Journal of Network and Computer Applications 191 (2021) 103083

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

# Fog radio access network optimization for 5G leveraging user mobility and traffic data

Longbiao Chen [a], Zhihan Jiang [a], Dingqi Yang [b], Cheng Wang [a], Thi-Mai-Trang Nguyen [c,*]

[a] Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Informatics, Xiamen University, China
[b] University of Macau, Macau SAR, China
[c] Sorbonne University, UMR 7606, LIP6, France

ARTICLE INFO

*Keywords:*
Fog-RAN
5G
Network optimization
Radio access network
Big data analytics

ABSTRACT

The surging data traffic and dynamic user mobility in 5G networks have posed significant demands for mobile operators to increase data processing capacity and ensure user handover quality. Specifically, a cost-effective and quality-aware radio access network (RAN) is in great necessity. With the emergence of fog-computing-based RAN architecture (Fog-RAN), the data processing units (BBUs) can be separated from base stations (RRHs) and hosted in distributed fog servers, where each server accommodates a community of RRHs to handle data traffic and user handover. The key problem in Fog-RAN optimization is how to cluster complementary RRHs into communities and allocate adequate numbers of BBUs for the fog servers, since real-world traffic and mobility patterns are highly dynamic to model, and it is not trivial to find an optimal RRH clustering and BBU allocation scheme from potentially enormous numbers of candidates. In this work, we propose a data-driven framework for cost-effective and quality-aware Fog-RAN optimization. In the RRH clustering phase, we build a weighted graph model to characterize user mobility patterns across RRHs, and propose a size-constrained community detection (SCUD) algorithm to cluster RRHs into communities with frequent internal handover events. In the BBU allocation phase, we formulate BBU allocation in each community fog server as a set partitioning problem, and propose a column-reduced integer programming (CLIP) algorithm to find optimal BBU allocation schemes that maximize BBU utilization rate. Evaluations using two large-scale real-world datasets collected from Ivory Coast and Senegal show that compared to the traditional RAN architecture, our framework effectively reduces the average handover overhead to 12.8% and 27.3%, and increases the average BBU utilization rate to 49.7% and 52.3% in both cities, respectively, which consistently outperforms the state-of-the-art baseline methods.

## 1. Introduction

The number of mobile subscriptions is growing rapidly over the past decades, reaching around 7.9 billion worldwide in 2019 (Ericsson, 2019). In the fifth generation (5G) era, 3.5 billion Internet-of-Things (IoT) devices will be connected to mobile network infrastructures in five years with new capabilities and use cases, such as autonomous cars, shipping drones, and industrial robots (Ericsson, 2019). The tremendous traffic volume generated by these heterogeneous and dynamic mobile subscribers has posed great challenges to the radio access network (RAN) architecture of 5G networks (J. Research, 2011). On one hand, the *data traffic* generated by these mobile subscribers is growing explosively as a large volume of multimedia contents are delivered (Cisco, 2016). In order to accommodate such surging traffic demand, operators need to expand the RAN scale and increase its capacity, which leads to increasingly high capital expenditure (CAPEX) and operating expenditure (OPEX) (J. Research, 2011; Checko et al., 2015). On the other hand, *user mobility* in 5G networks is highly dynamic, ranging from large-scale crowd movement to autonomous drone fleet migration (Andrews et al., 2014). Consequently, it is becoming extremely difficult for operators to ensure the quality of services in RAN, such as user connectivity and handover delay. Therefore, a *cost-effective and quality-aware* RAN architecture is of great necessity for the success of 5G networks (I et al., 2014; Gandotra and Jha, 2017).

To address these challenges, Cloud Radio Access Network (Cloud-RAN) (C. M. R. Institute, 2011) has been proposed as a cloud computing solution for next generation (5G) radio access networks. Cloud-RAN envisions a centralized architecture where traditional base stations are

* Corresponding author.
*E-mail addresses:* longbiaochen@xmu.edu.cn (L. Chen), Thi-Mai-Trang.Nguyen@lip6.fr (T.-M.-T. Nguyen).

https://doi.org/10.1016/j.jnca.2021.103083
Received 27 January 2020; Received in revised form 11 December 2020; Accepted 5 April 2021
Available online 1 July 2021

divided into remote radio heads (RRHs) and baseband processing units (BBUs), and the BBUs are hosted and shared in a centralized cloud pool (Checko et al., 2015). Cloud-RAN reduces the deployment and operational costs as a result of centralized resource sharing, and user mobility can be managed in a unified and efficient manner (Checko et al., 2015). However, as Cloud-RAN requires a large volume of *fronthaul* traffic to be transmitted between BBUs and RRHs, the latency between RRHs and BBUs is not negligible due to bandwidth and distance limitations in the fronthaul (Peng et al., 2016). Moreover, the design of one centralized BBU pool may be vulnerable to jamming attacks and raise network security issues (Tian et al., 2017). With the growing popularity of IoT devices in the 5G era, the fronthaul latency and security issues of Cloud-RAN have become critical challenges to satisfy the requirements of latency-sensitive IoT applications and services (Santoyo-González and Cervelló-Pastor, 2018; Peng et al., 2015).

To overcome the disadvantages of Cloud-RANs, researchers have turned to *fog computing*, which extends the cloud computing paradigm in Cloud-RAN to the edge of the network (Cisco, 2015). In *fog-computing-based radio access network (Fog-RAN)*, instead of deploying a centralized BBU pool, the BBU functionalities are provided by distributed *fog servers* located close to the RRHs (Bonomi et al., 2012). Fig. 1 shows a typical Fog-RAN architecture, where the whole RAN is divided into several communities, and the RRHs in each community are connected to a fog server via high-speed optical fibers, and then routed to the core network. Fog-RAN alleviates the challenges of the existing RAN architectures in the following aspects. First, compared with Cloud-RAN, the fronthaul traffic volume and transmission latency between RRHs and BBUs are greatly reduced. Second, user mobility can be managed internally in each fog server without explicitly transferring session data across RRHs, and thus reduces handover overhead and improves connection quality. Third, the BBU processing capacities in the fog servers can be shared across the connected RRHs to increase the utilization rate and thus reduce the operational costs. Furthermore, Fog-RAN is considered to be a more secure architecture than Cloud-RAN (Ni et al., 2017; Saharan and Kumar, 2015). In Fog-RAN, data are processed in local fog servers, decreasing the dependency on the core network and increasing the independency among different fog servers. Selected security functions can be carried out in fog servers, and computing, storage, and networking tasks can be dynamically relocated among fog severs (Zhang et al., 2017a). If a fog server has been attacked, its tasks can be dynamically rerouted to the adjacent fog servers, demonstrating the robustness of the Fog-RAN architecture. Also, the fog servers and RRHs in the Fog-RAN architecture are corresponding to gNB-CUs and gNB-DUs in the 5G wireless system (Sigwele et al., 2018), and the fog-RAN architecture can be easily applied to 5G network. In summary, Fog-RAN provides a promising solution to cost-effective and quality-aware 5G network architecture (Peng et al., 2016).

In order to fully unlock the power of the Fog-RAN architecture, we need to design optimal schemes to cluster RRHs into communities and connect each community to a fog server, so as to minimize not only the fronthaul traffic and latency between RRHs and fog servers, but also the handover overhead across RRHs. Moreover, we need to allocate an adequate number of BBUs in each fog server to accommodate the RRH traffic demands, so as to maximize the BBU utilization rate. However, designing such an *optimal RRH clustering and BBU allocation scheme* for a Fog-RAN architecture is not trivial due to the following challenges.

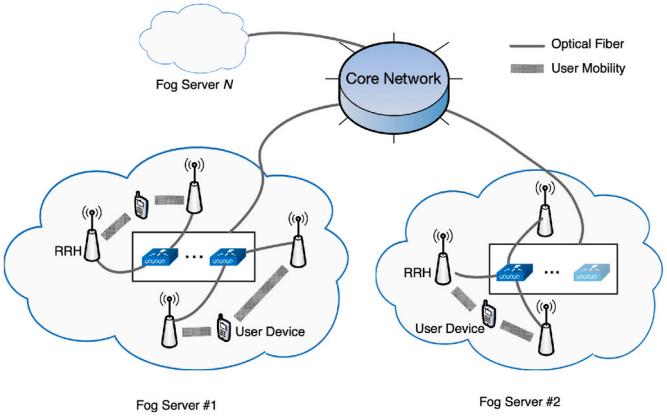



**Fig. 1.** An illustrative example of the Fog-RAN architecture.

1. **It is not easy to characterize user mobility patterns and traffic demands in real-world mobile networks.** In fact, the traffic generated in each RRH can vary significantly, depending on the number and types of connected user devices, the impacts of temporal contexts (e.g., weekdays or weekends), the intensities of human activities (e.g., commuting or eating), etc. Similarly, user mobility is driven by various latent factors, including human behaviors, IoT device tasks, city functions, etc., and demonstrates sophisticated spatial correlations. Existing work on traffic and mobility characterization usually employ probabilistic models for *simulation*, such as Poisson process (Taleb et al., 2017) for traffic and random walk process for mobility (Akyildiz et al., 2000), which may not be able to capture the spatial-temporal traffic and mobility dynamics in real-world networks.
2. **It is not trivial to design optimal RRH clustering schemes in Fog-RANs.** In Fog-RAN, by connecting a set of RRHs to a fog server, the objective is to reduce the user handover overhead among these RRHs, as well as reducing the fronthaul traffic volume and transmission latency between the RRHs and the fog server. To this end, the RRHs with frequent handover events across each other should be clustered to the same fog server to alleviate transferring a large amount of user session data across different fog servers (Liu et al., 2012). Meanwhile, the RRH community size, the geographic span of the fog server and its connected RRHs should be constrained within a range, so as to avoid fronthaul traffic jam and reduce transmission latency for latency-sensitive applications.
3. **It is not straightforward to design optimal BBU allocation schemes in fog servers.** In a fog server, BBUs are usually implemented as virtual machines (VMs) with a fixed capacity, while the traffic volume generated in each RRH can be highly dynamic. We need to allocate an adequate number of BBUs to accommodate the traffic demands of the connected RRHs. On the one hand, directly allocating one BBU for each RRH may lead to a low utilization rate. On the other hand, allocating inadequate number of BBUs for a fog server may result in traffic congestion and hinder the quality of service of the fog server. To this end, the RRHs connected to a fog server should be organized to share an adequate number of BBUs, such that the allocated BBU capacity is optimally utilized in each fog server.

With the emergence of big data and cloud computing technologies, a massive number of mobile network data can now be collected, stored, and processed in operators' infrastructures (Blondel et al., 2012; Zhang et al., 2017b). These mobile big data provide a great potential for us to understand the traffic demands and mobility patterns of mobile users, enabling researchers to design frameworks and algorithms for network optimization in a *data-driven* approach (Zheng et al., 2016; Chen et al., 2018). In this work, we propose a data-driven framework for Fog-RAN optimization leveraging the traffic demands and mobility patterns of mobile users. More specifically, we first extract the traffic volumes and user handover events of each RRH in a mobile network from large-scale, real-world Call Detail Record (CDR) datasets. We then cluster neighboring RRHs with frequent handover events into proper communities, and connect them to dedicated fog servers. Finally, in each fog server, we partition the connected RRHs into disjoint subsets and allocate BBUs for them, so that each subset of RRHs can share the BBU capacity with complementary traffic patterns and thus increase the BBU utilization

rate.

