources fairly and proportionally among them. It uses a control algorithm to adjust the minimum contention window of user devices to achieve the target throughput and ensure airtime fairness among SPs and their users. The developed scheme has been extensively evaluated using OMNeT++. The obtained results show various resource slicing capabilities to support differentiated services and performance isolation.**

***Index Terms*—LiFi AP, LiFi attocell networks, resource slicing, MAC access virtualization, hypervisor, LiFi traffic, VLC, SDN.**

I. Introduction

INTERNET infrastructure providers (IIPs) aim to extend the high-speed internet to diverse and challenging places. Service providers (SPs) aim to provide reliable services anywhere. These cover users at coffee shops, shopping malls, exhibition avenues, business and residential buildings, remote villages, airports, airplane cabins and underground metro and submarines. The LiFi optical communication access technology [1] can meet both SPs and IIPs objectives by turning any light bulb into a wireless access point (AP) to enable users to communicate and transmit data at optical (light) speed. The LiFi APs can be networked through the existing lighting networks, or any other high-speed optical transport networks, to form LiFi attocell access networks. Thus, the emerging LiFi optical wireless access technology has created a novel internet and wireless communication service infrastructure, which provides massive wireless network resources [1], [2]. However, these should be managed effectively by IIPs and SPs to support the unprecedented growth and various requirements of wireless data communication and diverse 5G real-time services and applications.

H. Alshaer and H. Haas are with the LiFi R&D Centre, Institute for Digital Communications, School of Engineering, The University of Edinburgh, EH9 3JL, Edinburgh, UK.

Email: {h.alshaer,h.haas}@ed.ac.uk.

Fig. 1. A use case on indoor LiFi attocell access point slicing.

The software-defined networking (SDN)-enabled [3], [4] and virtualized [5], [6] LiFi attocell APs facilitate the development of suitable LiFi access network architectures for resource abstraction, programmability and isolation. These represent a viable way to develop adequate mechanisms for resource management and support novel services using separate LiFi AP slices, as shown in Fig. 1. A slice can be defined as a set of resources which can be resized and programmed to support functional services to SPs and their users. For example, Table I describes a virtualized LiFi attocell network consisting of N LiFi APs, in which each can dynamically support M slices at most. A slice comprises a set of physical resource blocks (RBs) and some buffer space, which can be dynamically resized. The different 5G standardization bodies work on defining the requirements and architectures that will enable network slicing in 5G mobile access networks. The general consensus is to define network slice types and 5G service classification. A 5G service may fall into three categories: i) extreme mobile broadband (xMBB) which requires both high data rates and low latency in some areas, and reliable broadband over large areas, ii) massive machine-type communication (mMTC), which needs wireless connectivity for massive deployment devices and iii) ultra-reliable and low-latency communications (uRLLC) or ultra-reliable MTC, which covers all services requiring ultra-low latency connections with a certain level of reliability [7].

The LiFi attocell technology specifies both the physical layer and the medium access control (MAC) protocols. The physical layer adapts the modulation and coding scheme (MCS) level based on the channel condition, signal-to-interference-plus-noise ratio (SINR), of each user device (UD)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

TABLE I
LIFI AP SLICES.

AP	Slice spectrum (RBs)	Slice buffer (bytes)
	S_1 ,S_2,. . . ,S_M	B_1,B_2,. . . , B_M
AP_1	24,15, . . . ,40	70 ,250, . . . , 200
⋮	⋮	⋮
AP_N	28,28, . . . ,35	65 ,180, . . . , 400

to improve the available spectrum efficiency. The MAC layer distributes the available spectrum among the wireless UDs using a multi-user access protocol [2], [8]. This schedules the access to the LiFi air interface among the UDs using the time and/or frequency resources in downlink and uplink directions. A SDN function decouples the control plane of the LiFi air interface from its data plane. The control plane specifies the scheduling logic and service provisioning policy, whereas the data plane applies the scheduling logic and the traffic processing and handling policies. A virtualization layer between these control and data planes provides the virtualization function parameters of each slice configured in a virtualized LiFi AP for the control plane. The SDN function controls the different slices and programs their resource allocation through configuring their virtualization function parameters. Thus, the SDN and virtualization functions can enable full virtualization for the air interface of a LiFi AP.

A two-step Orthogonal frequency-division multiple access (OFDMA) scheduler can be deployed in each LiFi AP to allocate the downlink resources of LiFi air interface among the supported slices per frame. The virtualization of uplink air interface of LiFi APs is based on the nature of the uplink MAC protocol. For example, the uplink slices can be isolated by deploying a time division multiple access (TDMA) MAC protocol, which assigns an exclusive timeshare for each slice based on its requirements [9]. This, however, may result in under-utilizing the reserved timeshare of a slice because some of them may be left unused in the presence of inactive users of each slice [10], [9]. A random access protocol, however, such as carrier sense multiple access with collision avoidance (CSMA/CA) has a more dynamic and opportunistic nature. It can adaptively manage the timeshare of each slice according to the number of active users. A hybrid TDMA/CSMA-CA MAC protocol can overcome the utilization inefficiency of the TDMA part, while maximizing the expected sum throughput through the CSMA/CA part, because of its opportunistic nature [11].

The downlink spectrum can be managed among the SPs and their respective UDs at the AP level. However, the uplink spectrum can be managed at the UDs level through controlling their uplink access channel probability and contention window (CW). For example, a SP_i can obtain a double throughput of SP_j, if the SP_j is assigned a double CW of SP_i [12]. The uplink scheduler algorithm dynamically reserves a variable number of time-slots in each TDMA-CSMA/CA superframe based on the traffic demand of users and required resource reservation of their slice. The unused time-slots in superframes are allocated to unscheduled users in a contention-based access mode using CSMA/CA protocol [13], [11].

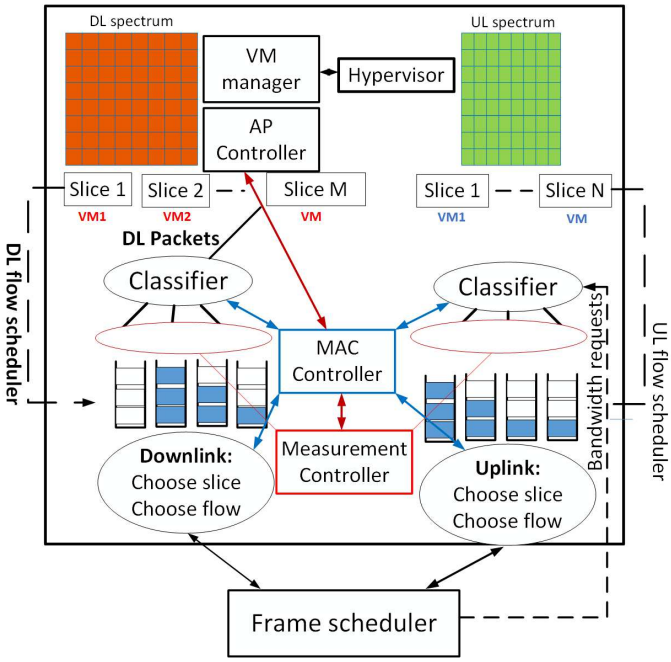

Fig. 2. A modified basic design of LiFi AP attocell virtualization [14].

The rest of this paper is organized as follows. In Section II, the research problem is stated while clarifying the contributions of this paper compared to the main relevant related work. In Section III, the LiFi AP virtualization model is described. In Section IV and Section V, the downlink spectrum slicing and MAC virtualization approaches are discussed. In Section VI, the uplink spectrum virtualization is explained. In Section VII, the SDN application to a virtualized LiFi access network is introduced. In Section VIII, the realization complexity of the proposed DAS_2 scheme is explained. In Section IX, the simulation results are presented. In Section X, the conclusions and next research steps are discussed.

II. PROBLEM STATEMENT & RELATED WORK

The LiFi attocell channel characteristics are different from those of a typical RF channel. The LiFi channel gain may not fluctuate when the user is stationary. However, the coverage of a LiFi AP signal is smaller and the UDs may experience signal obstructions, as they fall in and out of the FOV (field-of-view) of transmitters/receivers. While, stationary LiFi UDs generally experience more or less constant throughput, the UDs with mobility and continuous orientation variations may experience a drop in their throughput. Thus, the challenge is how to achieve an effective and fair slicing of LiFi attocell resources, independent of capacity variability and availability of LiFi AP resources. Also, each 5G service may have different requirements in terms of latency, bandwidth, security and reliability, which requires a viable way to provide a network slice tailored to each service.

A simple resource allocation approach partitions the spectrum band into multiple sub-bands and assigns each SP operator a sub-band permanently (i.e a fixed set of resource units). This dedicated resource sharing model separates the

control and user traffic of the different slices, which reduces slice elasticity as well as scalability, and limits the resource multiplexing gain. Thus, this model does not allow intra/inter operator resource borrowing. On the other hand, a hypervisor [4] can be deployed on top of the LiFi MAC and physical resources to dynamically allocate resources to virtual operators according to their traffic demands and users' channel conditions. This dynamic resource sharing model allows the different slices to share the physical resources and same control plane. However, the physical, MAC and network resources of LiFi attocell access networks should be allocated to virtual operators independent of each other in a scalable and isolated manner.

Table. II summarizes the main developed approaches to support wireless access networks slicing and virtualization. The concept of network virtualization substrate (NVS) is introduced in [14], as shown in Fig. 2. It proposes a slice scheduler, which dynamically allocates MAC layer resources to slices so that each slice (operator) can achieve its reserved physical resource units or guaranteed throughput rate. In [15] the authors extend NVS [14] to support partial resource reservation using a two-step admission control scheme. This reserves a minimum resource for each operator and shares the remaining spectrum among the users of the different slices following the first come first served scheduling discipline. However, the proposed slicing scheduler and two-step admission control approach cannot be deployed to support an arbitrary number of slices. In addition to the classical mobile broadband application services, the proposed virtual architecture should support some other vertical industry applications. These may include automotive systems, smart grids, smart city and IoT services. A MAC protocol is designed to arbitrate medium access among multiple users, but it should be redesigned to support a variable number of slices based on resource sharing readjustment.

