IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 23, 2019, accepted June 8, 2019, date of publication June 14, 2019, date of current version June 28, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2923056

Mini-Base Station: A Novel Smart Virtual eNB for 5G and Beyond Mobile Networks

FOUAD A. YASEEN AND HAMED S. AL-RAWESHIDY, (Senior Member, IEEE)
Wireless Networks and Communications Centre, Department of Electronic and Computer Engineering, College of Engineering, Design and Physical Sciences,
Brunel University London, London UB83PH, U.K.

Corresponding author: Fouad A. Yaseen (fouad.yaseen@brunel.ac.uk)

ABSTRACT This paper presents a proposal for an innovative base station for 5G and beyond for a mobile network system based on the evolved packet system (EPS). Functions of EPS elements can be achieved by the SVeNB, according to the type of function or service and the device that is used to achieve that function or service. The current system, even if these functions and services have been virtualized by the system, it may put them away from the users. Consequently, this causes a delay in completing the establishment of a connection. Since these functions and services can be virtualized and programmed in virtual machines (VMs), we have been simulated the kernel of those functions and services executed by the SVeNB through its VMs. Moreover, The SVeNB position is located close to the users, which helps to reduce the End-to-End delay. Our proposed approach gives the SVeNB the ability to share and contribute in making decisions with the core network (CN) and BaseBand Unit (BBU) pool to initiate a connection, further in making determinations within its vision regarding its users. This procedure leads to a reduction in the load on the BBU pool and the CN, thus lessening the latency, and to increase the data rate that is delivered to the users. That means the SVeNB can regionally decide for all local users that are covered by the SVeNB to connect to each other or the Internet. The MATLAB platform has been used to evaluate the SVeNB performance, with the results demonstrating that it surpasses. The obtained results point that the SVeNB passes the C-RAN system by 68% in terms of user profiles processing, whilst also and it reducing the End-to-End delay by 62%.

INDEX TERMS 5G, control plane, communication networks, SDN, NFV.

I. INTRODUCTION

5G communication network systems are geared towards minimizing the size, cost, power consumption, and deployment of physical hardware. In addition, it will increase the transmission data rate, speeds up of the delivery of data, increases performance, and virtualizes physical resources of the support of the Software Defined Networking (SDN) and Network Functions Virtualization (NFV). The most significant challenges for 5G mobile communication networks are providing sufficient capacity to accommodate the growth of mobile data traffic and decreasing End-to-End latency. The technologies of movable smart gadgets have been advancing. At the same time, the numbers of their users have been expanding, and intuitively, will lead to exceeding of the capabilities of mobile communication networks. In order to address such obstacles, these networks should be efficient to satisfy expandable needs of mobile networks of effective performance. To realize these goals, mobile networks should be able to handle this expansion without increasing the cost of these networks [1], [2]. Most of the current or proposed mobile communication networks working under 4G conditions have implemented virtualization in both hardware and software infrastructure. The theory of SDN depends on splitting the Control Plane (CP) from the Data Plane (DP) of network devices [3]. This separation is to simplify the control and management of networks. It is also to facilitate progression and development by abstracting the network control functions into a virtually centralized CP that allows for the network management to be programmable by software developers in a smooth and easy way [4]. Making decisions is based on software controllers that are logically centralized away from the network devices that forward the data packets. The forwarding devices can be controlled and programmed through an open interface by using the OpenFlow protocol to become pure data packet forwarding devices [5], [6]. SDN plays the influential role

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

in reducing the power, minimizing the number of physical devices, decreasing the cost of installing a network, reducing control delay time, and in allowing the network administrator to configure the instructions for the CP which are executed by the centralized controller [7]. The integrity of SDN is NFV technology [8], which enables building an End-to-End network underpinned by utilizing virtualization technology standards to empower the consolidation of different communication network devices [9]. Cloud computing paved the way to coping conveniently with mobile communication network infrastructure [10]. The infrastructure of communication systems, whether it be wireless, wired, motionless or mobile networks, the virtualization technology is growing dramatically to design and build these networks. Such virtualization can be seen in mobile cellular air networks, such as Cloud Radio Access Network (C-RAN), which is emerged as one of the results of cloud computing. In C-RAN the BBUs (whether virtualized or not) are gathered in a centralized BBU pool, where the fundamental RF functions are accomplished, such as establishing the initial connection, processing RF signals and managing and maintaining the established connections. The proposed SVeNB has the ability to perform these responsibilities depending on the users' profiles that are received from the BBU and the CN. These user profiles have already processed by the BBU and the CN and sent to SVeNB which covers the local users within its coverage area. This information is stored as a backup,for instance, to be used by SVeNB to empower the resident subscribers who are covered by it, without communicating with the BBU pool and the CN. This process enormously mitigates the load on the CN, the BBU pool, while also lowering decisions making latency, which leads to expediting up the establishment of the connection of the End-to-End device. Our proposal presents a new mini-base station for 5G and beyond for mobile communication networks that use virtualization techniques. The main contributions of this research are follows:

- Proposing a novel mini-base station that utilizes different virtualization technologies to simulate the kernel functions of mobile communication networks;
- Mitigating the burden on the BBU pool and the CN by making connection decisions between users without needing to contact the BBU pool and the CN;
- Decreasing the End-to-End delay by reducing the number of control messages amongst the serving entities of mobile networks.

II. BACKGROUND AND RELATED WORKS

There has been much research on virtualizing wireless mobile Base Stations (BSs) and some of this literature is considered here in chronological order. In 2010, Zhu et al. executed the first TDD-SDR BS on a server and also, the first functional model of a Virtual Base Station (VBS) pool was presented. The outcomes satisfied the demands of the system used at that time [11]. Likewise in 2010, Bhanage et al. introduced a proposal for a virtualization wide-area network for 4G devices in a cellular BS to empower users participating among multiple autonomous slice users [12]. By 2011, the design of a mobile system that could reconfigure itself called Reconfigurable Mobile Network (RMN) has been put forward by Khan et al. They employed virtualization techniques to distribute the resources of networks [1]. In 2012, a design of Network Virtualization Substrate (NVS) for base station uplink and downlink sources split into parts was provided by Kokku et al. They adopted two situations: firstly, they showed the optimal slice list with resources and bandwidth bases. Secondly, they customized the flow scheduling within the base station on a single-slice basis [13]. By 2013, the importance of isolation resource requirements being provided to BSs had been grasped by Kiess et al. They focused on OFDM as a wireless interface, preparing for the virtualizing of an OFDM base station [14]. In 2014, a structure for virtualizing the Long Term Evolution (LTE) BS was introduced to improve the subscribers' performance and decrease the interference of the downlink beacon. They employed the assets of OpenFlow and SDN. The outcomes revealed improvements in the throughput, the average interference, and lost packets transferred between End-to-End users [15]. Nakauchi et al. suggested a virtual base station, with the capability to produce dynamic virtualized wireless communication networks. They utilised a couple of techniques, first, a VBS configured dynamically and controlling several physical wireless Access Points (APs) by applying the OpenFlow features. Second, seamless handover is handled between VBSs pre-authentication and pre-association at the target VBS [16]. Xiaodong et al. utilised FNA within SND for next-generation mobile networks with MOFP have been provided. Furthermore, they introduced an adoption strategy to concentrate on the User-Centric scenario in FNA [17]. Dawson et la. depending upon C-RAN, suggested a VBS network structure that enables connection to the CN [18]. In 2015, Wang et la. proposed virtual base stations, where every cell is dynamically determined by allotting virtualized signal sources to the fronthaul link for both the RRHs and BBUs. The functional process of the BBU is cloudified and virtualized according to the functional entities performance [19]. The considerable advancement of industries of electronic communication devices, especially in relation to fixed, wireless, and mobile devices production making these devices in tiny sizes. The Field-Programmable Gate Array (FPGA) technology is a hopeful advancement for solving the needed elasticity for producing and implementing 5G and beyond mobile communication networks. FPGA assists in the LTE scheme virtualization by carrying out a particular task in the CP. This usage can be observed in multi-mode base station infrastructure devices [20]. Virtualizing a single entity of a CN was introduced by Heinonen et al., who worked on the mobility and latency of 5G network. They brought forward the Mobility Management Entity (MME) to be set in a virtualized network located between the radio access network and the CN [21]. In [22], a logical architecture for 5G systems is proposed, in the form of a scheme for managing mobility among different access networks based on SDN and