In brief, the **main contributions** of this work include:

- We propose a novel data-driven approach to optimize both operational cost and service quality for fog radio access networks. The proposed approach is capable of exploiting real-world data traffic demands and user mobility patterns to build a demand-responsive fog-RAN architecture.
- We propose a two-phase framework for RRH clustering and BBU allocation via distributed fog servers. In the *RRH clustering* phase, we build a *weighted graph model* to characterize user mobility patterns across RRHs, and propose a size-constrained community detection (SCUD) algorithm to cluster RRHs into communities with frequent internal handover events, and then connect each RRH community to a fog server. In the *BBU allocation* phase, we formulate BBU allocation in each fog server as a *set partitioning problem*, and propose a column-reduced integer programming (CLIP) algorithm to find optimal BBU allocation schemes that maximize the utilization rate.
- We evaluate the performance of our approach using two large-scale real-world datasets collected from Ivory Coast and Senegal. Results show that compared to the traditional RAN architecture, our framework effectively reduces the average handover overhead to 12.8% and 27.3%, and increases the average BBU utilization rate to 49.7% and 52.3%, respectively, which consistently outperforms the state-of-the-art baseline methods.

The rest of this work is organized as follows. We first present a literature review in Section 2, and then introduce the preliminaries and framework overview in Section 3. In Section 4 we propose the mobility-based RRH clustering algorithm, and in Section 5 we propose the traffic-based BBU allocation algorithm. We report the evaluation results and present case studies with real-world datasets in Section 6. Finally, we conclude our work in Section 7.

## 2. Related work

### 2.1. Radio access networks

The fast evolution of mobile networks have shown its great importance in modern urban communication systems (Hasan et al., 2011; Lu et al., 2013). Mobile network operators and researchers are continuously seeking for solutions to provide stable telecommunication, high speed data rate, and high quality of services to their users (C. M. R. Institute, 2011; Munaretto and Fonseca, 2007). However, the cost to build, operate and upgrade the network infrastructures is becoming increasingly expensive for mobile operators (I et al., 2014). As the deployment and commercial operation of 4G systems are reaching maturity, researchers and network operators worldwide have begun searching for next generation (5G) mobile network solutions (I et al., 2014).

#### 2.1.1. Cloud Radio Access Network

Cloud radio access network (Cloud-RAN) is targeted by worldwide mobile network operators as a typical realization of green and soft RAN architecture in 5G mobile networks (Checko et al., 2015). In 2010, IBM proposed wireless network cloud (WNC) (Lin et al., 2010). The WNC system exploits emerging cloud-computing technology and various wireless infrastructure technologies, such as remote radio head and software radio, to enable RAN resource processing operating in a cloud mode (Lin et al., 2010). In 2011, China Mobile Research Institute envisioned a cloud-based RAN architecture to provide mobile broadband Internet access to wireless customers with low bit-cost, high spectral and energy efficiency (C. M. R. Institute, 2011). Texas Instruments also proposed an enhanced version of its KeyStone multicore architecture to be used to create cloud base stations. For a comprehensive technology survey on Cloud-RAN, the readers are referred to (Checko et al., 2015).

One of the key vision in Cloud-RAN is to provide flexible and configurable data processing capacity according to the traffic demands (Andrews et al., 2014; C. M. R. Institute, 2011). Furno et al. (2016) coined such a vision as a *cognitive networking diagram*. To this end, cooperations among RRHs are necessary to cope with the fluctuations in traffic demands (Checko et al., 2015). However, as Cloud-RAN requires a large volume of *fronthaul* traffic to be transmitted between BBUs and RRHs, the latency between RRHs and BBUs are not negligible due to bandwidth and distance limitations in the fronthaul (Peng et al., 2016). Moreover, the design of one centralized BBU pool may be vulnerable to jamming attacks and raise network security issues (Tian et al., 2017). With the growing popularity of IoT devices in the 5G era, the fronthaul latency and security issues of Cloud-RAN have become critical challenges to satisfy the requirement of latency-sensitive IoT applications and services (Peng et al., 2015).

#### 2.1.2. Fog radio access network

To overcome the disadvantages of Cloud-RANs, researchers have turned to *fog computing*, which extends the cloud computing paradigm in Cloud-RAN to the edge of the network (Cisco, 2015). In fog-computing-based RAN architecture (Fog-RAN), instead of deploying a centralized BBU pool, the BBU functionalities are provided via distributed fog servers located close to the RRHs (Bonomi et al., 2012). Shih et al. (2017) introduced the Fog-RAN and its rationale in serving ultra low-latency applications. Zhang et al. (2017a) described the Fog-RAN architecture and discussed how the distinctive characteristics of Fog-RAN make it possible to efficiently alleviate the burden on the fronthaul, backhaul, and backbone networks, as well as reduce content delivery latencies. Fog computing is considered to be a more secure architecture than cloud computing due to its decentralized characteristics (Ni et al., 2017; Saharan and Kumar, 2015), but it also faces new security and privacy challenges (Yi et al., 2015). Mandlekar et al. (2014) presented a survey talking about how fog computing is used to defend data theft attacks. Tu et al. proposed a smart attack defense scheme for end users and a novel technique to tackle impersonation attacks in fog computing (Tu et al., 2018, 2020). Kumar et al. (2016) discussed the common security issues and proposed countermeasures in fog computing. By allowing dynamic relocation of the computing, storage, and control functions among fogs, the life circle management of the system and services can be more efficient and effective (Zhang et al., 2017a).

To unlock the power of the Fog-RAN architecture, many studies have been conducted to optimize Fog-RANs. Park et al. (2016) studied the joint design of cloud and edge processing for the downlink of a Fog-RAN, but this work only considered maximizing the delivery rate under fronthaul and enhanced RRH capacity constraints without optimizing handover overhead. Tandon et al. (Tandon and Simeone, 2016) and Sengupta et al. (2017) demonstrated the interplay of fronthaul, wireless and caching policies for the minimization of the delivery latency to develop the information-theoretic framework for the analysis of Fog-RANs. However, they focused on the worst-case end-to-end latency among end users without taking utilization into consideration. Xiang et al. (2020) proposed a deep reinforcement learning algorithm for Fog-RAN slicing considering the transmission delay without addressing the problem of caching and radio resource allocation. Pang et al. (2017) proposed a Fog-RAN model achieving the ultra-low latency by joint computing across multiple fog nodes and near-range communication at the edge without considering handover overhead and utilization optimization. Moreover, most of the existing works are based on simulation data, lacking insights into real-world traffic demands and mobility patterns. In this work, we exploit real-world data to build a demand-responsive Fog-RAN architecture with regard to RRH clustering and BBU allocation, efficiently increasing BBU utilization rate and reducing handover overhead as well as fronthaul latency.

#### 2.1.3. Fog radio access network and multiple-access edge computing

Multiple-access Edge Computing (MEC) is an extension of mobile computing through edge computing (Shi et al., 2016), defined as a platform providing IT and computing capabilities within the RAN (Giust et al., 2018), in close proximity to mobile subscribers. Similar to Fog-RAN, MEC is also a promising solution for the next generation (5G) access networks (Park et al., 2016; Li et al., 2017). Both Fog-RAN and MEC offload data from the cloud (Klonoff, 2017) and process data leveraging computing resources closer to end-nodes (Dinh et al., 2019; Yousefpour et al., 2019), and thereby mitigate latency issues (Tran et al., 2017; Peng et al., 2016). Fog-RAN is Edge networking seen from the piont of view of device constructors (Cisco, 2015; Bonomi et al., 2012), and MEC is Edge networking seen from the point of view of network operators (Porambage et al., 2018; Mao et al., 2017).

### 2.2. Mobile big data analytics

Mobile crowdsensing paradigms and operator's infrastructures can offer a massive number of mobile datasets (Wang et al., 2016, 2017; Guo et al., 2015). For example, the large-scale call detail records datasets released by Telecom Italia (Barlacchi et al., 2015) containing two-months of calls, SMSs and network traffic data from the city of Milan and the province of Trentino, Italy. Blondel et al. (2012) offered a large-scale anonymous call detail records datasets consisting of phone calls and SMS exchanges between five million of Orange's customers in Ivory Coast over half a year. These heterogeneous mobile big data have been applied to academic research and industrial analytics (Zheng et al., 2016), generating many interesting results (Chen et al., 2014, 2015; Yang et al., 2015; Tan et al., 2016). For example, Furno et al. (2016) applied the call detail records datasets released by Telecom Italia to facilitating the design and implementation of cognitive mobile networking. Besides, based on these datasets, Chen et al. (2018) proposed a deep-learning-based Cloud Radio Access Network (Cloud-RAN) optimization framework.

However, when it comes to the data-driven Fog-RAN optimization, it has not yet been widely studied in the literature. Gao et al. (2020) adopted data-driven bandit learning methods to integrate off-line history information into online learning to devise a cache placement scheme in Fog-assisted IoT systems. Luo et al. (2018) developed a data-driven method for fog-computing-aided process monitoring and control architecture design to optimize online performance in each fog computing node. Dao et al. (2017) proposed an adaptive resource balancing scheme for serviceability maximization in Fog-RAN with respect to a time-varying network topology issued by potential RRH mobilities. However, these works mainly focused on the optimization for resource allocation in each fog server and overlooked the fog server deployment optimization. Zhao et al. (2020) proposed a paradigm of federated learning-enabled intelligent Fog-RANs using the data collected by the nodes of the fog computing layer. However, the employed indivisible learning models are unable to support flexible computation offloading strategies, and thus the dispersive computation resources of fog servers cannot be optimized. In this work, we exploit large-scale real-world mobile open datasets to understand the traffic demands and mobility patterns in real networks. Then, based on the knowledge discovered from these mobile datasets, we cluster RRHs to deploy fog servers and find optimal BBU allocation schemes via distributed fog servers. The proposed framework decreases handover overhead, fronthaul latency, and fronthaul traffic, as well as increases the BBU utilization rate.

## 3. Preliminaries and framework overview

### 3.1. Preliminaries

Radio access networks (RANs) connect user equipments (UEs) to the core networks (CNs) through a set of base stations (BSs) deployed over a geographical area (Tse and Viswanath, 2005). Each base station provides the area with a network coverage for transmission of voice and data (Demestichas et al., 2013). In order to monitor and evaluate the processing capacity of base stations, large scales of anonymized statistical data have been collected by operators and made available for researchers (Zheng et al., 2016). In this work, we exploit the anonymized call detail record (CDR) data released by Orange Group via the Data for Development (D4D) Challenges (Blondel et al., 2012; de Montjoye et al., 2014). More specifically, we extract two city-scale datasets, each containing the communication traffic generated from base stations, and the user mobility trajectories across these base stations. The geographic positions of the base stations are also collected. Based on these datasets, we define the following preliminaries for data analytics.