This paper introduces an integrated resource virtualization framework for virtualizing LiFi attocell AP resources to support a dynamic number of slices. An autonomic virtualization scheme is developed to support autonomic downlink and uplink resource allocation and sharing among the different slices based on their traffic loads, owners priority and provisioned service requirements. The developed dynamic LiFi AP spectrum sharing (DAS_2) scheme runs across LiFi physical/MAC layers. It dynamically shares uplink and downlink resources based on information obtained from the physical, MAC and application layers, following a cross-layers design model. This enables UDs to receive services from their slices based on downlink spectrum units control and airtime control that ensures service isolation and fairness among UDs within and across slices. The proposed resource virtualization framework has been extensively evaluated to demonstrate the main benefits of virtualization and compare the performance of a virtualized LiFi AP to a non-virtualized LiFi AP.

In contrast to the state-of-the art work above, the proposed virtualization scheme enforces dynamic sharing of LiFi downlink and uplink resources slices among SPs and their respective user groups. It runs based on simple abstractions for the MAC and physical resources of LiFi APs to reliably support an arbitrary number of slices. An airtime control abstraction function selects a specific physical transmission rate for each supported slice, while guaranteeing a minimum service rate for active UDs. It dynamically allocates resources to the different SPs based on their traffic load and the availability of LiFi AP downlink and uplink resources. The downlink scheduler can support multiple service classes on each slice defined in a LiFi AP. It is associated with an admission control to protect the QoS of active slices. The performance results demonstrate the rigorousness, flexibility and elasticity of the proposed slicing scheme in sharing resources among slices defined in a LiFi AP.

III. LiFi AP Virtualization System model

The visible light communication (VLC) downlink and infrared (IR) uplink spectrum, as well as the buffer of a LiFi AP can be dynamically organized into slices in Docker Containers (DCs), as shown in Fig. 2. Each DC is characterized by its allocated spectrum capacity and buffer space, as explained in Table I. A DC can be self-contained with its resource scheduler and MAC protocol to serve its subscribed UDs following a dynamic sharing resource approach. The hypervisor uses the DC (virtual machine (VM)) monitor/manager to monitor, manage and dynamically coordinate access to physical and MAC resources among supported slices. This maintains isolation among physical resources allocated to active slices (DCs) per downlink and uplink MAC frames [4]. It also allows resource borrowing among slices. The AP controller arbitrates the access and control traffic admission to the physical resources, which are allocated in DCs. We assume that the LiFi AP uses physical layer perfect channel state information (CSI) and an adaptive modulation and coding (AMC) scheme. These cope with any degradation in channel quality to maximize the spectrum throughput. OFDMA and TDMA-CSMA/CA [11] MAC protocols are deployed to support multiple users access in LiFi downlink and uplink directions, respectively. A UD is equipped with a wireless transceiver that supports the VLC spectrum in downlink and IR in uplink directions and IEEE 802.11 WiFi. The UDs only communicate through WiFi when they are out of the LiFi AP coverage or experience light channel blocking.

The downlink spectrum is quantified in frequency and time domains in terms of minimum spectrum units called RBs and accessed in time slots. A single RB may consist of one or multiple subcarriers according to the wireless technology. The total number of subcarriers available on the LiFi downlink can be calculated by multiplying the number of RBs, N_{RB}^{DL}, by the number of subcarriers per RB, N_{sc}^{RB}. Thus, the bandwidth of a single subcarrier can be calculated by dividing the total coherent LiFi downlink bandwidth,B, by the number of downlink spectrum subcarriers, such as: $b_{sc} = \frac{B}{N_{RB}^{DL} \times N_{sc}^{RB}}$. The subcarrier bandwidth is assumed to be small, compared to the coherent wireless channel bandwidth. The LiFi downlink bandwidth,"B" Hz, and the buffer space,"Q", are dynamically shared among a set of slices, M. A SP is allocated a slice which can be shared among its users. A downlink virtual request, r, is represented by (ID, p, N_{RB}, t_s), where ID

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

TABLE II
WIRELESS NETWORKS RESOURCES SLICING AND VIRTUALIZATION RELATED WORK [5]

Technology	objective	uplink	downlink	Proof of concept
WiMax [16]	slicing framework	slice and flow scheduling	slice and flow scheduling	NVS
LTE[17]	resources assignment-based predefined load balancing levels	NO	downlink PRBs scheduler	NO
WiMax [18]	air-time fairness	NO	traffic shaping	WiMax
IEEE 802.16 (WiMax) [18],[19]	Air-tim fairness and slice isolation	NO	traffic shaping	WiMax Prototype
IEEE 802.16 (WiMax) [20]	Slice isolation	slice scheduling and traffic shaping	NVS with feedback based rate control	CellSlice
IEEE 802.11 (WiFi) [21]	Space and time-based slicing Click based policy manager	adjustment 802.11 EDCA parameters	Slice Scheduling traffic shaping	ORBIT NO
IEEE 802.11 (WiFi) [22]	UE virtualization to support multiple vMACs	slice scheduling NO	NO NO	Virtual WiFi
IEEE 802.11 (WiFi) [23]	efficient experiments isolation on an experimental wireless testbed	TDM-based slice scheduling	TDM-based slice scheduling	ORBIT Software interface
IEEE 802.11 [24]	TDMA-based slicing	NO	airtime based slicing	NO
IEEE 802.11 [25]	Uplink slicing	airtime-based slicing	NO	SplitAP
LTE RAN [15]	LTE RAN sharing	NO	partial sharing	NO

represents the slice identifier which indicates the slice type (e.g., which scheduler) in addition to the tenant ID who is in charge of the slice , p represents the priority level, N_{RB} represents the minimum number of requested RBs, and t_s represents the minimum number of required time slots.

The time access to the IR uplink spectrum of LiFi APs is divided into fixed-length hybrid frames. A hybrid frame consists of three phases: head (synchronization)transmission, TDMA phase and CSMA/CA phase, as shown in Fig. 3 [26]. The head of each frame has a sequence that synchronizes the transmitter clock of LiFi APs with that of UDs. The time slots of the hybrid TDMA-CSMA/CA frame are allocated adaptively to the different uplink slices based on their service requirements. High priority UDs send traffic during the time slots which are allocated to the TDMA phase. Whereas low priority UDs send traffic during the remaining time slots allocated to the CSMA/CA phase. In each superframe, t, a maximum number of time slots, $S_{max}(t)$, are dynamically allocated to the TDMA phase based on the total high priority traffic and QoS required for each slice.

The CSMA/CA protocol runs with each time-slot divided into t_{ts} back-off time units. The MAC controller monitors the average buffer level, $\overline{q_{tk}}$, of UD k, $\forall k \in K$, at the beginning of superframe t. It maintains the average buffer level information for UD k in the form $(\overline{q_{tk}}, f_{tk})$, where f_{tk} is the number of superframes which have passed after the latest report was received; and $\overline{q_k(t-1)}$ is the buffer level of UD k received in the latest report. The MAC controller estimates the average buffer level of UD k as follows:

$$\overline{q_k}(t) = \overline{q_k}(t-1) + \lfloor \lambda_{tk} f_{tk} \rfloor; \quad (1)$$

where λ_{tk} denotes the mean arrival rate of traffic generated by UD k per superframe t. The buffer status of UD k is determined as, $B_{tk} = \overline{q_k}(t)$, if $\overline{q_k}(t) < B_{max}$; otherwise $B_{tk} = B_{max}$. $\overline{q_k}(t)$ might be higher than B_{max} [11]. It sorts the UDs in ascending order of their buffer levels.

A downlink or an uplink flow request is admitted into a slice if its requested resources are successfully allocated. The admitted flow retains its allocated resources until it stops. If the flow request can be admitted during a given time slot, it remains in the buffer to retry admission in the next time-slot, if possible. When the flow request is not successfully admitted after a number of time-slots, it is removed from the buffer and the flow request is rejected. The maximum delay can be different for every priority slice request level. The LiFi AP maintains a mapping entry for each slice ID and its corresponding IP address, which is communicated by the slice orchestrator during the slicing instantiation process. The users of the different slices are associated with their virtualized LiFi AP (slice) through binding their IP address with a correspondent extension service set identifier (ESSID) advertised in the synchronization beacons [27], [28].

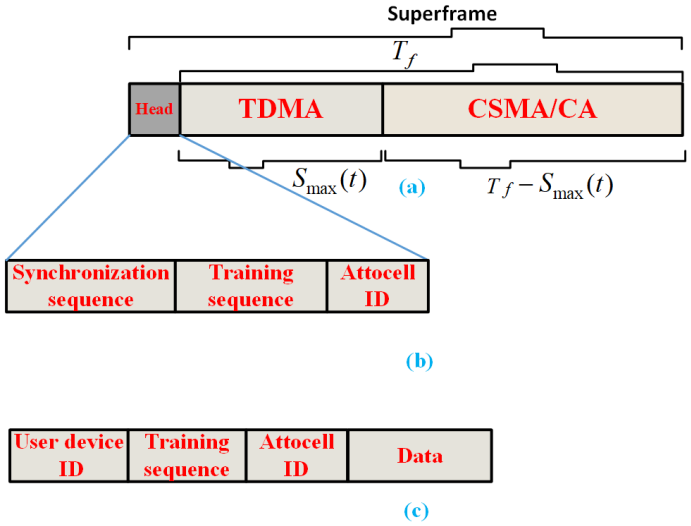

Fig. 3. Frame structure in LiFi system.(a) Hybrid uplink frame structure [26]. (b) Attocell data frame structure. (c) Feedback data frame structure.