NFV. However, to date, the current communication networks, especially wireless and mobile networks have not wholly been virtualized [14], [21].

III. PROPOSED SYSTEM

Mobile communication systems suffer from End-to-End latency due to different types of delays such as RF registration delay, control delay, transmission delay, and processing delay. Minimizing these can be achieved by adopting new procedures that manipulate the establishment and negotiation of a contact session. The servers (BBU and CN entities) are responsible for making decisions of an End-to-End connection based on executing the function programs. These servers are usually placed away from the User Equipments (UEs). To decrease the End-to-End latency, we proposed a novel BS that contributes to making connection decisions through virtualizing the functions of the BBU and the CN servers and administering these functions near the UEs. The next section presents an explanation and illustration of the proposed system.

A. EPS FUNCTIONS SLICING AND VIRTUALIZING

The emergence of new technologies is helping in the design and creation of different types of network topologies. These technologies such as SDN, NFV, network virtualization, network slicing, network cloudification, and other virtualizing techniques, are facilitating the network service providers and operators in eliminating the subordination of traditional physical network infrastructure elements and the dedication of physical servers for each function [8]. The functions of these servers can be performed by adopting an appropriate virtualization technique for emulating in making decisions that are yielded by completing the execution of those functions. These *decisions* can be defined as the outputs of those functions that are implemented by the BBU pool and the CN entities, which follow software or programs that control the functions. Because the EPS system makes decisions (outputs of function) to establish, create and maintain a connection between End-to-End according to those functions. Therefore SVeNBs' VMs host the kernel of those functions to make its decisions regarding the local UEs. Examples of the EPS functions that are hosted by SVeNB are:

- Policy Control and Charging Rules Function (PCRF): detects the service flows, determines how data stream moves, and it is responsible for charging policy;
- Packet Data Network Gateway (P-GW): is responsible for assigning IP addresses to UEs, implementing a QoS application, and following PCRF per flow rules;
- Serving Gateway (S-GW): creates, deletes, and modifies the bearers for each UE that connected to the EPS. Performing of these functions depends on per Packet Data Network (PDN) established connections for each UE;
- Mobility Management Entity (MME): achieves the high-level signaling of the UE, the issues of authentication and security, besides to the management of data streams that unrelated to radio communications.

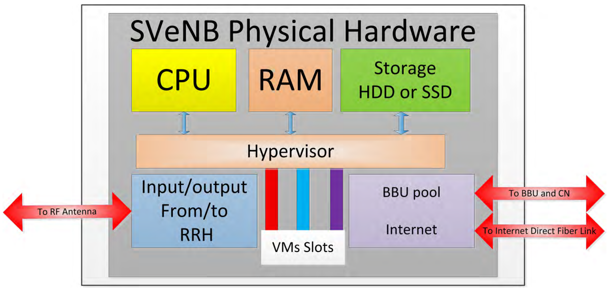

FIGURE 1. The architecture of SVeNB.

Functions slicing is defined as partitioning a function between two servers or VMs to complete a specific task to make decisions, one of which is complementary to the others functionality. One of these servers represents a *MASTER entity, which is located at the CN*, whereas the second is a *SLAVE installed in SVeNB as a VM*. The SLAVE entity receives a function task pre-processed in the MASTER entity, that is, the former acts to complement the function already pre-processed by the latter. While *functions virtualizing* technique supports the possibility to virtualize any task or function that could be achieved by software programs, to be virtualized through a VM to produce session establishment decisions. i.e., functions of the BBU pool and CN servers such as MME, S-GW, P-GW, and PCRF as traditional off-the-shelf servers could be logically virtualized their functions by suitable VM's configuration [23].

B. ARCHITECTURE AND FUNCTIONS OF THE SVeNB

Any virtualization of a network device needs three main physical hardware elements to build and implement a VM: CPU, RAM, and storage (HDD or SSD). Moreover, three main software elements, including an Operating System (OS), application programs, and data need to be considered. The SVeNB has been proposed to consist of one small device has CPU, RAM, storage, besides to peripheral slots allow to insert FPGA boards, which could be added for achieving at least one of the EPS server functions. Each slot represents at least a single VM depends on SVeNB physical hardware and shares its resources. The SVeNB has three main interfaces, the first being connect to the BBU pool and CN, whilst the second is to connected to the Internet gateway and the third interface connected to the RRH. A hypervisor is used to coordinate the VMs with physical resources. Figure 1 shows the proposed architecture of the SVeNB.

The functions of the SVeNB are that it will have the ability to achieve a connection between the local subscribers or to the Internet directly. It can perform this connection without waiting for the decisions that are usually made in the EPC to create the requirements for the session. SVeNB can serve the subscriber who wants to make this session or to connect to the Internet. This is based on the requirements that are pre-processed in the BBU pool and CN for all local subscribers covered by SVeNB. All user profiles are saved in the storage of SVeNB as cached user profiles. According to the user information profiles, SVeNB controls and governs the connection conditions between local subscribers. It takes part in making the decisions that are pre-negotiated with the BBU pool and the CN. It saves these decisions to be applied to make the final ones that enable the subscribers to communicate with each other. Also, it reserves pre-emption of bearer radio resources (ready to use), which are assigned by SVeNB to the users who want to connect with EPC without asking the BBU pool and CN again. The allocated IP addresses by P-GW are stored in SVeNB as ready to use. The SVeNB's VMs can detect and serve the type of required service, such as assigning an IP address. Consequently, based on this service detection the controller can decide to give QoS to the user. Moreover, it selects the interface to flow the classified data. Depending on the rules of PCRF (operates in real time in order to determine policy rules in the CN), the controlled flow-based charging functionalities in the Policy Control Enforcement Function (PCEF) that are virtualized and installed on the VM that works as the PCEF virtual server (supports offline and online charging interactions while PCRF does not support these). This virtual PCEF is responsible for dedicating an appropriated link connection, charging session establishment, and maintaining the established connection. The LTE and EPC functions that determine the decisions of a connection establishment are sent to SVeNB, which stores these decisions to be used for resolving the final decisions of the contact establishment. This procedure will lighten the load on the BBU pool and the CN. Furthermore, this will decrease the End-to-End delay, and expedite the process of initiating the communication connections, in particular, when the local users attempt to link with others or to the Internet.