**Definition 1.** ***Remote Radio Head (RRH):*** an RRH is a radio transceiver placed in a base station site to facilitate wireless communication between user devices and the network (Checko et al., 2015). We define an RRH as a triple $r = < id, lat, lng >$, where $id$ is the RRH identity, and $lat$ and $lng$ are the latitude and longitude coordinates of the RRH.

**Definition 2.** ***RRH Traffic Volume:*** the traffic volume of an RRH is defined as the quantity of radio resource units (Taleb et al., 2017) occupied by an RRH during a period of time, which can be derived from the total duration of calls, the overall volume of Internet data, etc. Specifically, we denote the traffic volume of RRH $r_i$ in the time slot $t$ as $\mathscr{T}(r_i, t)$.

**Definition 3.** ***RRH Handover Count:*** the handover count between a pair of RRHs is defined as the quantity of users moving between the two RRHs during a period of time. Specifically, we denote the handover count between RRH $r_i$ and RRH $r_j$ in the time slot $t$ as $\mathscr{H}(r_i, r_j, t)$.

**Definition 4.** ***Baseband Unit (BBU):*** a BBU is a device providing digital signal processing functionalities for the RRHs connected to it (Checko et al., 2015). In the proposed Fog-RAN architecture, BBUs are implemented as virtual machine instances with specific CPU, memory, and storage resources (Yang et al., 2013). Correspondingly, we define a BBU as a tuple $b = < id, param >$, where $id$ is BBU identity, and $param$ is the resource configuration parameters of the BBU virtual machine.

**Definition 5.** ***BBU Capacity:*** the capacity of a BBU is determined by its resource parameters, and measured in the same dimension as RRH. We denote the capacity of BBU $b_i$ as $c(b_i)$. In this work, for the simplicity of analytics, we consider BBUs with unified capacity $C$ in all fog servers, although this assumption could be easily extended by specifying a list of capacity configurations, and our method can easily adapt to it.

**Definition 6.** ***Fog Server:*** in the proposed Fog-RAN architecture, a fog server is defined as a distributed cloud server for general-purpose baseband processing, data caching, and other applications (Peng et al., 2016). In this work, we focus on the baseband processing functionality, and define a fog server as a triple $s = < id, R, B >$, where $id$ is the identity of the fog server, $R = \{r_i\}$ is the community of RRHs connected to the fog server, and $B = \{b_i\}$ is the set of BBUs allocated to accommodate the traffic and handover demands of the connected RRHs.

### 3.2. Framework overview

Based on the datasets and preliminaries, we propose a two-phase framework for data-driven Fog-RAN optimization, as illustrated in Fig. 2. In the mobility-based RRH clustering phase, we first extract the handover patterns across RRHs, and build a weighted graph to model user mobility patterns with geographic constraints. We then propose a size-constrained community detection (SCUD) algorithm to cluster neighboring RRHs into communities with frequent internal handover events. Based upon this, we connect each RRH community to a fog server. In the traffic-based BBU allocation phase, we first extract the traffic patterns of RRHs in each fog server, and model RRH traffic complementarity with regard to BBU capacities. We then formulate BBU

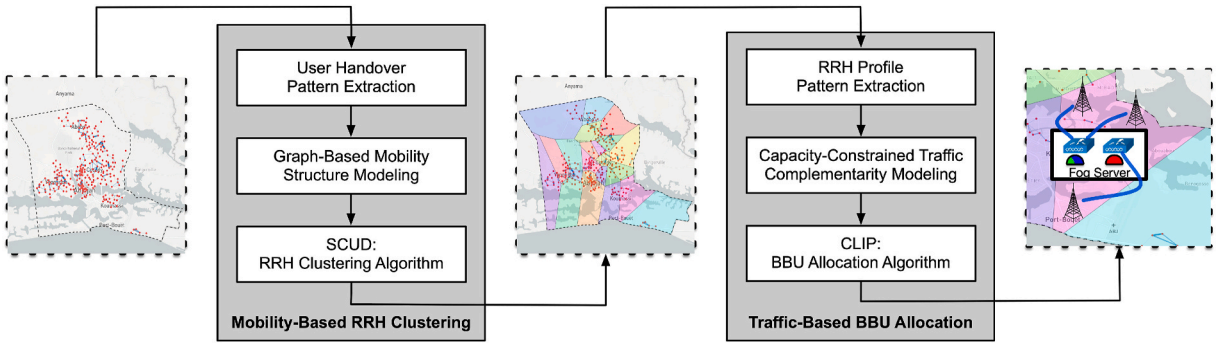



**Fig. 2.** Framework overview.

allocation as a *set partitioning problem*, and propose a column-reduced integer programming (CLIP) algorithm find optimal RRH partition schemes, so that each subset of RRHs share the BBU capacity to maximize the BBU utilization rate. In the following sections, we elaborate on the details of this framework.

## 4. Mobility-based RRH clustering

In this phase, our objective is to cluster neighboring RRHs with frequent handover events into communities. To this end, we first extract user mobility patterns across RRHs from handover counts, and then build a weighted graph to model user mobility structures with geographic constraints.

Specifically, we first model RRHs as graph nodes, and connect two RRHs with a link if they are geographically close to each other. We compute the link weight based on the handover intensity between the corresponding nodes. We then cluster RRHs based on the weighted graph. These communities should be densely connected internally and loosely connected among one another (Fortunato, 2010). In the literature, various algorithms have been proposed to find community structures in graphs (Fortunato, 2010), such as modularity maximization (Newman, 2004), label propagation (Raghavan et al., 2007), and the Girvan-Newman algorithm (Newman and Girvan, 2004). However, directly applying these community detection algorithms may not be adequate in the RRH clustering scenario, since we also need to constrain the geographic span of the formed RRH communities, so that the fronthaul latency between the RRHs and the fog server can be guaranteed to satisfy service quality requirements. As RRHs are not evenly distributed geographically, it is difficult to select a unified distance threshold for the RRH communities. Therefore, we proposed a size-constrained community detection (SCUD) algorithm to solve this problem. We elaborate the details as follows.

### 4.1. User handover pattern extraction

We extract the handover counts between RRHs in a mobile network based on the users' trajectories in the dataset. Specifically, we record a handover event when a user is observed in two consecutive RRHs. We then exploit a *tensor* structure to capture the spatial-temporal user mobility patterns (Kolda and Bader, 2009). Specifically, we build a tensor $\mathscr{H} \in \mathbb{R}^{N_r \times N_r \times N_t}$ with three dimensions to model the RRH handover counts, where $\mathscr{H}(r_i, r_j, t)$ refers to the handover count between RRH $r_i$ and RRH $r_j$ in the time slot $t$, $N_r$ is the number of RRHs, and $N_t$ is the number of time slots. We consider the case of symmetric handover counting where $\mathscr{H}(r_i, r_j, t) = \mathscr{H}(r_j, r_i, t)$, and a time slot of 1 h where $t = 1h$.

Based on the handover tensor, we first calculate the *average handover intensity* of each RRH pair as follows:

$$I(r_i, r_j) = \frac{1}{N_t} \sum_{t=1}^{N_t} \mathscr{H}(r_i, r_j, t) \tag{1}$$

As an example, Fig. 3(a) demonstrates the average handover intensities across a set of RRHs in Abidjan, Ivory Coast from 12/05/2011 to 04/22/2012. We can observe several RRH communities with strong internal handover intensities, which indicates the spatial locality of user handover patterns.

### 4.2. Graph-based mobility structure modeling

Based on the extracted user handover patterns, we model the structures of user mobility across RRHs as an undirected, weighted graph $G = (V, E)$, where $V = \{r_1, \ldots, r_N\}$ denotes the set of *nodes* corresponding to $N$ RRHs, and $E$ denotes the set of *links* between RRH pairs. We then define the adjacency matrix $W$ of graph $G$, which is an $N \times N$ symmetric matrix with entries $w(r_i, r_j)$ denoting the link weight between node $r_i$ and node $r_j$.

For each RRH node pair, we use their average handover intensity to determine their link weight, i.e., $w(r_i, r_j) = I(r_i, r_j)$. We consider the case of symmetric non-negative weights ($w(r_i, r_j) = w(r_j, r_i)$, $w(r_i, r_j) \geq 0$) with no loops ($w(r_i, r_i) = 0$). In this way, we model the user mobility patterns as a constructed weighted graph, which enables mobility-based RRH clustering in the next step.

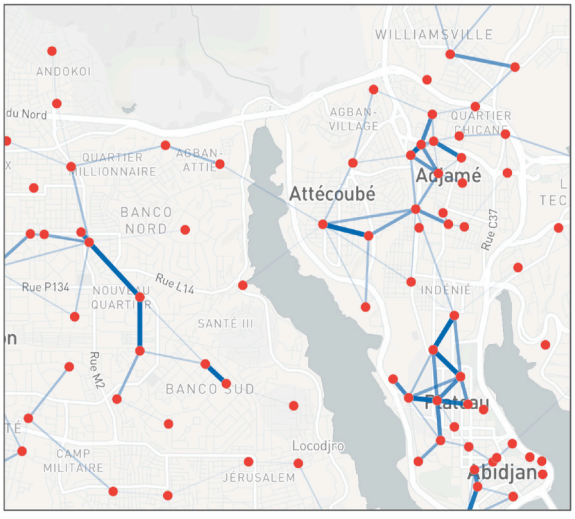



**Fig. 3.** An example of the daily user mobility profiles in Abidjan, Ivory Coast. The blue links on the map correspond to user handover counts between RRHs pairs, where thicker links correspond to larger handover counts.

### 4.3. SCUD: RRH clustering algorithm

Based upon the mobility graph structure, in this step, we cluster neighboring RRHs with frequent handover events across them into communities and connect them to the fog servers. We formulate this problem as a community detection problem (Fortunato, 2010), and elaborate the problem formulation and solution as follows.