IV. DOWNLINK RESOURCES SLICING

A gradient-based OFDMA scheduler allocates resources to supported slices in each LiFi attocell AP. Downlink traffic packets are policed and classified based on their slice ID.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

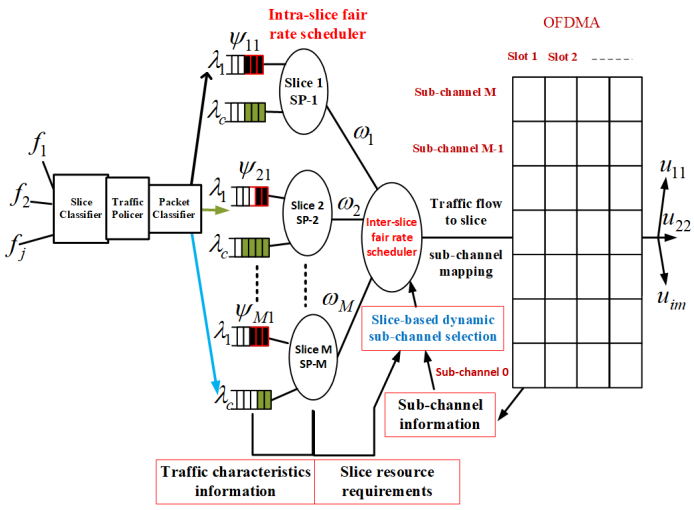

Fig. 4. LiFi Downlink spectrum virtualization.

They are mapped into a corresponding queue in the slice based on their service class, as shown in Fig 4. The MAC measurement controller calculates, per time slot, the total number of admitted traffic packets in AP_i onto slice m, $Q_{t_{m_i}}$, $\forall m \in M$, based on Eq.(1), as follows:

$$Q_{t_{m_i}} = \sum_{k=1}^{K} B_{tk_m}. \tag{2}$$

The local DAS_2 algorithm guarantees each slice defined in an AP_i a minimum number of RBs, R_{min}, per time slot. A slice m, which requires a number of RBs more than R_{min}, is considered overloaded and vice-versa. Based on Eq.(2), the MAC measurement controller calculates, per time slot, the traffic volume vector of slices defined in an AP_i, $\forall i \in N$, as follows:

$$Q_i(t) = [Q_{i1}(t), Q_{i2}(t), ..., Q_{iM}(t)], \forall i \in N, m \in M. \tag{3}$$

Based on Eq.(3), the downlink RBs of each AP are distributed among the defined slices based on proportional weights. These are calculated as follows:

$$\omega_{im} = \frac{Q_{im}(t)}{\sum_{m=1}^{M-U} Q_{im}(t)}; \tag{4}$$

where ω_{im} denotes the weight of slice m defined in AP_i; and U denotes the number of underloaded slices. Based on Eq.(4), an overloaded slice m in AP_i can be allocated a maximum RBs, $R_{max_{im}}$, as follows:

$$R_{max_{im}} = R_{min_{im}} + \lfloor (N_i^{eff} - \sum_{j=1}^{U} R_{im}) \omega_{im} \rfloor; \tag{5}$$

where N_i^{eff} denotes the number of effective RBs in AP_i; and R_{im} denotes the RBs required for the underloaded slice m in AP_i. Based on Eq.(5), the number of RBs can be determined for all the supported slices in each AP_i.

Each slice is assigned a service priority weight based on its traffic load and type. At each sub-frame (time slot), the top level scheduler serves the different slices based on their priority weight in ascending order. The unused RBs by the unloaded slices are scheduled among the overloaded slices based on Eq.(5). This process continues until the RBs are all allocated; or the slices are all served. At the lower level, an enhanced start-time fair queueing (SFQ) scheduler [29] serves packets waiting in the different queues for transmission in the different slices. It schedules the RBs among the different users according to the number of RBs allocated to their corresponding slices. Each queue is assigned a service data rate weight based on its service priority class, c, ψ_{ic}, such that, $\sum_{c \in C} \psi_{ic} \leq 1$. A global virtual time, v_i, tracks the packet service of all queues. Each queue has local start and finish times, which are denoted by S_{ic} and F_{ic}, respectively. When a new packet arrives at an empty queue, c , the start and finish times of the head-of-line (HOL) packet are updated as follows:

$$S_{ic} = max\{F_{ic}, v_i\}; \tag{6}$$

$$F_{ic} = S_{ic} + \frac{L_{ic}}{\psi_{ic}}; \tag{7}$$

where L_{ic} denotes the packet length of class c in AP_i. The packets which arrive at non-empty buffers wait for their service. The queue with the minimum HOL start time is selected for transmission. If the HOL packet can be transmitted in the assigned time slot, without fragmentation, it is dequeued and the global virtual time is updated to the start time, i.e., S_{ij} [30]. Afterwards, the related local start and finish times are updated. If the queue is nonempty, these updates are identical to the arrival case. Otherwise, the start time is set to a large value to remove it from contention, i.e., $S_{ic} = \infty$ [29], [30].

A. Slice service measurement module

This module estimates the instant bandwidth required to transmit the traffic flow in queue j. A guaranteed bandwidth (GB) flow, j, which is admitted in slice m, has an eligible bandwidth request, r_i, given as:

$$r_{ej} = max\{\frac{\gamma_{jt}}{8}[t - S_j(t)], 0\}; \tag{8}$$

where γ_{jt} denotes the target traffic rate (bps) of flow j; and t is the clock time of LiFi AP. The service measurement module maintains a service timer for each GB flow running in the slice i; and $S_j(t)$ is the service timer value of flow j at time t, which applies the concept of the Virtual Clock [31]. This service timer is synchronized with the clock of a LiFi AP, when a flow is admitted. The timer ticks with the following value upon the service of each packet in its corresponding flow:

$$A_j = \frac{8 \cdot L_j}{\gamma_{jm}}; \tag{9}$$

where A_j is the service timer increment of flow j, upon the service of packet with size of L_j bytes ; and γ_{jm} is the measurement rate (in bps) for flow j. For the service timer, this value should be the target rate, γ_{jt}. A separate variable records the virtual time of flow j instead of stamping each arrived packet with its virtual time, as in [31]. Some packets may be dropped due to their violation of maximum latency.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

The service timer thresholds the service rate of flow j within its target traffic rate.

B. Slice QoS enforcement module

This module maintains a QoS timer for each admitted GB flow. The QoS timer is initialized with the clock of LiFi AP, but increases with the value decided by the different measurement rate defined in Eq.(10), given as follows:

$$A_j = \frac{8 \cdot L_j}{r_{min_j}}; \tag{10}$$

where r_{min_j} denotes the minimum reserved traffic rate of flow j. The QoS timer enforces the service rate of flow j to meet a guaranteed value. The QoS enforcement module divides the bandwidth of flow into guaranteed bandwidth(GB) and non-guaranteed bandwidth (NGB) based on the value of the QoS timer, S_{tj}, given as follows:

$$r_j^{GB} = min\{r_j, \frac{r_{min_j}}{8}(t - S_{tj})\}, \quad S_{ti} < t;$$
$$r_j^{GB} = 0, \quad S_{tj} \geq t; \tag{11}$$
$$r_j^{NGB} = r_j - r_j^{GB}. \tag{12}$$

The bandwidth request of BE flow is considered always as NGB. The bandwidth of GB flow can be divided into two parts based on the delay of its packets. The bandwidth of some packets is guaranteed in the frame n, and the bandwidth of the rest of flow packets is guaranteed in the frame $n + 2$ based on the total delay requirement of flow j.

V. Uplink MAC Access Virtualization

The uplink hybrid TDMA-CSMA/CA MAC protocol manages MAC access based on user and generated traffic priority. During the TDMA phase, the frame slots are distributed among the different uplink slices based on their priority traffic. The remaining time slots in the frame are allocated to the different low priority users according to the contention process defined in the CSMA/CA protocol. A synchronization process is required to enable the different UDs to transmit in their time slots and avoid any collision during the TDMA phase. A complete design for an indoor ultra-dense VLC network is introduced in [32], which includes a full synchronization process. This is structured at system and cell levels to mitigate inter-cell interference and avoid medium access collisions in each attocell, respectively [32], [33]. The process synchronizes the spectrum allocation with the different LiFi APs and the users access to the transmission medium in each LiFi cell using TDMA.

The head of the uplink superframe contains information that enables the UDs to synchronize with the clock of their LiFi APs, as shown in Fig. 3a. A LiFi attocell is assigned a unique identifier (ID) based on its AP's position coordinates. A known bit-pattern representing a synchronization sequence is introduced at the beginning of each TDM frame (i.e. time cycle) to enable UDs to access the medium. The pattern of the synchronization sequence may have different lengths or has a specific pattern and same length. The uplink frame of the LiFi AP consists of three parts: synchronization sequence, training sequence and attocell ID, as shown in Fig. 3b. The synchronization is realized when the sequence pattern is detected by UDs. The uplink frame of UDs consists of UD ID, training sequence, attocell ID data, as shown in Fig. 3c. Upon receiving the synchronization sequence, the UD sets its clock to transmit in its designated time slot. The training sequence enables the UDs to detect the SNR and the power of received and transmitted signals. This information is used to evaluate the quality of channel conditions. During the CSMA/CA phase, the timing synchronization function (TSF) specified by the IEEE 802.11 standard assures the synchronization among the contending low priority UDs [34]. The TSF function is used for synchronization by periodically exchanging information through beacons containing timestamps [35].