C. UE – eNB CONNECTION IN C-RAN AND SVeNB

In LTE, the Random Access Channel (RACH) consists of 64 Random Access Preambles (RAP) allocated for each cell of an eNB and there are two ways to get RACH by a UE: (i) Contention-based, where any UE can access this when in need of an uplink connection; and (ii) Contention-free RACH can be used in cases, such as handover and downlink data stream, where low latency is required. The message exchange between the UE and eNB is either for location area updates or for initial access to set up a connection. In the latter case, the UE sends a request via a RAP message on the Physical Random Access Channel (PRACH) resources associated with Random Access Radio Network Temporary Identity (RA-RNTI) to the eNB, which covers the UE. The eNB replies with a Random Access Response (RAR) on the Physical Downlink Shared Channel (PDSCH), which is addressed by an ID and then, RA-RNTI is sent. The UE sends Layer 2 / Layer 3 messages. This information is the first scheduled uplink transmission on the PUSCH and formulates the use of a Hybrid Automatic Repeat Request (HARQ), including carrying of UE identifiers. This is the actual random access procedure message. The last action is the contention resolution message, which the UEs respond to on reception in two possible ways. The first is through a Positive Acknowledgement (ACK), which indicates that the UE has accurately decoded the message and thus, detected its own identity. The second probability is Discontinuous Transmission (DTX), which means that either the UE has correctly decoded the information and discovered that it contains another UE's identity (contention resolution), or it cannot decode the message. Figure 2 illustrates the random access procedure [24], [25].

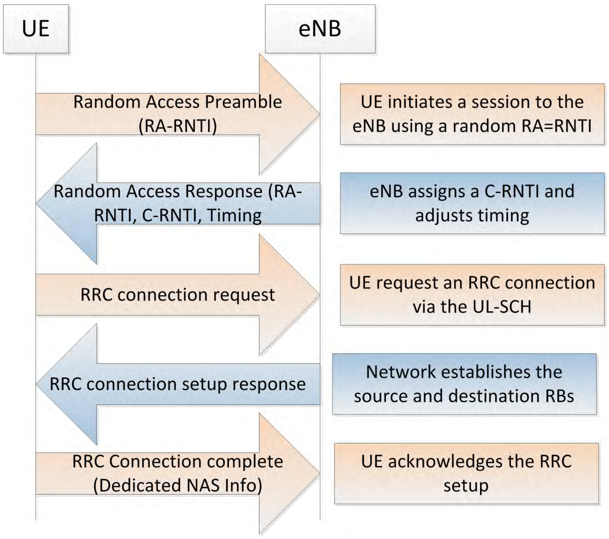

FIGURE 2. Contention-based random access procedure.

At this point, our proposed SVeNB works as legacy eNB, based on the local user profiles that are saved by its storage unit. The installed VMs check the user profiles to see who needs service and then, it takes a decision (to begin the establishment of a session or not), according to having checked the user charge allowance, authentication, and other user profile matters. Subsequently, it assigns an IP address to any user who needs to make a call or contact the Internet. The connection between SVeNB and the Internet can be made in two ways. The first, is via the direct fiber link that is connected directly to the Internet through the operator's S-GW, thereby bypassing the BBU pool and CN. This path is the default way to reaching all IP networks. The second way is through the BBU pool, CN, and Operator's S-GW, which is a backup path for connecting to all IP networks. The default path mitigates the load on the BBU pool and the CN, expedites the data delivery to the users, decreases end to end latency, increases the amount of delivered data by end users, and decreases the utilization of network bandwidth for control messages.

IV. END-TO-END DELAYS ANALYSIS

In this section, discussion on End-to-End delays or session establishment delays is presented. We compared the proposed SVeNB and C-RAN structure connected by the fiber

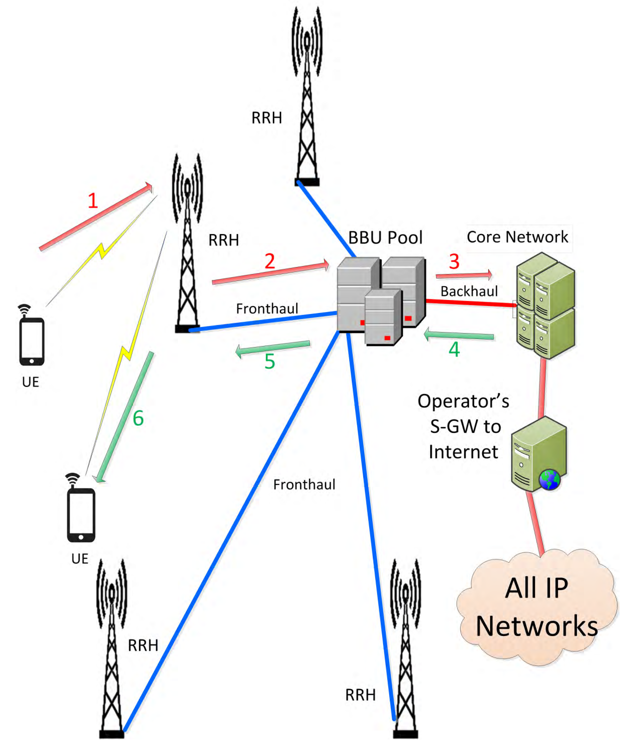

FIGURE 3. **The C-RAN End-to-End CP session establishing.**

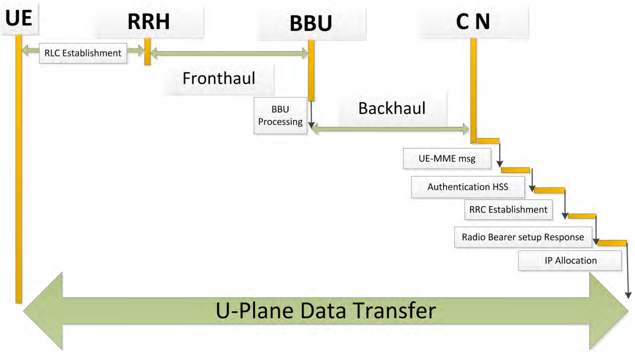

FIGURE 4. **The C-RAN End-to-End CP processing steps.**

in the fronthaul and backhaul links, concentrating on the End-to-End delay analysis. The scenario suggests that two local UEs try to connect to each other within one RRH. We compared our proposal with the C-RAN system to initiate connection steps. Figure 3 shows the representation of the C-RAN steps of session establishment verification. In step 1, the user contacts the RRH to get the radio bearer and IP address, whilst Step 2 the RRH asks the BBU for this request. The BBU processes the measurements of digital signaling and forwards the request to the CN to complete the session establishment. During Step 3, the CN performs the principal proceedings, such as determining the QoS for the bearer, IP distribution, and allowance charging, etc. This requires all the source CP messages crossing through the BBU and CN. Steps 4, 5, and 6 describe preparing to communicate with the target UE.