**Problem Formulation**: Given graph $G = (V, E)$, we first define a set of communities $\mathscr{C} = \{C_1, \ldots, C_K\}$, where

$$\cup_{\forall C_k \in \mathscr{C}} = V \quad \text{and} \quad \cap_{\forall C_k \in \mathscr{C}} = \varnothing \tag{2}$$

Then, given a node $v$, we define the *connectivity* of $v$ to a community $C$ as the sum of link weights between $u$ and the nodes in the community $C$:

$$con(v, C) = \sum_{u \in C} w(u, v) \tag{3}$$

Finally, we define the *adjacent communities* $\mathscr{C}(v)$ of $v$ as

$$\mathscr{C}(v) = \{C | con(v, C) > 0, C \in \mathscr{C}\} \tag{4}$$

With the definitions above, our objective is to find an optimal set of communities $\mathscr{C}$, so that the internal connectivity within a community is higher than the inter-community connectivity, i.e.,

$$\forall v \in C_k,\ con(v, C_k) \geq max\{con(v, C_l), C_l \in \mathscr{C}\} \tag{5}$$

Moreover, we need to constrain the size of each community (i.e., the quantity of nodes in a community) to prevent long fronthaul latency, i.e.,

$$\forall C_k \in \mathscr{C},\ size(C_k) \leq \tau_{max} \tag{6}$$

**Solution and Challenge**: The problem can be identified as a community detection problem and be solved by *modularity maximization* algorithms (Newman and Girvan, 2004). The modularity $Q$ of a graph $G$ is defined as the difference of the probability of the edges that fall within a given community $C_k$ and the expected probability of edges that were distributed at random in the graph (Ostroumova Prokhorenkova et al., 2016), i.e.,

$$Q = \sum_{k=1}^{K} (e_{kk} - a_k^2) \tag{7}$$

where $e_{kk}$ is the probability of inner-community edges, i.e.,

$$e_{kk} = \frac{|E(C_k)|}{|E(G)|} = \frac{\sum_{u,v \in C_k} w(u, v)}{\sum_{u,v \in V} w(u, v)} \tag{8}$$

and $a_k$ is the probability of a random edge connecting to community $k$, i.e.,

$$a_k = \frac{\sum_{v \in C_k} deg(v)}{2|E|} = \frac{\sum_{u \in C_k, v \in V} w(u, v)}{\sum_{u,v \in V} w(u, v)} \tag{9}$$

$|E|$ is the sum of link weights, and $deg(v)$ denotes the degree of node $v$. Modularity reflects the concentration of edges within community compared with random distribution of links between all nodes regardless of community.

The basic idea of community detection by modularity maximization is that if we maximize $Q(C_k)$, the resultant community structure will have dense connections between the nodes within community but sparse connections between nodes in different community (Newman and Girvan, 2004). However, optimizing the modularity of a graph is proven to be NP-hard (Ostroumova Prokhorenkova et al., 2016). Moreover, the community size constraints bring extra challenges in modularity optimization. To address these issues, we propose a *size-constrained community detection (SCUD)* algorithm based on the popular fast-unfolding algorithm (Blondel et al., 2008) to effectively partition nodes into communities with size constraints.

**Size-Constrained Community Detection (SCUD)**: The basic idea of the SCUD algorithm is iteratively moving a node $v_j$ from its old community to a new community $C_k$ that maximize the *modularity gain* while not exceeding the size threshold $\tau_{max}$. Specifically, the modularity gain is calculated as follows

$$\Delta Q(v_j, C_k) = e_{jk} + e_{kj} - 2a_j a_k = 2(e_{jk} - a_j a_k) \tag{10}$$

The details of the SCUD algorithm are presented in Algorithm 1. Specifically, the algorithm iteratively build communities by merging adjacent nodes. At the first step of a iteration, we assign each node to its adjacent community with the highest modularity gain without exceeding the community size constraint.[1] In the second step of the iteration, we generate a new graph $G'$ by regarding each community in the original graph as a node. Specifically, for the nodes $v_i', v_j' \in G'$, we calculate their link weight as follow

$$w(v_i', v_j') = \sum_{u \in C_i, v \in C_j} w(u, v) \tag{11}$$

where $C_i \rightarrow v_i'$ and $C_j \rightarrow v_j'$. We repeat the two steps in each iteration until the new graph structure is the same as the previous one, or the maximum iteration number *max_iter* is reached, as the convergence of such a heuristic algorithm is difficult to prove.[2] Finally, we obtain a set of communities $\mathscr{C}$ for the RRHs with frequent internal mobility behaviors.

## 5. Traffic-based BBU allocation

In this phase, we need to assign a set of distributed fog servers to the RRH communities obtained in the previous phase, and determine the optimal quantity of BBUs allocated for each fog server. In real-world deployment, we assume that the fog server for each RRH community can be placed at the geographic centroid of the community. The RRHs in a community and the corresponding fog server are connected via high speed optical fibers (Tinini et al., 2017).

The basic idea of optimal BBU allocation in a fog server is to partition the connected RRHs into subsets, and allocate a BBU for each subset, so that the aggregated traffic in each subset are *complementary*, i.e., being close to the BBU capacity to a maximal extent while not exceeding the BBU capacity. For example, an RRH occupying 70% of BBU capacity can be partitioned in a subset with another RRH occupying 30% of capacity to increase BBU utilization. To this end, we first extract the RRH traffic patterns for each fog server, and then propose a *deviation-based* metric to measure their complementarity. Finally, we model the BBU allocation problem as a *set partitioning problem* (Balas and Padberg, 1976). To solve this problem, exhaustively searching for RRHs with complementary traffic patterns to form subsets can be computationally intractable, since there are a tremendous number of partitioning schemes as the network scale grows (Chen et al., 2018). Therefore, we propose a column-reduced integer programming (CLIP) algorithm (Diaby, 2010) to effectively find the exact solution to the optimal set partitioning problem. We elaborate the details as follows.

### 5.1. RRH traffic profile extraction

We extract the RRH traffic volume based on the communication traffic logs in the dataset.[3] We then build a tensor $\mathscr{T} \in \mathscr{R}^{N_r \times N_t}$ with two dimensions to model the RRH traffic volume, where $\mathscr{T}(r, t)$ refers to the traffic volume generated by RRH $r$ in the time slot $t$. We derive $\mathscr{T}$ by summing up the absolute values of inbound and outbound traffic, i.e.,

[1] If two communities yield the same gain, we randomly choose one.

[2] Based on experiments, we empirically find that the algorithm converges quickly in most cases.

[3] In this work, we calculate the total duration of calls as a measurement of traffic volume, while our approach can directly adapt to other traffic metrics.

$\mathscr{T}(r,t) = |\mathscr{T}_{in}(r,t)| + |\mathscr{T}_{out}(r,t)|$, and calculate the traffic volumes on an hourly basis, i.e., $t = 1h$.

Based on the traffic tensor, we extract a *traffic profile* for each RRH to characterize its traffic pattern. Specifically, for each RRH $r_i$, we aggregate and average its hourly traffic volume in the dataset over a *typical weekday* and a *typical weekend* to build the temporal profile, i.e.,

$$\Phi(r_i) = [fw_1, fw_2 \ldots, fw_{24}, fn_1, fn_2 \ldots, fn_{24}] \tag{12}$$

where $fw_i (i = 1, 2 \ldots 24)$ and $fn_i (i = 1, 2 \ldots 24)$ correspond to the average traffic volume of the $i$th hour over all weekdays and weekends, respectively.

As an example, Fig. 4 shows the daily traffic profiles of two RRHs in Abidjan, Ivory Coast from 12/05/2011 to 04/22/2012. We can see that the RRH traffic patterns in different areas (e.g., $r_1$ in a business district and $r_2$ in a residential area) exhibit different variations and intensities during the typical weekday and weekend.

### 5.2. Capacity-constrained traffic complementarity modeling

Based on the extracted traffic profiles, we define the traffic complementarity of a subset of RRHs connected to a BBU. Specifically, given a subset of RRHs $R = \{r_1, r_2, \ldots, r_n\}$, we first calculate their aggregated traffic profile as

$$\Phi(R) = \sum_{i=1}^{n} \Phi(r_i) \tag{13}$$

For example, the dashed lines in Fig. 5 demonstrate the aggregated traffic profiles for the two RRHs. We can see that during weekday morning and weekend afternoon (indicated by the masks in Fig. 5), the aggregated traffic volumes are very close to the BBU capacity (indicated by the green lines). Therefore, we can allocate one BBU for the two RRHs to share the BBU capacity and increase the BBU utilization rate in these periods.

More specifically, we first define the *temporal group* as a duration in the typical weekday and weekend, i.e., $[T_s, T_e] \in$ (Ericsson, 2019; Tran et al., 2017). We note that different temporal groups may lead to different traffic aggregation and BBU allocation schemes. We then compare the aggregated traffic volumes in the temporal group with the BBU capacity to determine their complementarity. Specifically, we define the *complementarity score* $\eta$ between the aggregated traffic $\Phi(R)$ and the BBU capacity $\Gamma$ as their coefficient of determination (Nagelkerkeet al., 1991) during the temporal group, i.e.,

$$\eta(R) = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}} = 1 - \frac{\sum_{t=T_s}^{T_e} (\Phi(R,t) - \Gamma)^2}{\sum_{t=T_s}^{T_e} (\Phi(R,t) - \overline{\Phi(R)})^2} \tag{14}$$

where

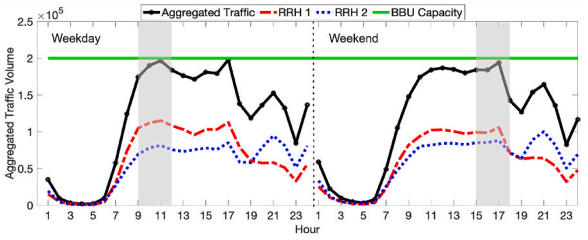



**Fig. 5.** An example of the aggregated traffic patterns of two RRHs in Abidjan, Ivory Coast. The blue masks indicate that the aggregated traffic volumes are very close to the BBU capacities (the green lines) during weekday morning and weekend afternoon.

$$\overline{\Phi(R)} = \frac{1}{|T_s - T_e|} \sum_{t=T_s}^{T_e} \Phi(R,t) \tag{15}$$

We note that the complementarity score $\eta$ is maximum (equals to 1) when the aggregated traffic and the BBU capacity are exactly the same during the temporal group. A lower $\eta$ corresponds to larger variations between the aggregated traffic and the BBU capacity.

Meanwhile, we define the *utilization rate* $\mu$ as the average percentage of the aggregated traffic to the BBU capacity during the temporal group, i.e.,

$$\mu(R) = \frac{1}{|T_s - T_e|} \sum_{t=T_s}^{T_e} \frac{\Phi(R,t)}{\Gamma} = \frac{\overline{\Phi(R)}}{\Gamma} \tag{16}$$

Normally, in order to avoid BBUs from overloading, we need to constrain $\mu(R) \leq 1$, so that the aggregated traffic does not exceed the BBU capacity.

### 5.3. CLIP: BBU allocation algorithm

Based on the definitions above, in this step, we partition a set of RRHs connected to a fog server into several subsets, so as to maximize the complementarity score of the formed subsets under the utilization constraint.

**Definitions**: Let $P^{(k)} = \{r_1^{(k)}, r_2^{(k)}, \ldots, r_{N_k}^{(k)}\}$ be the set of RRHs connected to a fog server $C^{(k)}$. We define a *partition* of $P^{(k)}$ as $\mathscr{P}^{(k)} = \{P_1^{(k)}, P_2^{(k)}, \ldots, P_J^{(k)}\}$, so that

$$\cup_{\forall P_j^{(k)} \in \mathscr{P}^{(k)}} = P^{(k)} \quad \text{and} \quad \cap_{\forall P_j^{(k)} \in \mathscr{P}^{(k)}} = \varnothing \tag{17}$$

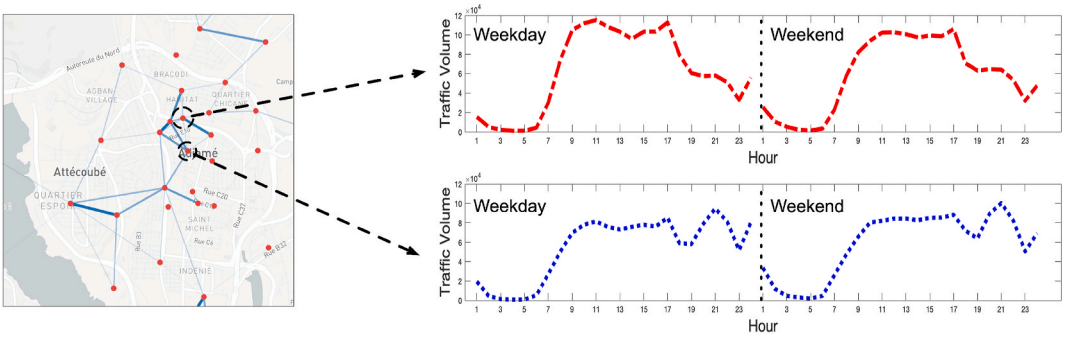



**Fig. 4.** An illustrative example of the daily traffic profiles of two RRHs in Abidjan, Ivory Coast. Red dots on the map correspond to RRHs, and curves on the charts correspond to hourly traffic volume in a typical weekday and weekend.