The UDs which have low-data rate occupy more airtime, compared to the UDs with a higher data rate and the same packet size [36]. The airtime, $a_{k_j^i}$, is the fraction of channel time access allocated to flow j generated by UD k, which is subscribed with slice's SP operator i in one-second interval, given as follows [12]:

$$a_{k_j^i} = \frac{\gamma_{k_j^i}}{r_k^i}; \tag{13}$$

where $\gamma_{k_j^i}$ denotes the guaranteed data rate of flow j generated by UD k; and r_k^i denotes the physical transmission data rate of UD k subscribed with SP i [12]. A user k generates a flow j with a number of packets, P_{k_j}, is allocated an airtime with a guaranteed data rate, γ_{k_j}, as follows:

$$P_{k_j} = \lceil \frac{d_j \gamma_{k_j}}{L_{k_j}} \rceil; \tag{14}$$

$$a_{k_j} = \frac{P_i L_k}{r_k} + OH; \tag{15}$$

where d_j and L_{k_j} denote the delay budget and the maximum packet size of flow j, respectively; and OH denotes the overhead time required to transmit the extra control, acknowledgment and synchronization bits along with the payload of flow j. Based on Eq.(2), the required airtime to slice i can be computed as follows:

$$a_i = \frac{\overline{Q_{t_i}}}{\sum_{i=1}^{M} \overline{Q_{t_i}}}; \tag{16}$$

where the required airtime to UD k can be calculated as follows:

$$a_{k_j} = \frac{\overline{q_{tk_j}}}{\overline{Q_{t_i}}}. \tag{17}$$

The UD k transmits in the CSMA/CA phase with a probability inversely proportional to its minimum contention window (CW_{min}). The throughout of UD k can be measured based on the statistical reports collected every sampling window of successful packets transmission. The measured throughput of UD, k, γ_{k_m}, is inversely proportional to its assigned CW_{min}. A policy is developed to adjust the CW_{min} of UDs subscribed with the different slices to guarantee their proportional throughput subject to their slices' maximal throughput. In essence, the MAC controller algorithm monitors the measured throughput,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

γ_{k_m}, of UD k and dynamically adjusts its CW_{min} to make it converge with the target throughput, γ_{k_t}, as follows [37], [38]:

$$CW_{min}^{k_{ne}} = CW_{min}^{k_{pr}} \cdot \frac{\gamma_{k_m}}{\gamma_{k_t}}; \tag{18}$$

where $CW_{min}^{k_{ne}}$ and $CW_{min}^{k_{pr}}$ denote the next and previous CW_{min} of UD k, respectively. Based on Eq. (13), this policy can also guarantee the airtime fairness among the UDs which have different data rates and compete with each other in the time slots during the CSMA/CA phase, as follows.

$$CW_{min}^{k_{ne}} = CW_{min}^{k_{pr}} \cdot \frac{\gamma_{k_m}}{r_k a_{kj}}. \tag{19}$$

If the measured throughput exceeds the target throughput, the UD k increases its CW_{min}^{k} to decrease its airtime share. This decreases, in turn, the throughput of UD k. The MAC measurement controller measures the average throughput over a sample number of packets to set the current CW_{min}^{k} of UD k. When there is no absolute priority UD (traffic), the UDs adjust their initial CW_{min} to get their fair airtime share. This would support UDs with high data rates, while enabling lower data rate UDs to have their fair airtime share access. This also maximizes the aggregate throughput of the LiFi AP .

VI. Uplink Spectrum Slicing

The proposed DAS_2 scheme supports LiFi uplink spectrum slicing. It dynamically allocates uplink resources to uplink slices while enabling intra/inter slices (SPs) resource borrowing. Uplink traffic packets are policed by a leaky bucket (LB) function running in the UDs [39]. The LB regulates generated traffic packets from applications, characterises and signals its QoS requirements to the LiFi AP.

The proposed DAS_2 scheme extends the Token Bucket Fair Queuing (TBFQ) scheduling algorithm [40], [41] to dynamically support intra/inter scheduling uplink traffic packets among LiFi uplink spectrum slices. It also tracks the usage of traffic flows admitted onto the LiFi uplink slices. A LB per flow per slice is considered instead of deploying a LB per flow per AP [40], as shown in Fig. 5. A local bank for each slice is considered, instead of considering a global tank [40] to serve all the admitted traffic flows into the AP. The local banks interact with a global bank through an autonomic process based on a more elaborate resource borrowing mechanism than discussed in [40]. Our extended token bucket fair queuing (e-TBFQ) scheme comprises a slice based flow classifier and an intra/inter slice scheduling function, as shown in Fig. 5. It also comprises a resource trading module, which enables SPs to trade unused resources among themselves per frame.

A slice may support multiple applications which generate different flows with different traffic characteristics. The slice-based flow classifier function classifies admitted traffic packets based on their ID and then forwards them through a LB corresponding to their application running onto the slice. The LB polices the negotiated parameters of flows generated by applications supported onto slices, which can be characterized in a generic manner as follows:

$$LB_{mj}^{y}, = (r_{mj}, b_{mj}, q_{mj}), \quad \forall m = 1, ..., M; \tag{20}$$
$$y : a_{1j}, a_{2j}, ..., a_{Mj};$$

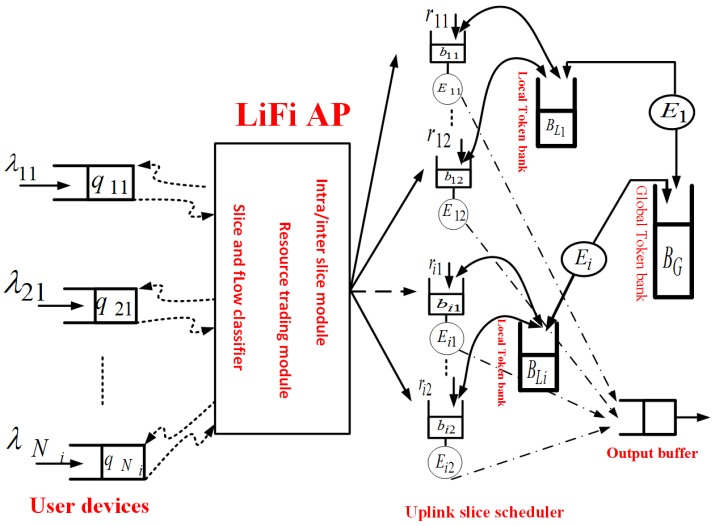

Fig. 5. LiFi Uplink slice scheduler.

where LB_{mj} polices a flow generated by application j running onto slice m; a_{mj} denotes an application j running on a slice m; r_{mj} denotes the token generation rate (bytes per second) of LB_{mj}; b_{mj} denotes the size of the token pool(bytes) of LB_{mj}; q_{mj} denotes the queue length designated for application j running onto slice m. The LB counter, E_{mj}, calculates the number of tokens borrowed from or given to the local token bank for active application flow j on slice m. The counter, E_m, of local bank m counts the number of tokens borrowed from or given to the global token bank. The token generation rates and counters enable the e-TBFQ scheduler to ensure that the transmitted traffic conforms to its negotiated service parameters of its slice during the flow establishment phase with the LiFi AP.

When the admitted flow rate j onto slice m is less than its LB token generation rate, the remaining tokens in the pool are given to the local token bank, where E_{mj} is increased by the same amount of given tokens. Whereas, when the tokens are withdrawn from the local bank for serving the remaining packets of flow j running on slice m, E_{mj} is decreased by the same amount of withdrawn tokens. The remaining tokens in the local bank of slices can be traded per frame with the other slices (SPs) through the global bank. The slice (SP) can receive compensation or charge, $\mho$, per frame based on the given or borrowed tokens, respectively, as follows:

$$\mho = r_o F_o; \tag{21}$$

where r_o is the interest rate; and F_o is the number of given or borrowed tokens.

The flows which are admitted to the same slice borrow tokens from the local bank based on a priority index, ω_{mj}, calculated as follows: $\omega_{mj} = \frac{E_{mj}}{r_{mj}}$ [40]. The flow with the highest priority index in the frame, f, has the highest priority in borrowing tokens from the local bank in frame $f+1$. This flow will be serviced first in the frame $f+1$. Based on the flow traffic load and its respective slice, the length of the TDMA slots are dynamically determined per frame for high priority

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

users. The slices are isolated over each superframe to minimize the performance impact of resource allocation and utilization in one slice over the others [26]. A local bank can borrow from the global bank based on a priority index, ω_i, which can be calculated as: $\omega_m = \frac{E_m}{B_{L_m}}$.

The local bank can borrow tokens from the global bank up to a certain credit, c_m, which defines the maximum number of borrowed tokens. Similarly, a flow j can borrow tokens from the local bank up to a certain credit, c_{mj}. The credit values ensure fairness among applications in borrowing tokens from the global and local banks. When the local bank and a flow have counters, E_m and $E_{mj} = 0$, this means that the token generation rate equals the flow data rate during the frame. The local bank and the flow may borrow tokens from the global and local banks up to a defined debit limit. Both the flow and the local bank should give a number of tokens so that they can borrow tokens. The debit limit for a local bank d_m and, d_{mj}, for a flow j on slice m are set to negative values, so malicious local bank or flow in the same slice cannot affect the QoS of other well behaved flows in the slice [40]. The e-TBFQ scheme exploits the statistical multiplexing of admitted flows in slices to improve the overall bandwidth utilization of the LiFi uplink spectrum [26].

VII. SDN CONTROLLER FUNCTION

The main SDN controller function consists of monitoring some critical parameters relating to supported slices and their users traffic management in each LiFi AP. The monitoring process of virtualized LiFi APs is vital in instantiating and configuring downlink and uplink slices while controlling and managing resource allocation in virtualized LiFi attocell APs. A centralized unit (CU) handles feedback information collected from UDs as well as the network state to enable the different VLC cells to access the spectrum. This aims to support reliable multiuser access in each cell [32]. In this reference, the developed network architecture follows the SDN concept with a primary aim to mitigate inter-cell interference and improve the quality of data transmission on the VLC wireless medium [32]. The DAS_2 scheme exposes some downlink and uplink parameter information to the centralized SDN controller. The parameters per LiFi AP may include: i) slices utilization level, ii) number of UDs and their traffic load per slice, iii) channel quality indicator (CQI) or (SNR) of each UD per slice, and iv) latency of the oldest packet in the UD's queue and its throughput.

The DAS_2 scheme uses the analysis of monitored information to admit flows into supported slices per LiFi AP. It also uses the information to schedule UDs over the virtual RBs assigned to the different slices. The SDN parameters information which characterizes uplink slices in a LiFi attocell AP including the e-TBFQ parameters. This includes the debit, credit, local and global banks counters, as well as token generate rates defined per slice in each LiFi AP. The SDN controller can dynamically configure uplink and downlink parameters to ensure the QoS requirements of applications supported on each slice defined in LiFi APs.