Figure 4 represents the principal steps of processing to set up an End-to-End association. The transmission delay between the RRH and the BBU (fronthaul link) as well as the BBU pool and the CN (backhaul link) relies upon the separation distance of the connection between them. In the CN, there are several servers or entities executing the jobs of establishing the End-to-End session, with each having its processing and queuing delay times. The connection steps above mentioned become more complicated if the users are under different RRH coverage, for, the BBU and the CN make a check on each user to initiate the connection between them. Consequently, in such a case the End-to-End delay will increase due to need for more processing to achieve the session connection. However, with our proposal, the session connection is manipulated and treated locally in SVeNB, which initiates the connection based on the information received from the BBU pool and CN. Moreover, the SVeNB controls several RRHs and acts as a macro base station. Hence it will cover a geographic area at least with a radius around 10 km.

SVeNB accomplishes the tie without attaching to the BBU pool or the CN again. The idea is a creative step because of the semi-centralized nature of initial connection processing and the capability of SVeNB to make decisions for the UEs as a semi-standalone base station. Figure 5 explains the structure of this scenario. The procedure of this scenario as follows.

- Step 1 depicts the UE's request to obtain a bearer and IP address. SVeNB responds to this request by checking the user profiles stored in its storage unit. Then, the installed VMs of SVeNB can allot and assign the bearer and the IP to that UE. These VMs perform the CP measurements of this session, such as authentication, security, charge allowance, and IP assigning.
- Step 2 depicts the association with the destination UE to begin the session and exchange data between it and the source.

Figure 6 explains the procedure for achieving this association. When the user contacts SVeNB, the latter checks the location of the UE by the virtual MME (VMME) to verify the position and authentication through the interaction with the virtual HSS (VHSS). A virtual machine (P/S-GW) acts as a gateway or mobile IP anchor of SVeNB to the Internet, with the radio bearer and IP address being assigned in this stage. The connection of two users that are covered by a different coverage area of RRHs is easily prepared and executed by SVeNB. This is owing to the information of RRHs, such as frequency bands and IP addresses, having been saved and ready to use by SVeNB. Hence, SVeNB can easily overcome the complication of communication between two supported by different RRHs. In other words, users do not need to contact the BBU pool and the CN to allow for session establishment. As a result, the End-to-End delay time decreases due to no more demand for processing to achieve the session connection.

At the end of these steps, the user has two choices. The first, is to connect to another user, which is also covered by the same SVeNB or it covered by another one. The second choice is to connect to the Internet via the direct fiber link,

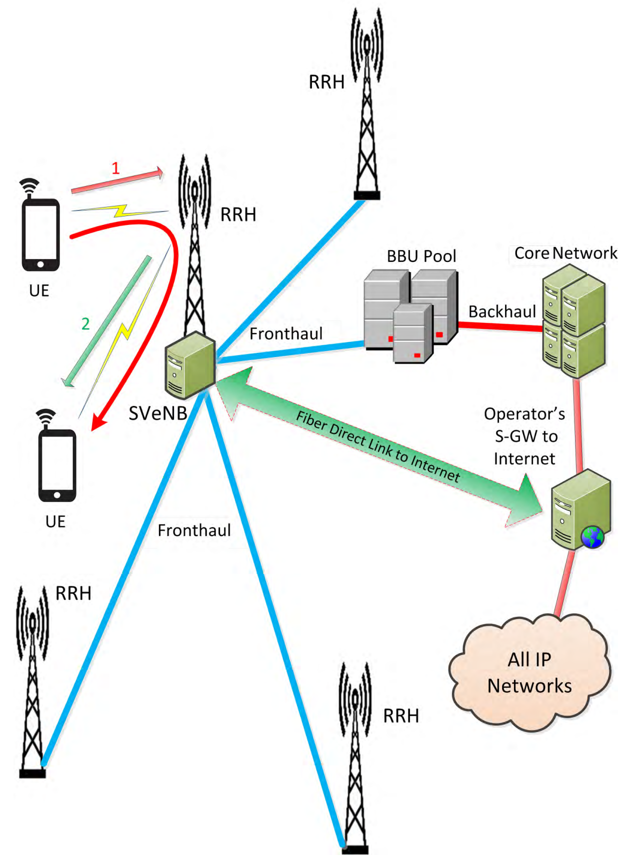

FIGURE 5. The proposed SVeNB End-to-End CP session establishing.

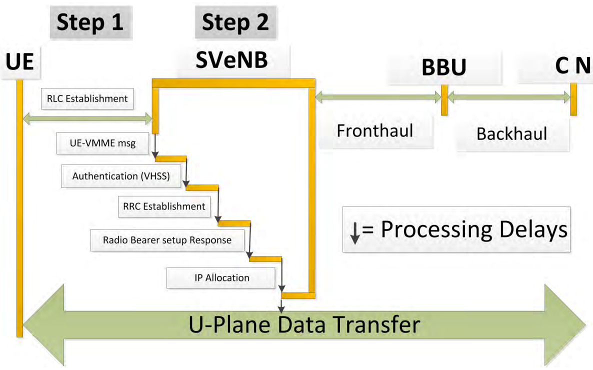

FIGURE 6. The proposed SVeNB End-to-End CP processing steps.

which has been proposed to connect SVeNB directly to the mobile operator's S-GW in case the UE requires surfing of the Internet, besides to the regular path (SVeNB-BBU pool-CN-S-GW path).

We assumed in the C-RAN system, the delays time of a transmission between a user and RRH is t_{LRC} which implies the radio link control delay. The link delay between the RRH and BBU pool which stands for the fronthaul delay is (t_F), and the delay between the BBU pool and the CN as the backhaul delay is (t_B). D_T is the summation of the transmission delays of the radio bearer of the RRH, the fronthaul, and the backhaul links. Hence, the total transmission delay can be represented by the Equation 1,

$$D_T = t_{LRC} + t_F + t_B \tag{1}$$

The processing delay time in the BBU pool is t_{PBB}, which expresses the overall time that is consumed by the BBU pool for the CP signal to initiate the association. The processing delay time that is spent by the CN is t_{PCN}, which denotes the overall time that is consumed by the CN for control information data to establish that connection. Hence, the summation of the processing delays of the BBU pool and CN is D_P. It can be estimated by,

$$D_P = t_{PBB} + t_{PCN} \tag{2}$$

D_Q is the summation of queuing delay time in the BBU is t_{QBB}, and the queuing delay time in the CN is t_{QCN}. Consequently, D_Q can be expressed by,

$$D_Q = t_{QBB} + t_{QCN} \tag{3}$$

Consequently, the total delay time D can be expressed with the following equations,

$$D = D_T + D_P + D_Q \tag{4}$$

The processing and queuing delay times in the RRHs are very small, so it can be ignored. All these time delays are introduced by the C-RAN and the CN just for a session establishment of End-to-End association (i.e. only CP messages exchange). Whereas with SVeNB, the equivalent connection can be made, but with the elimination of D_T. This is because this delay time has already been consumed by the first attachment of the BBU pool and the CN by SVeNB when the profiles of the subscribers are manipulated and sent to the SVeNB. The processing and Queuing delay times are reduced because they are processed by one physical device through many VMs supported by SVeNB. This leads to a decrease in the period of the delays time of the processing and the queuing. Therefore, to compare between the SVeNB and C-RAN, we should take into consideration those facts. With SVeNB the link delay time will be dropped from the delay calculation, whilst the processing and queuing delays should be estimated as a VM. Hence, the delay time equation for SVeNB will be,

$$D_S = D_{PS} + D_{QS} \tag{5}$$

where, D_S is the total delay time in SVeNB, D_{PS} is the processing delay, and D_{QS} is the queuing delay in the SVeNB. The D_{PS} and D_{QS} notably rely on the specifications of the SVeNB physical hardware and capital expenditure can decrease this delay through hardware with high specifications.