**Algorithm 1:** The SCUD algorithm

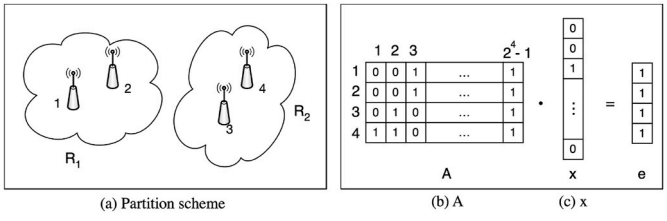
**Input:** Graph $G = (V, E)$, community size threshold $\tau_{max}$, maximum iteration number $max\_iter$
**Output:** Community label set $L$

Initialize: $L \leftarrow \{1, \ldots, N\}$
▷ randomize node list
$randomize(L)$;
**while** $(iter < max\_iter) \wedge (move > 0)$ **do**
  ▷ assign nodes to communities
  **for** $j \leftarrow 1$ **to** $N_r$ **do**
    **if** $size(v_j) \geq \tau_{max}$ **then**
      continue;
    **end**
    ▷ remove node $v_j$ from its community
    $old\_label \leftarrow L(v_j)$; $L(v_j) \leftarrow null$;
    $\mathbb{C}_{v_j} = get\_adjacent\_community(v_j, G, L)$;
    $max\_gain \leftarrow 0$;
    **for** $C_k \in \mathbb{C}_{v_j}$ **do**
      **if** $size(v_j) + size(C_k) > \tau_{max}$ **then**
        continue;
      **end**
      $gain \leftarrow \Delta Q(v_j, C_k)$;
      **if** $gain \geq max\_gain$ **then**
        $max\_gain \leftarrow gain$;
        $new\_label \leftarrow L(C_k)$;
      **end**
    **end**
    ▷ update node label
    $L(v_j) \leftarrow new\_label$
    **if** $old\_label \neq new\_label$ **then**
      $move \leftarrow 1$;
    **end**
  **end**
  ▷ update graph structure
  $\mathbb{C} = get\_all\_community(G, L)$;
  Initialize $G'(E', V')$;
  **for** $C \in \mathbb{C}$ **do**
    $v' \in V' \leftarrow C$;
    $\text{size}(v') \leftarrow \text{size}(C)$;
  **end**
  **for** $v'_i \in V'$ **do**
    **for** $v'_j \in V'$ **do**
      $w(v'_i, v'_j) \leftarrow \sum_{u \in C_i, v \in C_j} w(u, v)$
    **end**
  **end**
  $G \leftarrow G'$;
**end**
$\mathbb{C} = get\_all\_community(G, L)$;

In other words, $\mathscr{P}^{(k)}$ is a set of nonempty subsets of $P^{(k)}$, where every element in $R^{(k)}$ appears in only one of these subsets $P_j^{(k)}, j = 1,2,\ldots,J$. Let $\mathscr{R}^{(k)}$ be the power set of $P^{(k)}$, then $\mathscr{P}^{(k)}$ is a proper subset of $\mathscr{R}^{(k)}$, i.e., $\mathscr{P}^{(k)} \subset \mathscr{R}^{(k)}$. Fig. 6 shows an illustrative example of four RRHs partitioned into two subsets, where $P^{(k)} = \{r_1^{(k)}, r_2^{(k)}, r_3^{(k)}, r_4^{(k)}\}$, $P_1^{(k)} = \{r_1^{(k)}, r_2^{(k)}\}$, $P_2^{(k)} = \{r_3^{(k)}, r_4^{(k)}\}$, and $\mathscr{P}^{(k)} = \{P_1^{(k)}, P_2^{(k)}\}$ is a partition of $P^{(k)}$.

With the above-mentioned definitions, we present the formulation of the RRH set partitioning problem with the objective of maximizing the complementarity score under the utilization constraint.

**Problem**: (RRH Set Partitioning)




(a) Partition scheme (b) A (c) x

**Fig. 6.** An example of four RRHs partitioned into two subsets.

$$maximize \quad \eta(\mathscr{P}^{(k)}) \tag{18}$$

$$= maximize \quad \sum_{j=1}^{J} \eta(P_j) \tag{19}$$

subject to

$$\cup_{\forall P_j^{(k)} \in \mathscr{P}^{(k)}} = P^{(k)} \quad \text{and} \quad \cap_{\forall P_j^{(k)} \in \mathscr{P}^{(k)}} = \varnothing \tag{20}$$

$$\mu(\mathscr{P}^{(k)}) = \max \mu(P_j^{(k)}) \leq 1 \tag{21}$$

**Solution and Challenge**: The set partitioning problem (18) can be solved by integer programming algorithms (Diaby, 2010). First, we construct a (0,1)-matrix $A$ to describe all the possible subsets of $P^{(k)}$, where each column of $A$ represents a subset $P_j^{(k)} \in \mathscr{P}^{(k)}$, and each row of $A$ corresponds to an RRH $r_i^{(k)} \in P^{(k)}$. The binary element $A(i, j) = 1$ if and only if RRH $r_i^{(k)}$ is in subset $P_j^{(k)}$. For example, Fig. 6(b) shows the matrix representation of all the possible subsets in Fig. 6(a). We then associate a (0,1)-vector $\mathbf{x}$ with matrix $A$ to represent the set partitioning scheme $\mathscr{P}^{(k)}$. Specifically, we let $x_j = 1$ if and only if the $j$th column of $A$ is selected in the partitioning scheme, i.e., $P_j^{(k)} \in \mathscr{P}^{(k)}$. Since an RRH can be partitioned into one and only one subset, we derive

$$A\mathbf{x} = \mathbf{e} = (1, \ldots, 1)^T \tag{22}$$

For example, the partition scheme in Fig. 6(a) can be written as $\mathbf{x} = (0,1,1,0,0,0,0,0,0,0,0,0,0,0,0)^T$, corresponding to the selection of the two RRH subsets {0, 0, 1, 1} and {1, 1, 0, 0}. With the above-mentioned matrix representation, the objective function of Problem (18) can be rewritten as

$$maximize \quad \boldsymbol{\eta}^T \mathbf{x} \tag{23}$$

where $\boldsymbol{\eta}$ is the utilization rate vector for the corresponding subsets in $A$. Meanwhile, the utilization constraint can be expressed as

$$\|\boldsymbol{\mu}\mathbf{x}\|_\infty \leq 1 \tag{24}$$

where $\boldsymbol{\mu}$ is the complementarity vector for the corresponding subsets in $A$, and $\|\cdot\|_\infty$ is the vector infinity norm that computes the maximum of the vector elements.

Finally, the RRH set partitioning problem is rewritten as.

**Problem**: (Integer Programming Problem)

$$\underset{\mathbf{x}}{argmax} \quad \eta^T \mathbf{x} \tag{25}$$

subject to

$$A\mathbf{x} = \mathbf{e} = (1, \ldots, 1)^T \tag{26}$$

$$\|\mu\mathbf{x}\|_\infty \leq 1 \tag{27}$$

Since $\boldsymbol{\eta}$ and $\boldsymbol{\mu}$ are constant vectors for a given RRH set, we can compute their values in advance, and exhaustively search for the optimal $\mathbf{x}$ for () as a solution. However, as network scale grows, such an exhaustive search method quickly becomes intractable (Taleb et al., 2017). First, given a set of $m$ RRHs, the corresponding matrix $A$ contains $2^m - 1$ columns (subsets), making it difficult to store and manipulate for real-world networks with thousands and hundreds RRHs. Second, directly applying integer programming algorithms on such a large matrix $A$ is computationally intractable even with modern solvers (Taleb et al., 2017). Therefore, we propose a *column-reduced integer programming (CLIP)* algorithm to effectively solve Problem () as follows.

**Column-Reduced Integer Programming**: We reduce the number of columns in $A$ by exploiting a tree projection and pruning algorithm (Agarwal et al., 2001). Specifically, instead of enumerating all the RRH

subsets in the matrix, we generate a tree of RRH subsets by successively adding RRHs to the existing nodes. As an example, Fig. 7 shows the complete subset tree for the example in Fig. 6. Instead of generating all the tree nodes at once, we traverse the tree from top down in a depth-first manner, and prune branches based on the following Lemma.

**Lemma**. (*monotone property*) If the utilization rate $\mu(P^{(k)}) > 1$, then $\forall P^{(j)} \supset P^{(k)} \in \mathscr{P}$, we have $\mu(P^{(j)}) > 1$.

**Proof.** let $P^{(j)} = P^{(k)} \cup r_c$, we have

$$\begin{aligned}\mu(P^{(j)}) &= \frac{1}{|T_s - T_e|}\sum_{t=T_s}^{T_e}\frac{\Phi(P^{(j)},t)}{C} \\ &= \frac{1}{|T_s - T_e|}\sum_{t=T_s}^{T_e}\frac{\Phi(P^{(k)} \cup r_c,t)}{C} \\ &= \frac{1}{|T_s - T_e|}\sum_{t=T_s}^{T_e}\frac{\Phi(P^{(k)},t) + \Phi(r_c,t)}{C} > \mu(P^{(k)}) > 1\end{aligned} \tag{28}$$

Therefore, to satisfy utilization constraint (27), we can safely remove nodes with $\mu(P_k) > 1$ and all its child nodes. In this way, we generate the column-reduced matrix $A \in \mathbb{R}^{m\times n}$ with $n$ columns for $m$ RRHs, in which each column corresponds to a subset satisfying the utilization constraint. In practice, we find out that $n \ll 2^m - 1$, which effectively reduces the search space for the optimal solution.

Finally, we solve the integer programming problem () with column-reduced $A$. Such a problem is proven NP-hard (Karp, 1972), and various techniques have been proposed to solve it, such as cutting plane, branch and bound, and heuristic search (Nemhauser and Wolsey, 1988). The basic steps include narrowing the solution space, finding integer-feasible solutions, and discarding space without better integer-feasible solutions. In this work, we employ the Integer Linear Programming Solver from the Matlab Optimization Toolbox[4] to find the optimal solution.