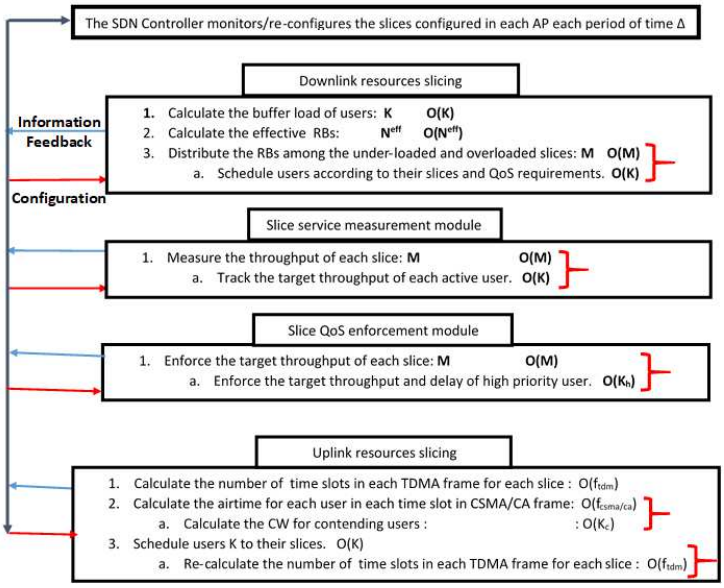

Fig. 6. Resource slicing scheme pseudocode: Complexity analysis.

A. SDN Controller Operation

The centralized SDN controller maintains the SDN parameters used for slicing by the DAS_2 scheme in databases. It can update the SDN parameters and introduce (define) new ones based on new applications requirements. These parameters can be used to program resources allocation to the different slices created or instantiated in LiFi attocell APs. The LiFi AP resources are dynamically shared and allocated to the different slices. The SPs provide their services and application information requirements to the SDN controller, which are received through the northbound interface. The SDN controller configures the parameters of slices or instantiates new ones in the relevant APs, which can better serve them. It communicates this information through the southbound interface to the local AP's controller. This updates resource allocation to the different slices in LiFi APs, which can meet the QoS requirements of users. It controls and manages the resources of APs and configures them to cope with network dynamics and meet QoS requirements. The global admission control mechanism runs in the SDN controller to manage the number of slices configured in each AP. The slices (SPs) can implement their flow-level admission control to ensure that flows with critical resource requirements do not suffer, and perform graceful degradation in the event of resource overload.

B. Airtime Abstraction Function

The QoS of applications which run onto slices defined/instantiated in a LiFi attocell AP can be degraded by a number of elements. These include the number of available RBs, channel quality conditions and type of service in terms of delay or throughput. An admission controller mechanism should be deployed in each LiFi AP to restrict the admission of traffic flows based on the QoS requirements of applications running onto slices supported in each LiFi AP. The admitted users are granted a maximal balance of fairness and utilization

of services. A new flow with an airtime share, a_{ik}, can be admitted to transmit on the LiFi uplink channel subject to the following constraint [12]:

$$a_{k'j} + \sum_{k \in K} \sum_{j \in J} a_{k_j} \leq EA; \tag{22}$$

where EA denotes the effective airtime of the LiFi uplink channel. It is defined as the maximum percentage of available airtime that can be allocated to UDs [12]; $a_{k'_j}$ denotes the airtime share allocated to flow j generated by UD $k^{'}$.

An airtime sharing application can be developed to run on the centralized SDN controller, in coordination with the local LiFi AP controller, to control the UL traffic of UDs. The local control and reporting module in the local AP controller runs the following functionalities: i) reporting the parameters of UDs (e.g. average packet size and physical data rate) which can be calculated based on the LiFi driver traffic statistics. These are found in the rate table maintained in the LiFi driver installed in the UD, ii) sending the collected parameters of UDs in step 1 to the SDN controller, which calculates the maximum airtime limits for the active UDs and sends them to the local AP controller and iii) converting the maximum airtime limit of each UD to a transmission rate value based on the available resources of LiFi APs. This forces the shaping module in each UD to reset the transmitted rate limit to the value enforced by the SDN controller. The shaping module can be implemented by using the Click [42] modular router, which transparently controls the outbound traffic from the UD interface.

VIII. Complexity Analysis

The complexity analysis provides an approximate order number for the operations or steps to create and configure slices in each LiFi AP in an attocell network, as well as to guarantee the target QoS requirements for both slices and their UDs. To simplify the complexity computation of the DAS_2 scheme, Fig. 6 shows the analysis for a sliced LiFi attocell AP. The operations underlining the DAS_2 scheme do not involve matrix-vector multiplication and matrix inversion. Therefore, the realization complexity of the DAS_2 scheme is considered very low, as explained in Fig. 6. The complexity of slicing downlink resources in a LiFi attocell AP requires a number of operations of order $\mathcal{O}(MK)$, where M and K denote the number of slices and UDs, respectively. The same number of operations is required to measure and enforce the QoS requirements of slices and UDs in a LiFi attocell AP. It has approximately the complexity of order $\mathcal{O}(MK)$. However, the slicing process of uplink MAC access and spectrum has the complexity of order $\mathcal{O}(KK_c f_{csma/ca})$, where K_c denotes the number of contending UDs in the CSMA/CA frame part, $f_{CSMA/CA}$. The SDN controller runs each period of time, $\triangle$, to reconfigure the resource allocation to slices and their respective UDs to guarantee their target downlink and uplink throughput, as shown in Fig. 6. Thus the complexity of the proposed DAS_2 scheme for a sliced LiFi attocell network comprising N sliced LiFi APs is approximately of order $\mathcal{O}(NMK\triangle)$. A number of scalar operations of order $\mathcal{O}(1)$ is also involved in the process.

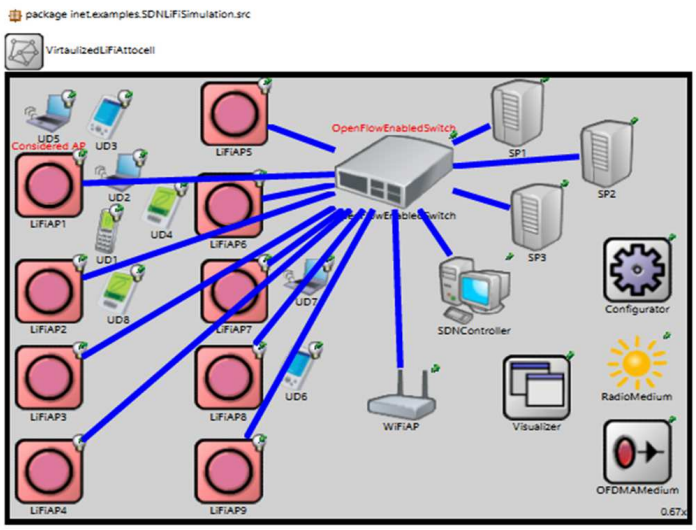

Fig. 7. Virtualized LiFi attocell access network simulation environment in OMNeT++.

IX. Simulation and Results

A simulation environment has been developed based on ns-3 [43] and [44] using OMNeT++ [45] to study the performance of the proposed DAS_2 scheme for slicing downlink and uplink resources in LiFi attocell networks, as shown in Fig. 7. In the simulation setup, 9 LiFi APs are distributed in the indoor Optima lab room of size 14 $\times$ 6 $\times$ 4 (W $\times$ L $\times$ H), at the University-of-Edinburgh (UoE). The downlink and uplink physical data rates of the LiFi AP are taken to be 25Mbps and 19Mbps, respectively. The channel gains are generated based on the Lambertian model described in [46]. The model considers all signal reflections from the different walls. The main parameters of the LiFi AP are summarized in Table 1 [2], [39] and in Section III [8] with a half-intensity radiation angle $\phi_{1/2} = 60^o$ and total number of OFDMA subcarriers, $N_{sc} = 1024$. Three SPs are considered to share the resources in the LiFi attocell access network shown in Fig. 7. Each SP has an application running on a slice in each of the different LiFi APs to provide services to their subscribed UDs. The service applications generate downlink and uplink variable-bit rate (VBR), constant-bit rate (CBR), and best effort (BE) traffic flows with average data rates of 10Mbps, 2Mbps and 5Mbps, respectively. The UDs and IoT devices (IoTDs) are distributed across the different attocells, with a higher density around the LiFi AP1 shown in Fig. 7. The packet size of VBR, CBR and BE traffic is taken to be 500, 80, 1500 bytes, respectively. A number of performance evaluation scenarios have been conducted to demonstrate the capabilities of the DAS_2 scheme to support resource slicing/virtualization in LiFi attocell networks. In this research study, the conducted simulation analysis focuses on evaluating the performance of the DAS_2 scheme in the LiFi AP1 shown in Fig. 7.

1) Dynamic fair slice spectrum allocation: This first scenario assesses the capability of dynamic fair RBs (subcarriers) allocation to active slices based on their measured traffic load. An active slice has traffic to transmit or process for UDs or

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

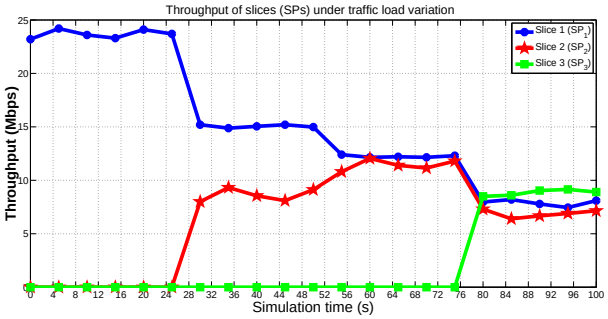

Fig. 8. Throughput of slices under dynamic traffic load variations.