A. RADIO LINK DELAY ANALYSIS

According to 3GPP TS 36.213 and TS 36.306, for transmission in mobile networks, the Radio Link Control (RLC) has been using in the LTE to enhancement the Frame Error Rate (FER), according to the condition and availability of the Bandwidth (BW) channels. However, we assume using different BW for LTE (5, 10, 15, and 20 MHz) and then, the capacity of each channel can be found by a simple calculation. Consequently, the number of frames per packet (k) can be calculated for each channel rate. The LTE frame duration τ equals to 10 ms. Also, assume modulation scheme QPSK is used. The 1526 bytes is a Maximum Transmission Unit (MTU) for IPv4 packets. While IPv6 can convey much larger packets that make it well matched with the LTE system, which can handle the large size of packet or frame. The LTE frame is known as a Transport Block (TB) and it contains numerous sub-frames. Every sub-frame consists of two slots, each of which has six or seven OFDM symbols according to the extended or normal cyclic prefix, respectively. The number of bits in each symbol depends on the modulation scheme. In LTE, the maximum control message size does not exceed 10% of the sub-frame that is assigned to the UE, with the number of bits (b) in one sub-frame being:

$$\begin{aligned} b &= 2(RB) \times 12\ (sub - carriers) \\ &\quad \times 7\ (assuming 7\ OFDMsymbols) \\ &\quad \times 2\ (slotspersub - frame) \times M \\ &= 336 \times M \end{aligned}$$

M represents the modulation scheme. If the QPSK is used for control messages, then M equals 2. So, 336 x 2 x 0.1 (10% as assumed above) $\approx$ 68 bits. Consequently, the number of bytes per Transmission Time Interval (TTI) can be calculated for the specific channel as the channel BW multiplied by TTI divided by 8, for instant, 10 Mbps x 10 ms x$\frac{1}{8} = 1250$ Bytes. *This leads to the smallest transmission channel capacity that will cover the largest transmitted control message.*

Assume the probability for QPSK to transmit an RLC frame successfully is:

$$P_s = 1 - P(B_n) \tag{6}$$

The frames that are dropped or lost is denoted as B_n, and n is the number of re-transmission trial (typically 3). The probability of effective packet loss is:

$$P_f = P(B_n) = P[P(2-P)]^{\frac{(n+n^2)}{2}} \tag{7}$$

Assuming that the LTE channel operates under normal conditions, the probability of the FER is p $= 10^{-2}$, and n $\leq$ 3, the P_f can be calculated as follows:

$P_f = 10^{-2}[(2 \times 10^{-2} - 10^{-4})]^6 = 6.21 \times 10^{-13}$

From [26], [27], and [28], the typical value of the propagation delay is $D_{pr} = 100$ ms. Thus, the equation of RLC transmission delay can be calculated as,

$$\begin{aligned} t_{RLC} &= D_{pr} + \frac{[P_f - (1-P)]}{P_f^2} \\ &\quad \times |\sum_{j=1}^{n} \sum_{i=1}^{j} P(C_{ij}) \\ &\quad \times [2jD_{pr} + (\frac{j(j+1)}{2}) + i) \times \tau]| \end{aligned} \tag{8}$$

The first frame is received by end point properly is denoted as $C_{(ij)}$, at re-transmission ith of the transmitted frame at jth re-transmission trial.

B. FIBRE LINK DELAY ANALYSIS

The speed of light v decreases when it transfers into fiber optics cables due to the refractive index n_r of the material made these cables. The speed of light c in free space is 299792.458 km/s. In the single-mode optical fiber transmission, the standard wavelengths are 1310 nm and 1550 nm with refractive indices 1.4676 and 1.4682, respectively. Then, the delay time for a 1 km length of the fiber and a different refractive index according to a particular wavelength can be calculated by Equation 9,

$$t_\lambda = \frac{1km}{v_\lambda} = \frac{(1km \times n_\lambda)}{c} \tag{9}$$

The delay times for wavelength 1310 nm and 1550 nm are 4.895 μs and 4.897 μs, respectively [29]. So, for the Round Trip Time (RTT) for these values should be doubled. In the C-RAN architecture, the distance between the BBU and RRH is restricted by processing and transmitting delay times which should be not more than 3 ms (the period from uplink to downlink) [30]. In other words, when increasing the distance between the BBU and RRH, the processing delay of the former should decrease. The maximum distance of fiber optics cable that links the BBU pool and the RRHs can be given by solving the following equation

$$M_d = \frac{RTT \times c}{2 \times n_\lambda} \tag{10}$$

Equation 10 can be used to determine the maximum distance between the BBU and RRH with different kinds of optical fiber link for a specific λ.

C. QUEUING DELAY ANALYSIS

Queuing delays between the sender and the receiver UEs depend upon the amount of data packets in every specific queue at each entity in the network. The queuing delay time for establishing the EPS session can be represented as the summation of the delays of all the entities involved in initiating the session of the CN network and the BBU queuing delays. AS these entities successively process its functions due to some of the entities (*Master*) administrator and control the other entities (*Slave*). In other words, some decisions determine the beginning or lunching of other function to make decisions. For example, the decisions regarding authentication and allowance charging functions are first made, then the other functions are implemented to make the final decisions to establish a session connection. The final decisions are yielded from execution the functions of diverse entities to establish this connection. As a result, we can consider the M/M/1 queue assumption of CP messages at these entities. We proposed for each entity of the EPS network a queuing model of M/M/1 queue and Poisson signaling arrival rate process. For M/M/1, if the first input rule is Poisson, then the subsequent stage input is additionally Poisson and independent of the input process and so on [31], [32].

Assume Z is the number of entities (the BBU and the CN) included in session establishment, δ is the Poisson arrival process rate (packet/sec), and μ is the transmission arrival packet rate of the queue (packets/sec). Every element allows a traffic load of $\rho_i = \delta_i/\mu, i = 1, 2, \ldots, Z$. Moreover, in mobile communication networks some of the servers depend on others to make their decisions, i.e. the *ith* entity receives traffic from the leader (*Master*) entity (any chief entity could be either a leader or follower (*Slave*) as per the instance of the necessity of information preparing among them) with a packet arrival rate δ_i. Consider the queuing delay time at the recipient buffer only (for simplicity) [33]. The expected sum of the queuing delay time is the sum of the expected queues in tandem at every element, which can be represented as follows:

$$E[X] = \sum_{i=1}^{Z} E_i\left[\frac{1}{\mu_i}\right] \tag{11}$$

where, $X = 1/\mu$ is the service delay time of the entity, and μ is the transmission packet rate of the queue. These amounts of delays can be determined by formulas that are derived by the Markov chain. Then, the possibility of there being packets in the queue is, $P = \rho^N(1-\rho)$, where, N is the number of packets that are sent by a channel, and $\rho = \delta/\mu$, from this we can get $N = \rho/(1-\rho)$. So, the delay time for one entity is,

$$t_Q = \frac{N}{\delta} = \frac{\rho}{\delta(1-\rho)} = \frac{1}{(\mu-\delta)} \tag{12}$$