## 6. Evaluation

We evaluate the performance of our framework based on real-world mobile network datasets. Specifically, we assess its capability of improving network quality and reducing network cost. We first describe the experiment settings, and then present the evaluation results and case studies.

### 6.1. Dataset description

We exploit two large-scale, anonymized CDR datasets released by Orange Group via the D4D challenges (Blondel et al., 2012; de Montjoye et al., 2014) for evaluation. The datasets contain CDRs from Orange customers from Ivory Coast for half-a-year, and Senegal in one year, respectively. After data preprocessing, we extract two city-scale datasets for *Abidjan* and *Dakar*, the two largest cities in Ivory Coast and Senegal, respectively. The details of the datasets are listed in Table 1.

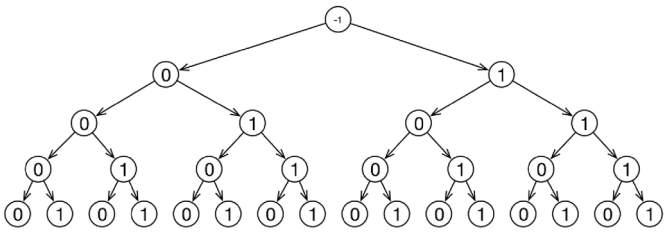



**Fig. 7.** A complete RRH subset tree for the example in Fig. 6.

In each city, we assume that a Fog-RAN is to be deployed upon the existing network infrastructure. Specifically, the existing base stations are replaced by light-weight RRHs. A set of distributed fog servers are deployed for each RRH community, and connected to the RRHs via high speed optical fibers. We calculate the RRH traffic volume by aggregating the incoming and outgoing call duration in each RRH, and derive the user handover count by traversing the trajectories of user mobility across RRHs. Due to privacy concerns, the user mobility data in the original datasets is randomly sampled from a portion of Orange customers (1% for Ivory Coast and 3.33% for Senegal, respectively) (Blondel et al., 2012; de Montjoye et al., 2014). Therefore, we estimate the actual handover count by multiplying the sampling rate.

### 6.2. Evaluation plan

Based on the two datasets, we first extract handover profiles for the RRH pairs in Abidjan and Dakar, respectively. Subsequently, we run the proposed SCUD algorithm to cluster RRHs into communities. For each community, we extract the RRH traffic profiles, and allocate BBUs for complementary RRHs using the proposed CLIP algorithm. We dynamically generate RRH-BBU mapping schemes for different temporal groups based on the observations from traffic variation patterns. We adjust the size of RRH community and BBU capacity to compare the performances of different schemes and find proper community sizes and BBU capacities for Abidjan and Dakar, respectively. The parameter selection procedures are detailed later.

### 6.3. Evaluation metrics

We derive the following network quality and cost metrics to evaluate the performances of different RAN architectures. First, compared with the traditional RAN architecture, the proposed Fog-RAN architecture reduces handover overhead and increases BBU utilization. We quantitatively evaluate the improvements using the following two metrics.

**Handover Overhead**: Given a set of RRHs and an RRH clustering scheme $\mathscr{C} = \{C_1, \ldots, C_K\}$, we calculate the handover overhead as the cost of migrating the user session data between RRHs for a random handover event. If two RRHs are connected to the same fog server, such migration cost can be negligible since no data copying is needed. For handover between different fog servers, we assume that each migration cost is constant. Consequently, we calculate the session migration delay as proportional to the handover event counts between fog servers, e.g.,

$$M_H(\mathscr{C}) = \sum_{k=1}^{K}\sum_{u\in C_k, v\notin C_k} w(u,v) \tag{29}$$

We note that for the traditional RAN architecture without RRH clustering, we have $M_H(\mathscr{C})$ is maximized as none of the RRHs are in the same community and every handover event are processed with a cost.

**BBU Utilization**: Given an RRH clustering scheme $\mathscr{C} = \{C_1,\ldots,C_K\}$, we assign a fog server to each community to process the aggregated RRH traffic. In each fog server, the BBU processing capacities are shared across the connected RRHs to increase the utilization rate. Specifically, given a set of RRHs $P^{(k)}$ in a fog server with the partitioning scheme

**Table 1**
Datasets description.

| City | Abidjan | Dakar |
|---|---|---|
| Area | 422 km$^2$ | 83 km$^2$ |
| Population | 4,707,404 | 1,146,053 |
| Base stations | 270 | 257 |
| Duration | 20 weeks<br>12/05/2011–04/22/2012 | 50 weeks<br>01/07/2013–12/22/2013 |
| Average call duration | 5.18 min | 6.82 min |
| Handover per hour | 78,662 | 113,082 |

[4] https://www.mathworks.com/help/optim/index.html.

$\mathscr{P}^{(k)} = \{P_1^{(k)}, P_2^{(k)}, \ldots, P_J^{(k)}\}$, we calculate its average BBU utilization rate as

$$M_U(\mathscr{P}^{(k)}) = \frac{1}{J}\sum_{j=1}^{J}\mu(P_j^{(k)}) \tag{30}$$

Upon this basis, we derive the *overall BBU utilization* for the entire Fog-RAN as

$$M_U(\mathscr{C}) = \frac{1}{K}\sum_{k=1}^{K}M_U(\mathscr{P}^{(k)}) \tag{31}$$

Second, compared with the Cloud-RAN architecture, the proposed Fog-RAN architecture reduces the fronthaul traffic volume and transmission latency between the RRHs and BBUs (Checko et al., 2016). We quantitatively evaluate the improvements with the following two metrics.

**Fronthaul Traffic**: In the fog and cloud-RAN architectures, BBUs are hosted in centralized servers, thus we benchmark the fronthaul traffic volumes between RRHs and the connected fog or cloud servers. Specifically, given an RRH clustering scheme $\mathscr{C} = \{C_1, \ldots, C_K\}$, we calculate the fronthaul traffic as the maximum traffic volume of the communities, i.e.,

$$M_\Phi(\mathscr{C}) = \max_{k=1,\ldots,K}\overline{\Phi(P^{(k)})}. \tag{32}$$

where $P^{(k)}$ is the set of RRHs in the fog sever corresponding to the community $C_k$, and $\overline{\Phi(P^{(k)})}$ is the average traffic volume of $P^{(k)}$ in a temporal group. We note that in the Cloud-RAN architecture, since all the RRHs are connected to a centralized cloud server, the fronthaul traffic volume equals to the sum of the RRH traffic volume.

**Fronthaul Latency**: Another key metric for evaluating a clustering scheme is the transmission delay between RRHs and BBUs in the network (Tinini et al., 2019). We assume that the fog or cloud servers are placed at the geographic centroids of the corresponding communities (Leskovec et al., 2014). Accordingly, we measure the average fronthaul delay as proportional to the radius of the fog or cloud, which is the maximum distance from the community centroid to the connected RRHs, i.e.,

$$M_D(\mathscr{C}) = \frac{1}{K}\sum_{k=1}^{K}\max_{r\in P^{(k)}} dist(r, \overline{C_k}) \tag{33}$$

where $\overline{C_k}$ is the geographic centroid of the community $C_k$, and $dist(r, \overline{C_k})$ is the Euclidean distance.

### 6.4. Baseline methods

Taking into consideration the traditional and state-of-the-art RAN architectures, we design the following baselines to compare to the proposed method.

**BTS-RAN**: this baseline directly connects each RRH to a BBU located at the same site. Each RRH-BBU pair is usually deployed and operated as a stand-alone base station (BTS) (Checko et al., 2015). In this way, no BBUs are shared across RRHs. This architecture has been widely adopted in many traditional networks, e.g., the 3G/4G mobile networks (Checko et al., 2015).

**Cloud-RAN**: this baseline adopts the RAN architecture proposed in (Chen et al., 2018), which deploys a centralized cloud server (BBU pool) for a city-wide network, and connects all the RRHs to the BBU pool via optical fibers. Similarly, in the cloud server, we partition RRHs into subsets and allocate BBUs for them to share the BBU capacity. However, it is computationally intractable to directly apply integer programming algorithms to find the exact optimal solution for such a city-wide cloud server. Instead, we adopt the *greedy* algorithm proposed in (Chen et al., 2018) to find an optimal approximation. Specifically, the algorithm incrementally allocates BBUs to accommodate RRH traffic demands in heuristic iterations until all the RRHs are connected.

**Simple-Fog-RAN**: this baseline clusters RRHs into communities based on their *geographic* distances without considering user mobility patterns in the network. The algorithm and constraints are the same as the proposed CLIP method. In each fog server, it performs RRH partitioning and BBU allocation using the *greedy* algorithm as proposed in (Chen et al., 2018).

**Fog-RAN**: the proposed Fog-RAN architecture clusters RRHs into communities based on handover events leveraging the proposed algorithm (CLIP), and allocates BBUs in fog servers using the exact optimization algorithm (SCUD).

### 6.5. Parameter selection

The following key parameters in the proposed framework need to be carefully selected to achieve optimal performance.

**Temporal groups**: In the RRH partitioning and BBU allocation phase, we need to dynamically switch to different partitioning schemes in different temporal groups. Fig. 8 shows the traffic variation patterns of Abidjan and Dakar, respectively. Based on the observations, we derive six temporal groups in weekdays and weekends, as shown in Table 2.

**RRH Community Size**: In the RRH clustering phase, a key parameter is the community size threshold $\tau_{max}$ (i.e., the largest quantity of RRHs in each community). A small threshold may result in fragmented communities, high handover overhead $M_H(\mathscr{C})$ and large number of communities $K$, while a large threshold may lead to over-sized communities with high fronthaul traffic $M_\Phi(\mathscr{C})$. Based on cell planing practices and fog network surveys (Amzallag et al., 2005; Yousefpour et al., 2019), we vary the threshold $\tau_{max}$ from 2 to 20 RRHs, and calculate the cost to compare different size thresholds as follows

$$Cost(\mathscr{C}|\tau_{max}) = M_H(\mathscr{C})^* M_\Phi(\mathscr{C})^* K \tag{34}$$

To minimize $Cost(\mathscr{C}|\tau_{max})$, we conduct repeated experiments over groups with different community size thresholds in both cities, and present the results in Fig. 9.

Furthermore, for cities without handover count data, we can estimate the handover overhead between two RRHs using their Euclidean distance. Based on the observations, closer RRHs usually have more handover counts, and the handover count between two RRHs is about the inversely proportional to the Euclidean distance between them (see Fig. 10). Therefore, we estimate the handover overhead and the cost as follows

$$\widehat{w}(u,v) = \frac{1}{dist(u,v)}, \quad \widehat{M_H}(\mathscr{C}) = \sum_{k=1}^{K}\sum_{u\in C_k, v\notin C_k}\widehat{w}(u,v) \tag{35}$$

$$\widehat{Cost}(\mathscr{C}|\tau_{max}) = \widehat{M_H}(\mathscr{C})^* M_\Phi(\mathscr{C})^* K \tag{36}$$

Fig. 11 shows the results of repeated experiments over groups with different community size thresholds using estimated handover counts, and the optimal values of $\tau_{max}$ for Abidjan and Dakar are the same as those determined using real-world data.