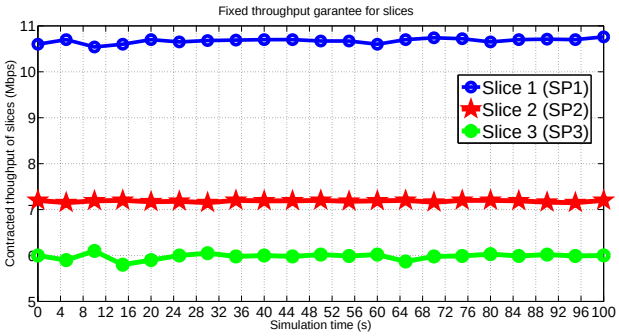

Fig. 9. Contracted throughput of slices(S_1 (45%), S_2 (30%), S_3 (25%)).

their owners SPs. The slices start to transmit different and equal amounts of load at different times during the simulation. As a result, Fig. 8 shows that the slice (S_1) is allocated almost the full downlink spectrum, when there is no traffic generated in the slices 2,3 ($S_{2,3}$) from $t = 0$ to $t = 30$s. When, however, S_2 starts to generate traffic at $t = 30$, the full spectrum is gradually allocated in a proportional and fair manner among active $S_{1,2}$ until $t = 55$. This trend of gradual dynamic fair spectrum allocation among slices continues, as generated traffic in S_3 increases at $t = 75$. At this time, the different slices ($S_{1,2,3}$) generate an almost equal traffic load and therefore they equally share the full spectrum. Thus, the downlink virtualization scheme could adjust the transmission rate of active slices based on their generated traffic load to guarantee fair throughput among them. This demonstrates that the proposed scheme ensures statistical resource multiplexing. It distributes the unused resources of slices with light load among the slices with high load. This increases, in turn, the total throughput of the LiFi AP, while ensuring the performance fairness among all the supported active slices.

2) Fixed slice spectrum guarantee: This second scenario assesses the capability of the DAS_2 scheme to guarantee the target downlink throughput level of $S_{1,2,3}$ agreed in the SLA with the IIP. S_1, S_2,S_3 are guaranteed 45%,30%,25%, of the effective downlink spectrum, respectively. The UDs move a half meter per second among the different attocells toward the marked LiFi AP1 in the network shown in Fig. 7. As a result, Fig. 9 shows that the required resources to achieve the contracted target throughput levels are guaranteed for each active slice. This enables SPs to provision guaranteed services for UDs with a guaranteed data rate level. Despite the UDs' mobility among the different attocells, the DAS_2 scheme maintains the throughput of the different slices. This demonstrates that the slices are completely isolated and traffic class and volume in one slice do not influence the throughput performance of other slices.

3) Minimum slice spectrum guarantee: This third scenario assesses the capability of allocating a minimum throughput to each slice based on a measured traffic class of the supported application, irrespective of the heavily generated best effort (BE) background traffic in a LiFi attocell. The minimum throughout (data rate) of slices $S_{1,2,3}$ is taken to be 6 Mbps. The slices, $S_{1,2,3}$, support guaranteed bandwidth application $(G - VBR)$, non-guaranteed bandwidth application ($NG - VBR$) and constant bit-rate application (CBR), respectively. The background BE traffic is generated across the slices using Pareto distribution with the hurst parameter gradually increasing in this interval $H = [0.54, 0.89]$. As a result, Fig. 10 shows that the DAS_2 scheme guarantees a minimum throughput for $S_{1,3}$, which is required to support different traffic classes independent of BE traffic volume. When the BE traffic exceeds a specific volume level, the $NG - VBR$ traffic rate generated on S_2 falls below the minimum guaranteed throughput. This result demonstrates the controllability of DAS_2 to maintain the resource ratio to the target throughput ratio of the different slices and applications providing guaranteed services to end UDs.

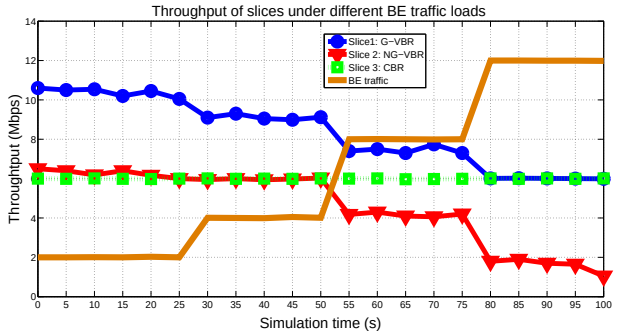

Fig. 10. Throughput of slices under different BE traffic loads.

4) Slice average delay guarantee: This scenario assesses the capability of the DAS_2 scheme to support delay constrained applications on virtualized LiFi AP slices. The slices $S_{1,2,3}$ are configured to support VoIP application, guaranteed bandwidth (G-VBR) video traffic and BE traffic, with a maximum delay constraint of 20 ms and 40ms for $S_{1,2}$, respectively, as well as an unlimited delay upper bound for S_3, respectively. As a result, Fig. 11 shows that the average delay of constrained traffic packets in each slice is guaranteed within the acceptable delay bounds of their application priority class. Generally, the achieved throughput and delay are based on the channel conditions of the LiFi UDs subscribed with each SP. However, the proposed DAS_2 scheme guarantees the delay for each slice's SP based on the generated traffic class of supported applications. The obtained results show an elastic and rigorous control of LiFi downlink resources allocation for instantiated slices in the LiFi AP1 based on their measured

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

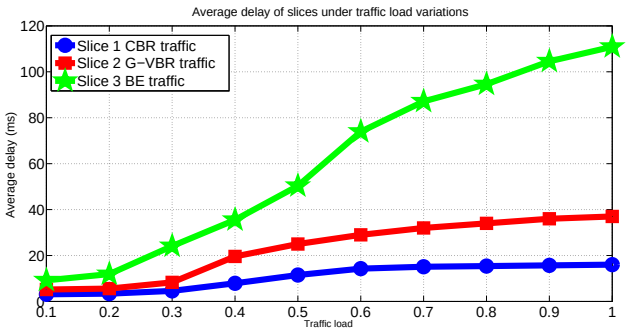

Fig. 11. Average delay of slices under different traffic loads.

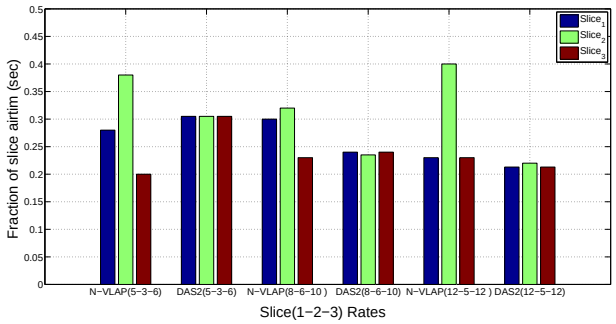

Fig. 12. Fractional of slice airtime under different physical transmission rates.

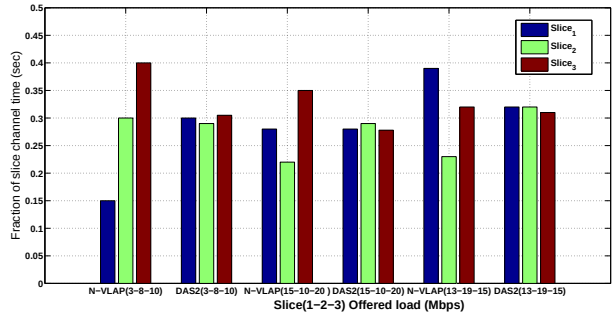

Fig. 13. Fractional of slice airtime under varying offered traffic load.

traffic and QoS enforcement modules. This makes it possible to share resources among the supported slices based on QoS constraints to achieve specific business targets. This enables, in turn, the IIPs to effectively allocate LiFi infrastructure resources to the different SPs to increase their revenue and diversify their services provisioning.

Now, we discuss the results of the conducted scenarios to study the performance of the DAS_2 scheme in virtualizing LiFi AP uplink resource. Various numbers of active UDs are subscribed with the different slices defined in each LiFi AP. However, our analysis is focused on the performance of the DAS_2 scheme in the LiFi AP1 shown in Fig. 7. The UDs may generate different traffic classes. Jain Fair's Index (JFI) [47], [25] is used to evaluate the UL airtime fairness across uplink flows and admitted flows within slices. JFI measures the variations of UL channel time utilization in and across slices, which is given as follows [25]:

$$JFI = \frac{(\sum_{m=1}^{M} r_m)^2}{M \times \sum_{m=1}^{M} r_m^2}; \tag{23}$$

where r_i denotes the sum of the channel time fraction used by all UDs to transmit uplink traffic in slice m; and M denotes the number of slices. The performance of our proposed DAS_2 in virtualizing LiFi AP uplink resources is compared to a non-virtualized LiFi AP (N-VLAP) running the same discussed uplink hybrid MAC protocol. The following subsections discuss the different conducted scenarios to evaluate LiFi AP uplink resources slicing.

5) Varying uplink physical transmission rate: This first uplink resources virtualization scenario evaluates the impact of varying the characteristics of traffic generated by active UDs on the airtime ratio of their slices. Fig. 12 compares the fractional channel time of slices which have UDs transmitting at different physical transmission rates. This demonstrates that the DAS_2 scheme can proportionally control airtime allocation among UDs within and across slices. The DAS_2 scheme could guarantee an equal percentage of airtime sharing among slices, comparing to a N-VLAP. This failed to guarantee the airtime utilization among the UDs in the slices and among the different slices. The airtime utilization of UDs in the different slices is inversely related to their physical data transmission rates and minimum congestion window, CW_{min}, assigned to them during the MAC access process. The generated traffic is shaped at the UDs and conformed at the LiFi AP to provide proportional fair channel utilization among the UDs within and across their slices. The DAS_2 scheme allocates different slice weights to proportionally vary the airtime among the active UDs within their slice [25]. This allows, in turn, fair co-existence of transport protocols with different requirements.