For the C-RAN system, the total queuing delay time can be given as:

$$D_Q = \sum_{i=1}^{Z} \frac{1}{(\mu_i - \delta_i)} \tag{13}$$

Whereas in SVeNB the queueing model is M/M/m [33] because its VMs perform m jobs and consequently, the queuing delay time will be expressed as [2]:

$$D_{QS} = \sum_{i=1}^{Z} \frac{1}{(m\mu_i - \delta_i)} \tag{14}$$

D. PROCESSING DELAY TIME

To estimate the processing delay time that is required to build up a session in the EPS, we need to delimit the time that is consumed per entity to complete its responsibility. Authors in [30] mentioned that the BBU processing for one round trip is 3 ms. The EPC entities (servers) need to perform further processing on arrived packets, i.e. the entity should decapsulate the arrived packets to take a decision and then encapsulate them again. The processing delay time depends upon the packet length. Let L denotes the IP length (32 bit or 128 bit), M_s is the machine word size, W represents arrived word size, R is the user profile record size, and S represents the size of the server's processor architecture, e.g., 32 or 64 bits of the processor of an entity. We propose that the memory access correlates with the lookup delay time at around 100 ns. So, the processing delay time for one entity can be obtained from [34]:

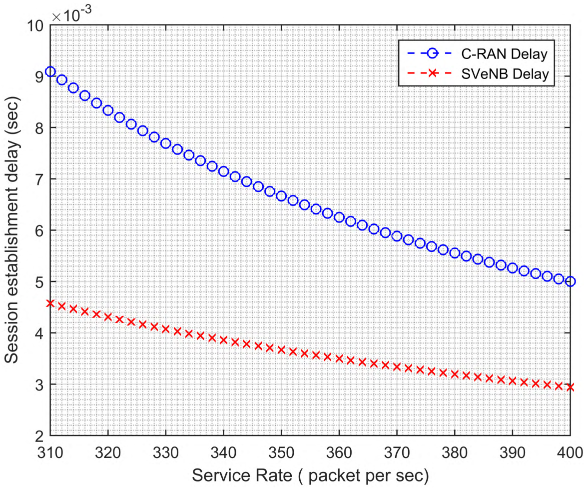

FIGURE 7. Session establishment CP delay with service rate comparison.

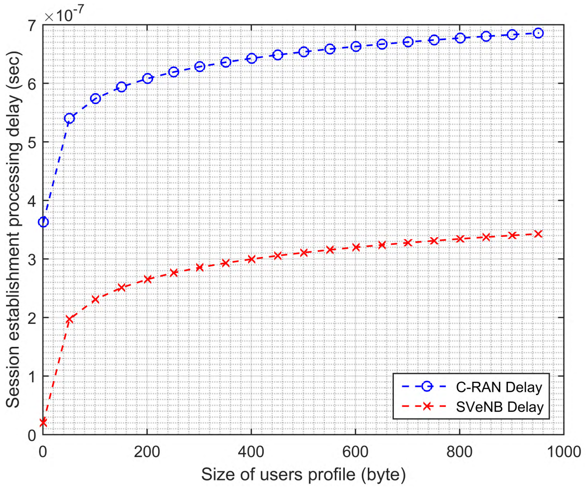

FIGURE 8. Session establishment processing delay and user profiles size.

$$t_P = 100\frac{W}{M_s} \times \left[log_{sys}R + \frac{L}{S}\right] \tag{15}$$

Next, we have derived the total processing delay time equation as:

$$D_P = \sum_{i=1}^{Z}\sum_{j=1}^{F} 100\frac{W_{ij}}{M_{si}} \times \left[log_{sys}R_{ij} + \frac{L_j}{S_i}\right] \tag{16}$$

where, Z represents the number of entities of the EPS, F denotes the number of the user's messages to a specific entity. *j*th is user's parameters to be treated by *i*th entity, and *sys* is the system constant [2].

V. PERFORMANCE EVALUATION ANALYSIS

This section discusses the results of the local UEs' End-to-End delay control messages, with consideration of both SVeNB and C-RAN. Figure 7 explains the relationship between the service rate and the End-to-End delay of the control signaling to establish a session for local UEs that are covered by a base station. At the service rate of 310 packets/second SVeNB performs with a delay time of almost 4.5 ms. Whilst in C-RAN it is slightly more than 9 ms. Both systems produce lower values in the delay time when increasing the average service rate. However, the SVeNB processing delay time becomes one-third of that of C-RAN at the service rate 400 packet/second. The performance of SVeNB for control plane signaling to establish End-to-End session is at least double of the C-RAN in the case of the low value of average service rates. The delay time reduces with increasing the serviced packets rate. When initiating the session, SVeNB and C-RAN almost work in a similar way (i.e. the initial access to set up a session establishment), with each executing almost the same procedure to set up the RACH for every user. However, following the setting up of the RACH the delay time for SVeNB becomes much less than with C-RAN due to the former not needing to connect the BBU and the CN, because all the information that it requires to achieve a session is already stored into its storage unit [2].

Figure 8 explicates SVeNB's superiority over C-RAN when serving the same number of served user profiles. The figure exhibits that when a user attempts to establish a connection, both systems take a long time of processing in the beginning, due to having to prepare the associated data of the user to switch from the inoperative to operative state. Subsequent to that, both systems arrive in a steady state case and the processing delay time experienced with SVeNB is around 0.2 μs at 100 bytes of user profile size. In contrast, C-RAN consumes approximately 0.6 μs for the same user profile size. Expanding the size of the user profile in the range of 100 times will provide an additional delay time of about 0.1 μs in both systems for the steady-state case.

In C-RAN, the BBU and the CN serve several cells and hence, the processing delay time grows according to the increase in the number of served cells. As a result, the processing delay time in C-RAN is greater than SVeNB, this being due to the massive number of user profiles and growth in their size. Intuitively, the base station should wait for more time to be served by the C-RAN structure.

Fig. 9 shows the average delay versus packet arrival rate for a single user. Considering the transmission, queuing, and processing delays, SVeNB provides higher performance than C-RAN in delivering data to a user for the same circumstances. This means that the data rate is increased as the total delay (transmission, queuing, and processing) is decreased, when comparing the attainment between SVeNB and C-RAN. Moreover, the average delay in C-RAN expands dramatically due to the accumulation of different devices which cause delay, while SVeNB does not suffer from such delays. This fact is true if we exclude the first contact to the BBU and the CN by SVeNB for acquiring the profiles of the users. The provided service rate of a network is influenced by the number of users that are served by it. That is, the service rate decreases with an increasing number of users in the network. This degradation is the result of the numerous operations performed by separated physical machines in the C-RAN network to perform a particular task. SVeNB overcomes this obstacle through virtualizing and mimicking these separated servers and functions, whilst gathering them into a single physical device. Fig. 10 presents the service rate versus the number of users. Also, the figure shows the performance of the proposed SVeNB against C-RAN network. In practical life, UE needs to communicate with the BBU and the CN to perform the important measurements to initial the session. Fig. 11 illustrates the average delay of transmission links that connect the RRH, BBU, and CN. At the starting point of the graph, both SVeNB and C-RAN required the same time to begin a session. SVeNB has the ability to handle and process the requirements of a session without the need to communicate with the BBU pool and CN again. Conversely, the C-RAN network cannot initiate a session without contacting them more than once and thus, the line graph of SVeNB drops to a minimum value, whereas that of C-RAN rises to a maximum one.