Also, we conducted a series of experiments over historical data with different time windows to verify the effectiveness of using partial data to calculate the $\tau_{max}$ and show the moderate consistency of real-world user mobility and traffic data. First, we use the data in the first week to calculate $\tau_{max}$. Then, we extend the time window to the first two weeks and get the corresponding $\tau_{max}$. The process is repeated until the $\tau_{max}$ remains the same for more than two rounds or all data have been included. The results are shown in Fig. 12, indicating that the $\tau_{max}$ can be estimated with around three and two weeks of data in Abidjan and Dakar, respectively. Actually, the mobility and traffic patterns in a city show moderate regularity (see Appendix A), making it practical to estimate $\tau_{max}$ using limited historical data. Furthermore, we change the $\tau_{max}$ from 10 to 15 in Abidjan and from 15 to 20 in Dakar and calculate

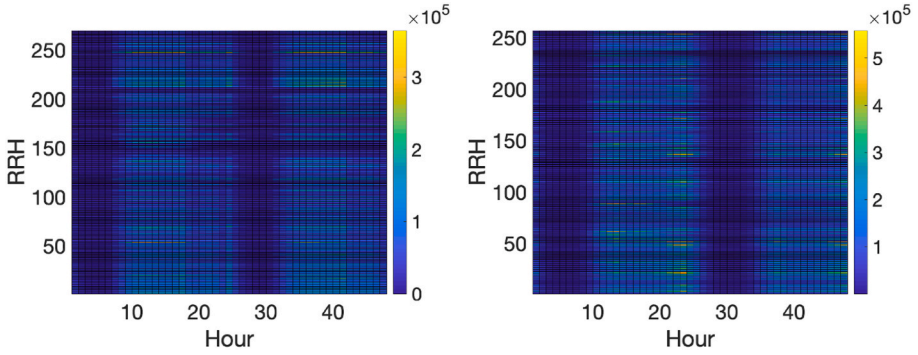



**Fig. 8.** Traffic variation patterns of the 270 and 257 RRHs in Abidjan (left) and Dakar (right), respectively. Each row corresponds to a traffic profile of an RRH denoted by the typical weekday and weekend.

**Table 2**
Temporal groups for dynamic scheme switching.

| Day type | Group name | Time span |
|---|---|---|
| Weekdays | working hours | 08:00–17:00 |
| | evening time | 17:00–22:00 |
| | night time | 22:00–08:00 |
| Weekends | day time | 10:00–19:00 |
| | evening time | 19:00–02:00 |
| | night time | 02:00–10:00 |

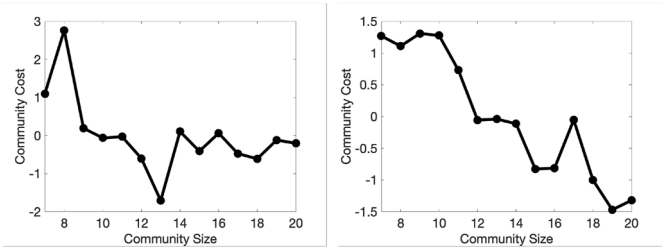



**Fig. 9.** The costs of forming different sizes of communities in Abidjan (left) and Dakar (right), respectively.

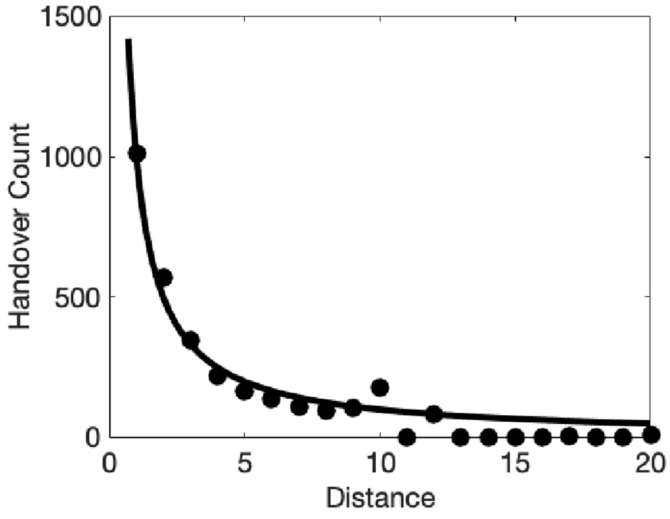



**Fig. 10.** The handover count between two RRHs is about the inversely proportional to their Euclidean distance.

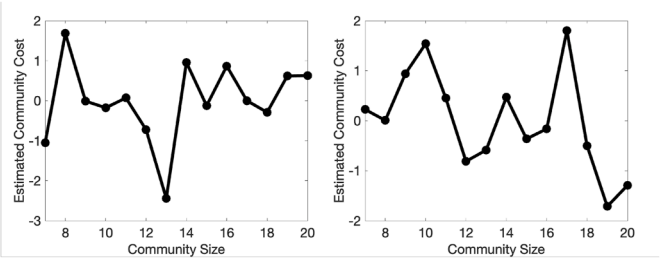



**Fig. 11.** The estimated costs of forming different sizes of communities in Abidjan (left) and Dakar (right), respectively.

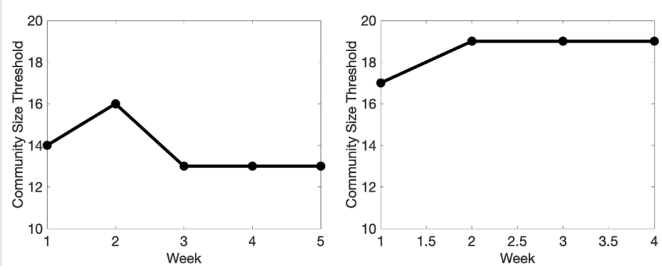



**Fig. 12.** The optimal $\tau_{max}$ estimated using historical data with different time windows. The $n$ in x-axis means using the first $n$ weeks of the data.

the corresponding values of $\tau_{max}$, the results (see Appendix B) show that the proposed approach can achieve a good performance so long as the $\tau_{max}$ is controlled in a reasonable scope. Based on the above discussions, we select optimal $\tau_{max} = 13$ for Abidjan and $\tau_{max} = 19$ for Dakar.

**BBU capacity**: Based on the observations and repeated experiments, we select the BBU capacity $\Gamma = 2 \times 10^5$ for Abidjan and $\Gamma = 3 \times 10^5$ for Dakar.

### 6.6. Evaluation results

#### 6.6.1. Results of RRH clustering

**Handover Overhead**: Fig. 13 shows the handover overhead using different RRH clustering methods on the two datasets. The traditional *BTS-RAN* baseline obtains the highest handover overhead in both cities (normalized to 100%), since each user handover event is processed between different BBUs. The *Simple-Fog-RAN* baseline shows moderate improvements on handover overhead in both cities, due to the adoption of distributed fog servers. Finally, the proposed *Fog-RAN* method achieves the lowest handover overhead (12.8% and 27.3%, respectively), validating the effectiveness of exploiting user mobility community structure in reducing handover overhead.

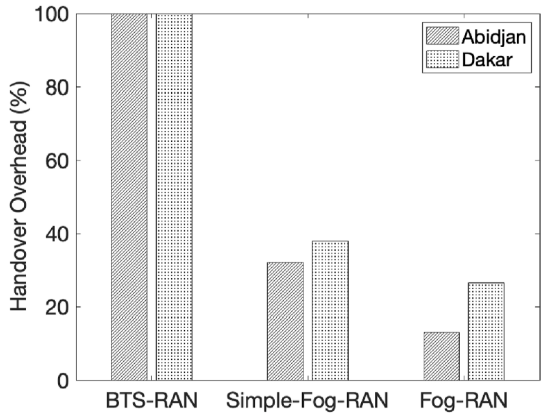



**Fig. 13.** The handover overhead comparison of BTS-RAN, Simple-Fog-RAN and the proposed Fog-RAN in Abidjan and Dakar.

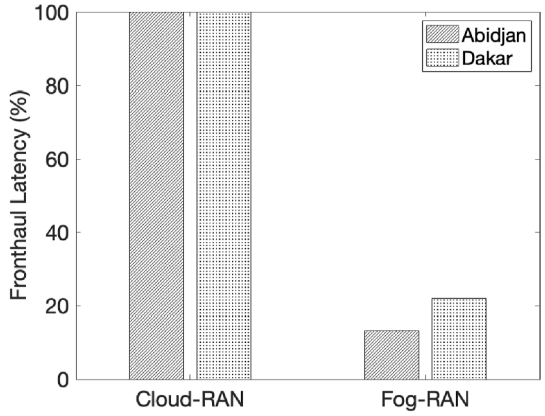



**Fig. 14.** The fronthaul latency comparison of Cloud-RAN and the proposed Fog-RAN in Abidjan and Dakar.

**Fronthaul Latency**: Fig. 14 shows the fronthaul latency of the *Cloud-RAN* baseline and the proposed *Fog-RAN* method. The *Cloud-RAN* baseline shows the worse performance in both cities, since the centralized BBU pool needs a large community radius to cover all the RRHs. In comparison, the proposed *Fog-RAN* method achieves significant improvements on fronthaul latency due to the distributed, size-constrained fog servers in the Fog-RAN architecture.

### 6.6.2. Results of BBU allocation

**BBU Utilization**: In Fig. 15, we present the BBU utilization rate of different methods on the two datasets. We can see that the proposed *Fog-RAN* method achieves the best BBU utilization rate in both cities in all the temporal groups using the SCUD algorithm (with an average BBU utilization rate of 49.7% in Abidjan and 52.3% in Dakar, respectively). In comparison, the traditional *BTS-RAN* method performs the worst since there is no sharing of BBU capacities across the base stations. The *Simple-Fog-RAN* methods achieves improved BBU utilization, due to the adoption of BBU sharing in the fog servers with the greedy allocation algorithms. However, since the greedy algorithms do not always guarantee the optimal results, the overall performance is not as good as the *Fog-RAN* method.

**Fronthaul Traffic**: Fig. 16 shows the fronthaul traffic in different temporal groups using *Cloud-RAN* and proposed *Fog-RAN* methods on the two datasets. The *Cloud-RAN* baseline shows the worse performance in both cities. In comparison, the proposed *Fog-RAN* method achieves significant improvements on reducing fronthaul traffic due to the distributed, size-constrained fog servers in the Fog-RAN architecture.