6) Varying uplink offered load: This scenario compares the fractional channel time of slices under different uplink traffic loads generated by active UDs in the different slices instantiated in each LiFi AP. The DAS_2 scheme could guarantee an equal share of airtime among the supported slices in the LiFi AP, as shown in Fig. 13. However, there is a disparity in the utilization of the fractional channel airtime in the N-VLAP. The packet size variations have an impact on the overall sharing of airtime at the LiFi AP for the UDs in the different slices. Without the DAS_2 scheme the result shows that the airtime utilization at the AP directly depends on the size of the uplink traffic packets generated by the active UDs in the different slices. The DAS_2 scheme controls uplink traffic by assigning proportional airtime usage among the UDs in the different slices. This depends on packet statistical multiplexing on the air interface of UDs. The DAS_2 scheme enables the control of airtime among the UDs in the different slices.

7) Uplink spectrum fairness sharing: This scenario compares the fairness in uplink airtime sharing among the UDs within and across slices instantiated in a LiFi AP using the DAS_2 scheme for the same UDs receiving services in the N-VLAP. The modified JFI metric in Eq.(23) [47], [25] is used

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2018.2842055, IEEE Transactions on Network and Service Management

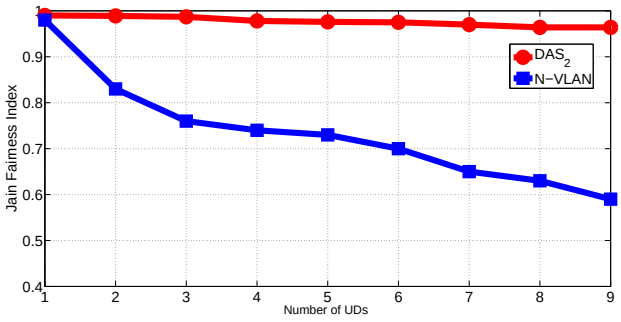

Fig. 14. Airtime fairness comparison of DAS_2 and N-VLAP.

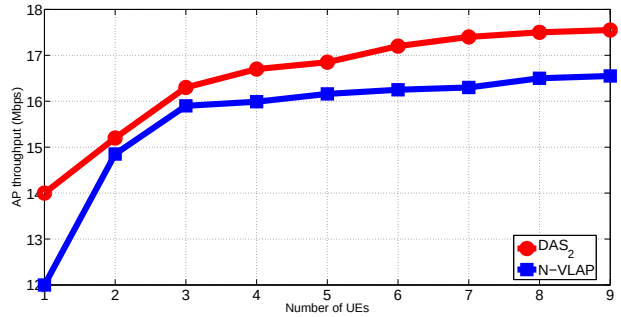

Fig. 15. Throughput comparison of DAS_2 to N-VLAP.

to compute the airtime sharing fairness and variations. The capability of the DAS_2 scheme to ensure proportional fairness enables an explicit control of the spectrum ratios among the UDs given sufficient available total power transmission [48]. As a result, Fig. 14 shows that the actual airtime is almost the same as the target for different values of the minimum congestion window, CW_{min}. This result demonstrates the capability of the DAS_2 scheme to guarantee fairness among the supported slices compared to the N-VLAP. The proposed DAS_2 scheme maintains the intra/inter slice fairness index JFI greater than 0.97, whereas the fairness index falls down to 0.6 in the N-VLAP. The DAS_2 scheme dynamically measures the airtime for every slice and adapts its resources, which result in a more stable fairness performance. It cannot reach the full channel capacity, but it is able to share the remaining airtime fairly.

8) Uplink throughput: This scenario compares the impact of the DAS_2 scheme on the average throughput of the LiFi uplink channel under a different number of UDs to the throughput of N-VLAP. As a result, Fig. 15 shows that the DAS_2 scheme achieves a better throughput performance than the average throughput in N-VLAP. The DAS_2 scheme can better utilize the uplink spectrum of the different slices by using statistical multiplexing. This explains the increase in the measured LiFi uplink throughput.

X. Conclusion

This paper has introduced the DAS_2 scheme to support resource virtualization and slicing in LiFi attocell access networks. It supports uplink and downlink resource slicing of LiFi APs. It also allocates resources to active slices in a dynamic and autonomic manner. The downlink and uplink resources are effectively shared among the different slices instantiated in the LiFi attocell APs. The DAS_2 scheme has extended the hybrid TDMAS/CSMA/CA MAC protocol and the global token bank to support LiFi uplink MAC access and spectrum virtualization. The SDN controller provides the necessary information feedback to make the DAS_2 scheme traffic class aware and thus allocates autonomically the resources to the different active slices. The downlink and uplink traffic packets are managed by the local LiFi AP resource manager. This maintains the target throughput and delay of the different supported traffic classes on the slices supported in each LiFi AP in the LiFi attocell network. The obtained results show a significant improvement in the sharing of the downlink resources and uplink airtime fairness, while maintaining the throughput level of the supported slices in each LiFi AP. The IEEE 802.11 distributed control function (DCF) dynamically adapts the minimum contention window of contending UDs to maintain airtime fairness. The obtained results demonstrate the capability of DAS_2 to support an arbitrary number of slices in a LiFi AP while guaranteeing the QoS requirements of applications running on the different slices. The DAS_2 scheme can be integrated into any LiFi AP to support multiple services, which can offer differentiated quality-of-experience (QoE) for end-users while using LiFi resources in an efficient, effective and isolated manner. The conducted research study represents the first step to demonstrate the feasibility of the developed DAS_2 scheme to support an autonomic wireless resource virtualization architecture for virtualizing LiFi attocell access networks. These characteristics distinguish the DAS_2 scheme from the other proposed slicing and virtualization approaches [5]. The proposed scheme requires further development to support multiple experiments to run on the same LiFi attocell network, while guaranteeing the activities of experiments running on the different slices isolated from each other. The next step in this research study will be developing distributed resource virtualization algorithms that can better optimize resource sharing in LiFi attocell access networks.

Acknowledgment

We gratefully acknowledge support from the Engineering and Physical Sciences Research Council (EPSRC) under grant EP/L020009/1 (TOUCAN Project). Professor Haas acknowledges support from the EPSRC under Established Career Fellowship grant EP/K008757/1.

References

[1] H. Haas, “High-speed wireless networking using visible light,” *SPIE Newsroom*, April 2013.

[2] D. Tsonev, S. Sinanovic, and H. Haas, “Practical MIMO Capacity for Indoor Optical Wireless Communication with White LEDs,” in *Proc. of IEEE VTC-Spring*, June 2013.

[3] A. S. Thyagaturu, A. Mercian, M. P. McGarry, M. Reisslein, and W. Kellerer, “Software Defined Optical Networks (SDONs): A Comprehensive Survey,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2738–2786, Fourth quarter 2016.

[4] A. Blenk, A. Basta, M. Reisslein, and W. Kellerer, “Survey on Network Virtualization Hypervisors for Software Defined Networking,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 655–685, April 2016.

[5] M. Richart, J. Baliosian, J. Serrat, and J. L. Gorricho, "Resource Slicing in Virtual Wireless Networks: A Survey," *IEEE Transactions on Network and service management*, vol. 13, no. 3, pp. 462–476, September 2016.

[6] H. Alshaer, "An overview of network virtualization and cloud network as a service," *International Journal of Network Management*, vol. 25, no. 1, pp. 1–30, January/February 2015.

[7] K. Samdanis, X. C. Perez, and V. Sciancalepore, " From Network Sharing to Multi-Tenancy: The 5G Network Slice Broker," *IEEE Communication Magazine*, vol. 54, pp. 32–39, July 2016.

[8] Z. Chen, N. Serafimovski, and H. Haas, "Angle Diversity for an Indoor Cellular Visible Light Communication System," in *Proc. of IEEE 2014 Vehicular Technology Conference (VTC Spring)*, Seoul, Korea, December 2014, pp. 1–5.

[9] B. Shrestha, E. Hossain, and K. W. Choi, "Distributed and Centralized Hybrid CSMA/CA-TDMA Schemes for Single-hop Wireless Networks," *IEEE Journal in Selected Areas Communications*, vol. 13, no. 7, pp. 4050–4065, July 2014.

[10] Y. Khan, M. Derakhshani, S. Parsaeefard, and T. L. Ngoc, "Self-Organizing TDMA MAC Protocol for Effective Capacity Improvement in IEEE 802.11 WLANs," in *Proc. of IEEE Globecom Workshops (GC Wkshps)*, December 2015, p. 16.

[11] B. Shrestha, K. W. Choi, and E. Hossain, "A dynamic time slot allocation scheme for hybrid CSMA/TDMA MAC protocol," *IEEE Wireless Communications Letter*, vol. 2, no. 5, pp. 535–538, October 2013.

[12] C. T. Chou, N. S. Shankar, and K. G. Shin, "Achieving per-stream QoS with distributed airtime allocation and admission control in IEEE 802.11e," in *Proc of the IEEE Infocom*, April 2005.

[13] J. Robinson and T. Randhawa, "Saturation throughput analysis of IEEE 802.11e enhanced distributed coordination function," *IEEE Journal in Selected Areas Communications*, vol. 22, no. 5, p. 917–928, June 2004.

[14] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, "NVS: A Virtualization Substrate for WiMAX Networks," in *Proc.of MobiCom '10*, Chicago, Illinois, USA, September 2010, pp. 233–244.

[15] T. Guo and R. Arnott, "Active LTE RAN Sharing with Partial Resource Reservation," in *Proc.of IEEE 78th Vehicular Technology Conference (VTC Fall)*, Las Vegas, NV, USA, September 2013, pp. 1–5.

[16] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, "NVS: A substrate for virtualizing wireless resources in cellular networks," *IEEE/ACM Transactions on Networking*, vol. 20, pp. 1333–1346, October 2012.

[17] Y. Zaki, L. Zhao, C. Goerg, and A. T-Giel, "LTE mobile network virtualization," *Mobile Networks and Applications*, vol. 16, pp. 424–432, June 2011.

[18] G. Bhanage, R. Daya, I. Seskar, and D. Raychaudhuri, "VNTS: A virtual network traffic shaper for air time fairness in 802.16e systems," in *Proc. of IEEE ICC*, Cape Town,South Africa, 2010, pp. 1–6.