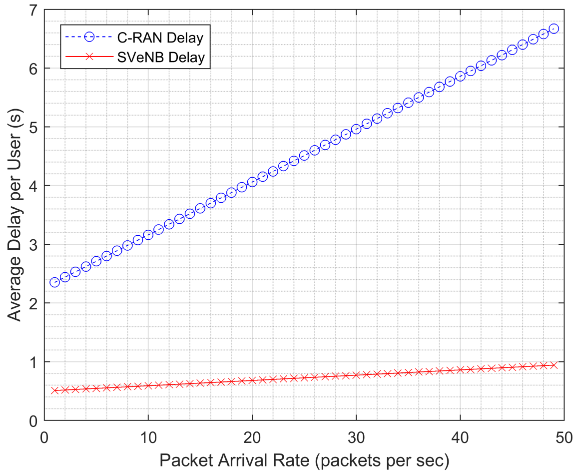

FIGURE 9. Average delay per user against packets arrival rate.

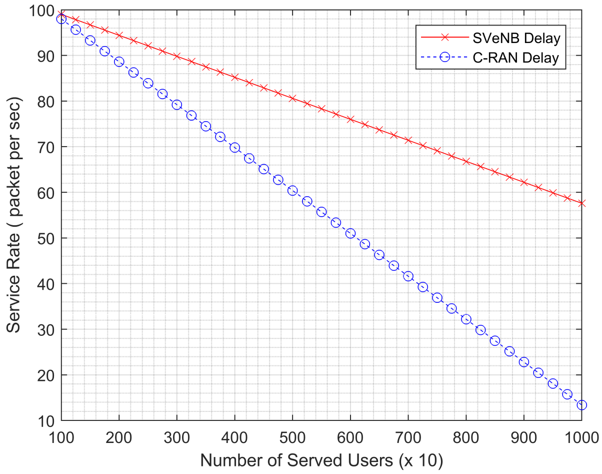

FIGURE 10. Impact of increasing number of users on service rate.

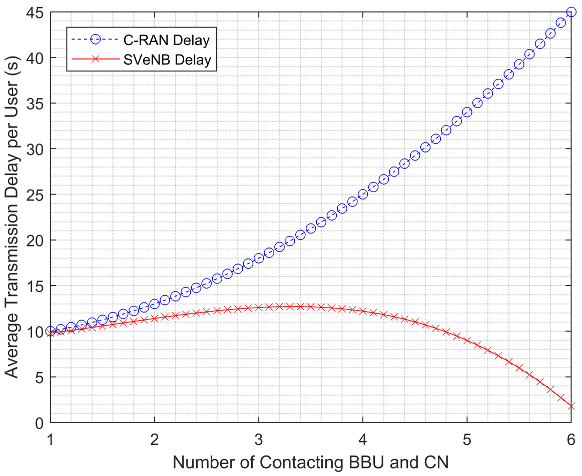

FIGURE 11. **Average transmission delay per user and communication with the BBU and CN.**

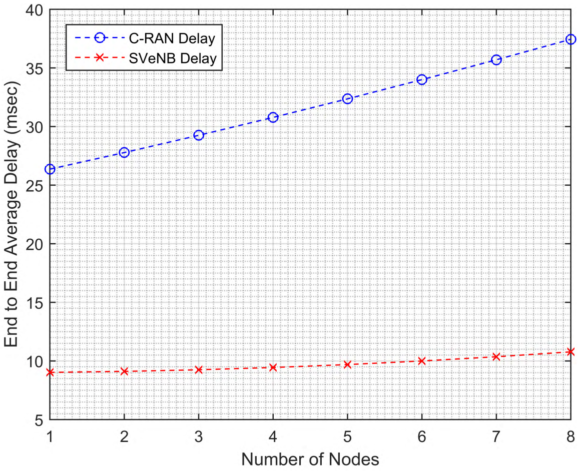

FIGURE 12. **Average End-to-End delay.**

Fig. 12 exhibits the End-to-End average delay comparison between SVeNB and C-RAN. This increases as the number of nodes increases due to each node (physical or virtual entity) suffering from its own delays, such as transmission, queuing, and processing. SVeNB can handle multiple criteria for user profiles at the same time and on the same server. This procedure leads to a reduction in the End-to-End delay time required to establish a session between users. Moreover, the End-to-End average delay in C-RAN grows dramatically due to the successive delays of different devices that include establishing a connection. While SVeNB does not suffer from the accumulative delays as well as in C-RAN which needs to acquire the profiles of the users from different servers.

VI. CONCLUSION

This paper has presented a novel base station idea based on virtualization technologies and the End-to-End delay time has been analyzed. The outcomes indicate that the delay time to establish the End-to-End association by the proposed SVeNB is less than that by the C-RAN. Mobile networks need to make substantial progress to meet the requirements of 5G networks and beyond. This requires more interaction between the new technology of the physical infrastructure and VMs to achieve complete virtualized networks. The most significant delay time is consumed by the processing of the arrived control data packets from/to the UE and CN for establishing the End-to-End link. Most notably, the processing delay of the control plane messages is where the highest delay time occurs. Moreover, control plane signaling in best conditions requires more than 60 ms to establish a session. Our proposal concentrates on lowering this delay time by virtualizing and slicing the primary functionalities of the BBU pool and CN. Moreover, bringing these functions close to the end user enables SVeNB to produce a significant reduction in control plane delay time. A comparison has been performed with the results that were obtained by SVeNB and the C-RAN emulating approach. The comparison involved variant values of average service rates. SVeNB exhibited around 62% less delay time than C-RAN. In addition, the performance of SVeNB is higher than of the C-RAN by 68% in terms of user profile processing. Our recommendation is that a considerable solution to overcoming the End-to-End delay problems is by virtualizing and slicing functions of the BBU and CN. Also, this requires bringing those virtualized functions close to the end user. The semi-decentralization of the processing functions mitigates the load on the BBU pool and CN.

REFERENCES

[1] A. Khan, D. Jurca, K. Kozu, W. Kellerer, and M. Yabusaki, "The reconfigurable mobile network," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, Jun. 2011, pp. 1–5.

[2] F. A. Yaseen, N. A. Al-Khalidi, and H. S. Al-Raweshidy, "Smart virtual eNB (SVeNB) for 5G mobile communication," in *Proc. 2nd Int. Conf. Fog Mobile Edge Comput. (FMEC)*, May 2017, pp. 88–93.

[3] M. Karakus and A. Durresi, "A survey: Control plane scalability issues and approaches in software-defined networking (SDN)," *Comput. Netw.*, vol. 112, pp. 279–293, Jan. 2017.

[4] F. Olivier, G. Carlos, and N. Florent, "New security architecture for IoT network," *Procedia Comput. Sci.*, vol. 52, pp. 1028–1033, Jun. 2015.

[5] M. Yang, Y. Li, D. Jin, L. Zeng, X. Wu, and A. V. Athanasios, "Software-defined and virtualized future mobile and wireless networks: A survey," *Mobile Netw. Appl.*, vol. 20, no. 1, pp. 4–18, Feb. 2014.

[6] B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1617–1634, 3rd Quart. 2014.