[19] G. Bhanage, I. Seskar, R. Mahindra, and D. Raychaudhuri, "Virtual base station: Architecture for an open shared WiMAX framework," in *Proc. of Proc. 2nd ACM SIGCOMM Workshop*, New Delhi, India, December 2010, pp. 1–8.

[20] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, "Cellslice: Cellular wireless resource slicing for active RAN sharing," in *Proc. of IEEE COMSNETS*, Bengaluru,India, 2013, pp. 1–10.

[21] R. Mahindra, G. D. Bhanage, and G. H. et al., "Space versus time separation for wireless virtualization on an indoor grid," in *Proc. of Next Generation Internet Networks (NGI)*, Krakow, Poland, 2008, pp. 215–222.

[22] L. Xia, S. Kumar, and X. Yang, "Virtual WiFi: Bring virtualization from wired to wireless," *ACM SIGPLAN Notices*, vol. 46, pp. 181–192, October 2011.

[23] G. Smith, A. Chaturvedi, A. Mishra, and S. Banerjee, "Wireless virtualization on commodity 802.11 hardware," in *Proc. of Int. Workshop Wireless Networks*, Montreal, QC, Canada, 2007, pp. 75–82.

[24] A. Checco and D. J. Leiths, "Fair Virtualization of 802.11 Networks," *IEEE/ACM Transactions on Networking*, vol. 23, no. 1, pp. 148–160, February 2015.

[25] G. Bhanage, D. Vete, I. Seskar, and D. Raychaudhuri, "SplitAP: Leveraging Wireless Network Virtualization for Flexible Sharing of WLANs," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM)*, December 2010.

[26] A. Shoaei, M. Derakhshani, S. Parsaeefard, and T. L-Ngoc, "Learning-based Hybrid TDMA-CSMA MAC Protocol for Virtualized 802.11 WLANs," in *Proc. of IEEE PIMRC*, December 2005, pp. 1861–1866.

[27] L. S. Puthalath, "Programming the Enterprise WLAN An SDN Approach," Master's thesis, Institu Superior Tecnico Universidade Tecnica Lisboa, June 2012.

[28] L. S. Puthalath, J. S.Zander, R. Merz, A. Feldmann, and T. Vazao, "Towards programmable enterprise WLANS with Odin," in *Proc. of HotSDN '12*, Helsinki,Finland, August 2012, pp. 115–120.

[29] P. Goyal, H. M. Vin, and H. Cheng, "Start-time Fair Queuing: A Scheduling Algorithm for Integrated Services Packet Switching Networks," in *Proc. of SIGCOMM*, Stanford,CA,USA, August 1996.

[30] N. Ghani and J. W. Mark, "Hierarchical scheduling for integrated ABR/VBR services in ATM networks," in *Proc. of IEEE GLOBECOM*, Phoenix,AZ,USA, October 1997.

[31] L. Zhang, ""virtual clock": A new traffic control algorithm for packet-switched networks," *Proc. of ACM Transactions on Computer Systems*, vol. 9, pp. 101–124, 2004.

[32] M. Zeng, K. Zhou, C. Gong, S. Lou, X. Jin, and Z. Xu, "Design and demonstration of an indoor visible light communication network with dynamic user access and resource allocation," in *9th International Conference on Wireless Communications and Signal Processing (WCSP)*, Nanjing, China, December 2017, pp. 2472–7628.

[33] H. Lonn, "A Fault Tolerant Clock Synchronization Algorithm for Systems with Low-Precision Oscillators," in *EDCC-3 Proceedings of the Third European Dependable Computing Conference on Dependable Computing*, September 1999, pp. 88–105.

[34] I. Std.802.11, "Wireless LAN Medium and Physical Layer (PHY) specification," 1999.

[35] Y. Nakayama and S. Aikawa, "Cell Discard and TDMA Synchronization Using FEC in Wireless ATM Systems," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 1, p. 330337, January 1997.

[36] M. Heusse, F. Rousseau, G. B. Sabbatel, and A. Duda, "Performance Anomaly of 802.11b," in *Proc. of INFOCOM*, San Francisco,USA, March/April 2003.

[37] Q. Li, T. Li, T. Turletti, and Y. Xiao, "Saturation Throughput Analysis of Error-Prone 802.11 Wireless Networks," *Wiley Wireless Communications and Mobile Computing*, vol. 5, no. 8, pp. 945–956, November 2005.

[38] T. Joshi, A. Mukherjee, Y. Yoo, and D. P. Agrawal, "Airtime Fairness for IEEE 802.11 Multirate Networks," *IEEE Transactions On Mobile Computing*, vol. 7, no. 4, pp. 513–527, April 2008.

[39] H. Alshaer and H. Haas, "SDN-enabled Li-Fi/Wi-Fi wireless medium access technologies integration framework," in *Proc. of IEEE 2016 on Standards for Communications and Networking (CSCN)*, Berlin,Germany, December 2016, pp. 1–6.

[40] W. K. Wong, H. Zhu, and V. C. M. Leung, "Soft QoS Provisioning Using Token Bank Fair Queuing Shceduling Algorithm," *IEEE Wireless Communications*, vol. 10, no. 3, pp. 8–16, June 2003.

[41] W. K. Wong and V. C. M. Leung, "Scheduling for Integrated Services in Next Generation Broadcast Networks," in *Proc. of IEEE WCNC*, New Orleans,USA, September 1999.

[42] E. Kohler, R. Morris, and B. C. et al., "The Click Modular Router," *IEEE Transactions on Computer Systems*, vol. 18, no. 3, pp. 263–297, August 2000.

[43] A. Aldalbahi, M. Rahaim, and A. K. et al., "Extending ns−3 To Simulate Visible Light Communication at Network-Level," in *Proc. of 23rd International Conference on Telecommunications (ICT)*, June 2016, pp. 1–6.

[44] D. Klein and M. Jarschel, "An OpenFlow Extension for the OMNeT++ INET Framework," in *Proc. of OMNeT++*, March 2013, pp. 322–329.

[45] OMNeT++, "https://omnetpp.org/."

[46] J. Barry, J. Kahn, W. Krause, E. Lee, and D. Messerschmitt, "Simultion of Multipath impulse response for Indoor Wireless Optical Channels," *IEEE Journal Selected Area Communications*, vol. 11, no. 3, pp. 367–379, 1993.

[47] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," *Technical report*, September 1984.

[48] Z. Shen, J. G. Andrews, and B. L. Evans, "Adaptive resource allocation in multiuser OFDM systems with proportional rate constraints," *IEEE Transactions on Wireless Communications*, vol. 4, no. 6, pp. 2726–2737, November 2005.

Hamada Alshaer (M'2002, SM'2013) received the PhD degree in computer engineering and telecommunications from Pierre et Marie Curie University (PMCU), France, in 2005. He is the recipient of the 2009 Royal Academy of Engineering travel award, as well as scholarships from UNRWA, French government and Pierre et Marie Curie University for academic distinctions during his BEng, MSc and PhD studies. He is currently involved in software defined wireless networking and mobile networks virtualization research development in the LiFi R&D centre in the school of engineering at the University of Edinburgh, UK. He has worked for more than 13 years with British Telecom (BT) and etisalat, INRIA-France and Leads University in academic and industrial research development. He has authored more than 50 papers in various research topics in electrical and computer engineering including information technology and systems. He authored a book entitled "Demanding Traffic Control and Management in Next Generation Networks: QoS analysis, network simulation, performance modeling"(LAP LAMBERT Academic Publishing, 2009). He is a regular reviewer for research grants submitted for prestigious funding bodies in Europe, Asia and Middle-East. He has served on the technical program committee of various IEEE conferences, including the IEEE Intelligent Vehicles Symposium, Vehicular Technology Conference, Globecom, ICC and WCNC, and chaired some of their sessions. His research interests include cross-layer design for vehicular networks, software-defined wireless networking, wireless networks virtualization, mobile communications security, optical communications and networking, wireless sensor networks and intelligent transportation.

Harald Haas arald Haasarald HaasH (S98-AM00-M03-SM16-F17) received the Ph.D. degree from The University of Edinburgh in 2001. He is currently the Chair of Mobile Communications at The University of Edinburgh, and he is the Initiator, Co-Founder, and Chief Scientific Officer of pureLiFi Ltd., and the Director of the LiFi Research and Development Centre, The University of Edinburgh. He has authored 430 conference and journal papers, including a paper in Science and co-authored a book Principles of LED Light Communications Towards Networked Li-Fi (Cambridge University Press, 2015). His main research interests are in optical wireless communications, hybrid optical wireless and RF communications, spatial modulation, and interference coordination in wireless networks. He first introduced and coined spatial modulation and LiFi. LiFi was listed among the 50 best inventions in TIME Magazine in 2011. He was an invited speaker at TED Global 2011, and his talk on Wireless Data from Every Light Bulb has been watched online over 2.4 million times. He gave a second TED Global lecture in 2015 on the use of solar cells as LiFi data detectors and energy harvesters. This has been viewed online over 2 million times. He was elected as a fellow of the Royal Society of Edinburgh and a Fellow of the IEEE in 2017. In 2012 and 2017, he was a recipient of the prestigious Established Career Fellowship from the Engineering and Physical Sciences Research Council (EPSRC) within Information and Communications Technology in the U.K. In 2014, he was selected by EPSRC as one of ten Recognizing Inspirational Scientists and Engineers Leaders in the U.K. He was a corecipient of the EURASIP Best Paper Award for the Journal on Wireless Communications and Networking in 2015 and the Jack Neubauer Memorial Award of the IEEE Vehicular Technology Society. In 2016, he received the Outstanding Achievement Award from the International Solid State Lighting Alliance. He was a co-recipient of recent best paper awards at VTC-Fall, 2013, VTC-Spring 2015, ICC 2016, ICC 2017 and ICC 2018. He is an Editor of the IEEE TRANSACTIONS ON COMMUNICATIONS and the IEEE JOURNAL OF LIGHTWAVE TECHNOLOGIES.