[7] R. Alvizu, G. Maier, N. Kukreja, A. Pattavina, R. Morro, A. Capello, and C. Cavazzoni, "Comprehensive survey on T-SDN: Software-defined networking for transport networks," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2232–2283, 4th Quart. 2017.

[8] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca, and J. Folgueira, "Network slicing for 5g with sdn/nfv: Concepts, architectures and challenges," Mar. 2017, *arXiv:1703.04676*. [Online]. Available: https://arxiv.org/abs/1703.04676

[9] P. Demestichas, A. Georgakopoulos, D. Karvounas, K. Tsagkaris, V. Stavroulaki, J. Lu, C. Xiong, and J. Yao, "5G on the horizon: Key challenges for the radio-access network," *IEEE Veh. Technol. Mag.*, vol. 8, no. 3, pp. 47–53, Sep. 2013.

[10] Q. Yan, F. R. Yu, Q. Gong, and J. Li, "Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 602–622, 1st Quart. 2016.

[11] Z. Zhu, P. Gupta, Q. Wang, S. Kalyanaraman, Y. Lin, H. Franke, and S. Sarangi, "Virtual base station pool: Towards a wireless network cloud for radio access networks," in *Proc. 8th ACM Int. Conf. Comput. Frontiers*, May 2011, p. 34.

[12] G. Bhanage, I. Seskar, R. Mahindra, and D. Raychaudhuri, "Virtual basestation: Architecture for an open shared Wimax framework," in *Proc. 2nd ACM SIGCOMM Workshop Virtualized Infrastruct. Syst. Archit.*, Sep. 2010, pp. 1–8.
[13] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, "NVS: A substrate for virtualizing wireless resources in cellular networks," *IEEE/ACM Trans. Netw.*, vol. 20, no. 5, pp. 1333–1346, Oct. 2012.
[14] W. Kiess, P. Weitkemper, and A. Khan, "Base station virtualization for OFDM air interfaces with strict isolation," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, Jun. 2013, pp. 756–760.
[15] S. Costanzo, D. Xenakis, N. Passas, and L. Merakos, "OpeNB: A framework for virtualizing base stations in LTE networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 3148–3153.
[16] K. Nakauchi, Z. Lei, Y. Shoji, Y. Kitatsuji, M. Ito, and H. Yokota, "Bring your own network—Design and implementation of a virtualized WiFi network," in *Proc. IEEE 11th Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2014, pp. 483–488.
[17] X. Xu, H. Zhang, X. Dai, Y. Hou, X. Tao, and P. Zhang, "SDN based next generation mobile network with service slicing and trials," *China Commun.*, vol. 11, no. 2, pp. 65–77, Feb. 2014.
[18] A. W. Dawson, M. K. Marina, and F. J. Garcia, "On the benefits of RAN virtualisation in C-RAN based mobile networks," in *Proc. 3rd Eur. Workshop Softw. Defined Netw.*, Sep. 2014, pp. 103–108.
[19] X. Wang, S. Thota, M. Tornatore, H. S. Chung, H. H. Lee, S. Park, and B. Mukherjee, "Energy-efficient virtual base station formation in optical-access-enabled cloud-RAN," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1130–1139, May 2016.
[20] K. Amiri, M. Duarte, J. R. Cavallaro, C. Dick, R. Rao, and A. Sabharwal, "FPGA in wireless communications applications," *DSP Embedded Real-Time Syst.*, vol. 1, pp. 75–101, Jan. 2012.
[21] J. Heinonen, P. Korja, T. Partti, H. Flinck, and P. Pöyhönen, "Mobility management enhancements for 5G low latency services," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, May 2016, pp. 68–73.
[22] H. Zhang, N. Liu, X. Chu, K. Long, A. Aghvami, and V. C. M. Leung, "Network slicing based 5G and future mobile networks: Mobility, resource management, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 138–145, Aug. 2017.
[23] V. G. Nguyen, A. Brunstrom, K.-J. Grinnemo, and J. Taheri, "SDN/NFV-based mobile packet core network architectures: A survey," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1567–1602, 3rd Quart. 2017.
[24] S. Zarei, "Channel coding and link adaptation," in *AusgewÃdhlte Kapitel aus Nachrichtentechnik, WS*. Dec. 2009, pp. 1–14.
[25] G.-Y. Lin, S.-R. Chang, and H.-Y. Wei, "Estimation and adaptation for bursty LTE random access," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2560–2577, Apr. 2016.
[26] G. D. L. Roche, A. Alayón-Glazunov, and B. Allen, *LTE-Advanced and Next Generation Wireless Networks: Channel Modelling and Propagation*. Hoboken, NJ, USA: Wiley 2012.
[27] M. Ulvan, R. Bestak, and A. Ulvan, "IMS signalling in ltebased femtocell network," in *Proc. 4th Int. Conf. Mobile Ubiquitous Comput., Syst., Services Technol.*, 2010, pp. 1–7.
[28] H. Fathi, S. S. Chakraborty, and R. Prasad, "On SIP session setup delay for VoIP services over correlated fading channels," *IEEE Trans. Veh. Technol.*, vol. 55, no. 1, pp. 286–295, Jan. 2006.
[29] V. Bobrovs, S. Spolitis, and G. Ivanovs, "Latency causes and reduction in optical metro networks," *Proc. SPIE*, vol. 9008, Feb. 2014, Art. no. 90080C.
[30] I. Al-Samman, A. Doufexi, and M. Beach, "A C-RAN architecture for LTE control signalling," in *Proc. IEEE 83rd Veh. Technol. Conf. (VTC Spring)*, May 2016, pp. 1–5.
[31] E. Gelenbe, G. Pujolle, and J. Nelson, *Introduction to Queueing Networks*. Hoboken, NJ, USA: Wiley 1998.
[32] M. Panda, H. L. Vu, M. Mandjes, and S. R. Pokhrel, "Performance analysis of TCP NewReno over a cellular last-mile: Buffer and channel losses," *IEEE Trans. Mobile Comput.*, vol. 14, no. 8, pp. 1629–1643, Aug. 2015.
[33] H.-L. To, S.-H. Lee, and W.-J. Hwang, "A burst loss probability model with impatient customer feature for optical burst switching networks," *Int. J. Commun. Syst.*, vol. 28, no. 11, pp. 1729–1740, Jul. 2015.
[34] B. Lampson, V. Srinivasan, and G. Varghese, "IP lookups using multiway and multicolumn search," *IEEE/ACM Trans. Netw.*, vol. 7, no. 3, pp. 324–334, Jun. 1999.

FOUAD A. YASEEN received the B.Sc. and M.Sc. degrees in electronic and communications engineering from the University of Technology, Baghdad, Iraq, in1994 and 2010, respectively. He is currently pursuing the Ph.D. degree with Brunel University London. His research interests include wireless communication networks and mobile communication systems.

HAMED S. AL-RAWESHIDY (SM'03) received the B.Eng. and M.Sc. degrees from the University of Technology, Baghdad, Iraq, in 1977 and 1980, respectively, and the Ph.D. degree from Strathclyde University, Glasgow, U.K., in 1991. He is currently a Professor of communications engineering. He is also the Director of the Wireless Networks and Communications Centre, Brunel University London, U.K. He has published more than 380 papers in international journals and refereed conferences.

• • •