## 6.7. Case studies

### 6.7.1. Abidjan

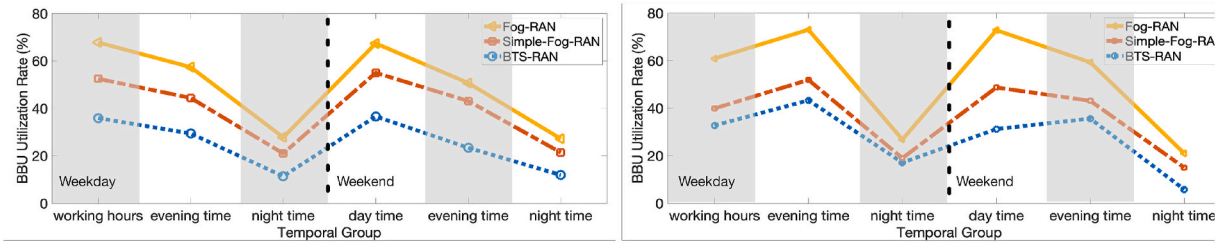
We visualize the clustering results of the proposed method in Fig. 17 (a) using a Voronoi diagram (Aurenhammer and Edelsbrunner, 1984), where each polygon corresponds to an RRH community. We also draw the user mobility patterns by lines, where thicker lines correspond to more handover events between the corresponding RRH pairs. We can see that the handover events are frequently observed across the business districts (e.g., Plateau), as shown in Fig. 17(b). Our method successfully finds RRH communities with frequent internal handover events and thus reduces the user handover overhead. Fig. 17(c) shows the BBU allocation scheme in Plateau, Abidjan during working hours (8:00–17:00) in weekdays, and the aggregated traffic pattern in one of its BBUs. We can see that in this BBU, the aggregated traffic pattern in working hours (08:00–17:00 in weekdays) is close to the BBU capacity and thus improves the overall BBU utilization.

### 6.7.2. Dakar

Fig. 18(a) shows the RRH community structure and mobility patterns in Dakar. As the capital of Senegal, Dakar features various administrative and business areas, as well as populated residential neighborhoods. In particular, our method identifies the *Dakar-Plateau* arrondissement (borough), as shown in Fig. 18(b), where most ministries and public administrations are located. Fig. 18(c) shows the BBU allocation scheme in Dakar-Plateau during evening time (19:00–02:00) in weekends, and the aggregated traffic pattern in one of its BBUs. We can see that the traffic tends of RRH 1 and RRH 2 during this temporal group are complementary to each other. Therefore, aggregating these two complementary RRHs to allocate a BBU can significantly increase BBU utilization.

## 7. Conclusion

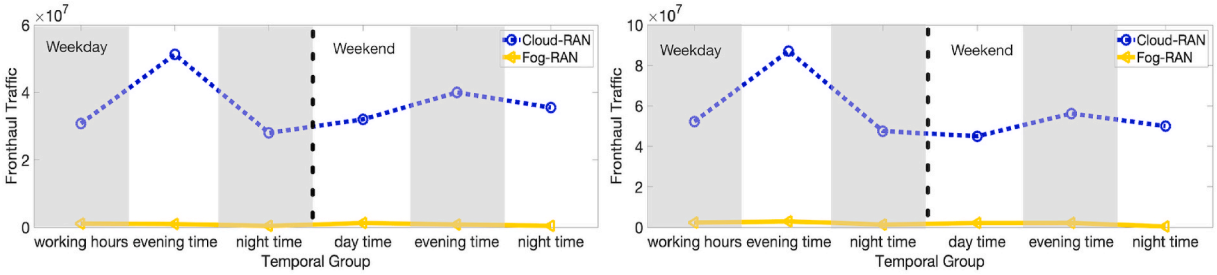
In this work, we propose a data-driven optimization framework for the Fog-RAN architecture. We focus on two of the most important objectives in Fog-RAN optimization, i.e., increasing infrastructure utilization and improving handover quality. Accordingly, we propose a two-phase framework to map RRHs to BBUs hosted in distributed fog servers. Specifically, we first exploit user mobility patterns to cluster




(a) The BBU utilization rates in different temporal groups in Abidjan.

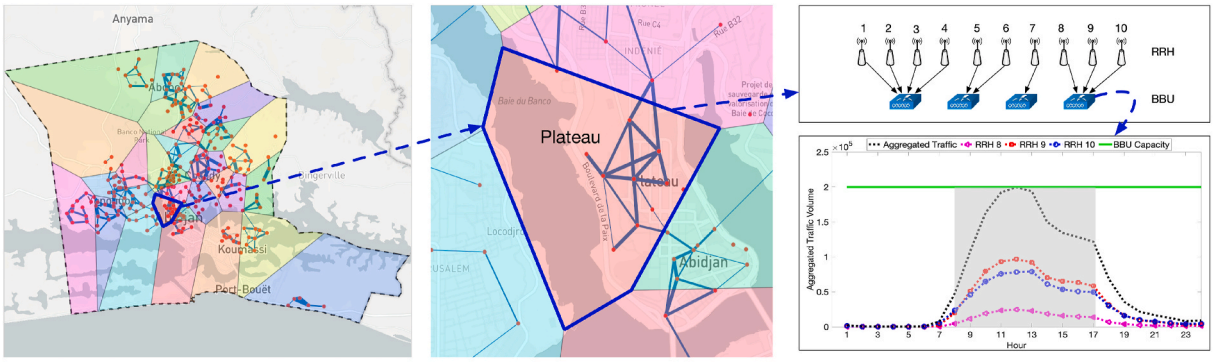
(b) The BBU utilization rates in different temporal groups in Dakar.

**Fig. 15.** The BBU utilization rate comparison of BTS-RAN, Simple-Fog-RAN, and the proposed Fog-RAN.



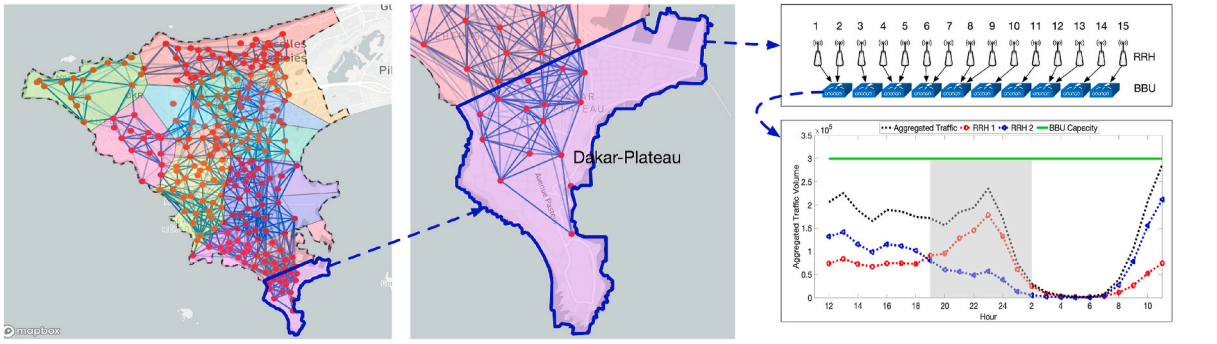

(a) The fronthaul traffic in different temporal groups in Abidjan. (a) The fronthaul traffic in different temporal groups in Dakar.

**Fig. 16.** The fronthaul traffic comparison of Cloud-RAN and the proposed Fog-RAN.




(a) RRH clustering results in Abidjan. (b) The community of Plateau. (c) BBU allocation scheme in Plateau community.

**Fig. 17.** The RRH clustering results in Abidjan during working hours (8:00–17:00) in weekdays.




(a) RRH clustering results in Dakar. (b) The community of Dakar-Plateau. (c) BBU allocation scheme in Dakar-Plateau community.

**Fig. 18.** The RRH clustering results in Dakar during evening time (19:00–02:00) in weekends.

RRHs into communities, and then assign a fog server to each RRH community. In each fog server, we partition the connected RRHs into subsets and allocate BBUs based on the RRH traffic demands. The proposed size-constrained communication detection (SCUD) algorithm is capable of finding RRH communities with intensive internal mobility patterns, and the proposed column-reduced integer programming (CLIP) algorithm is effective in partitioning RRHs into subsets with complementary traffic patterns. Real-world evaluation results in Abidjan and Dakar show that compared with the traditional RAN architecture, our framework effectively reduces the average handover overhead to 12.8% and 27.3%, and increases the average BBU utilization rate to 49.7% and 52.3% in both cities, respectively, which consistently outperforms the state-of-the-art baseline methods.

In the future, we plan to improve this work in the following directions. First, we plan to explore the dynamic mapping schemes between RRHs and fog servers in packet routing RAN networks, to support the real-time optimization of Fog-RAN. Second, we plan to investigate the variations in the BBU pool, such as considering different BBU capacity levels, and various resource constraints in the fog servers. We believe that such a data-driven optimization paradigm can benefit the design and deployment of the Fog-RAN architecture in the 5G era.

## Credit author statement

**Longbiao Chen**: Conceptualization, Methodology, Writing - Original Draft, Project administration, Funding acquisition. **Zhihan Jiang**: Formal analysis, Investigation, Writing- Reviewing and Editing, Methodology, Software, Visualization, Data Curation. **Dingqi Yang**: Validation, Writing- Reviewing and Editing. **Cheng Wang**: Writing- Reviewing and Editing, Resources, Supervision, Funding acquisition. **Thi-Mai-Trang Nguyen**: Writing- Reviewing and Editing, Conceptualization, Resources.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

We would like to thank the reviewers and editors for their constructive suggestions. This research is supported by NSF of China No. 61802325 and No. 61872306.

## Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jnca.2021.103083.

## References

Agarwal, R.C., Aggarwal, C.C., Prasad, V.V.V., 2001. A tree projection algorithm for generation of frequent item sets. J. Parallel Distr. Comput. 61 (3), 350–371.
Akyildiz, I.F., Lin, Yi-Bing, Lai, Wei-Ru, Chen, Rong-Jaye, 2000. A new random walk model for PCS networks. IEEE J. Sel. Area. Commun. 18 (7), 1254–1260.
Amzallag, D., Livschitz, M., Naor, J., Raz, D., 2005. Cell planning of 4G cellular networks: algorithmic techniques and results. In: 2005 6th IEE International Conference on 3G and beyond, pp. 1–5.
Andrews, J.G., Buzzi, S., Choi, W., Hanly, S.V., Lozano, A., Soong, A.C.K., Zhang, J.C., 2014. What will 5G Be? IEEE J. Sel. Area. Commun. 32 (6), 1065–1082.
Aurenhammer, F., Edelsbrunner, H., 1984. An optimal algorithm for constructing the weighted voronoi diagram in the plane. Pattern Recogn. 17 (2), 251–257.
Balas, E., Padberg, M., 1976. Set partitioning: a survey. SIAM Rev. 18 (4), 710–760.
Barlacchi, G., Nadai, M.D., Larcher, R., Casella, A., Chitic, C., Torrisi, G., Antonelli, F., Vespignani, A., Pentland, A., Lepri, B., 2015. A multi-source dataset of urban life in the city of Milan and the Province of Trentino. Scientific Data 2, 150055.
Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E., 2008. Fast unfolding of communities in large networks. J. Stat. Mech. Theor. Exp. 2008 (10), P10008.
Blondel, V.D., Esch, M., Chan, C., Clerot, F., Deville, P., Huens, E., Morlot, F., Smoreda, Z., Ziemlicki, C., 2012. Data for Development: the D4D Challenge on Mobile Phone Data arXiv:1210.0137.
Bonomi, F., Milito, R., Zhu, J., Addepalli, S., 2012. Fog computing and its role in the internet of things. In: Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing. ACM, New York, NY, USA, pp. 13–16.
C. M. R. Institute, 2011. C-RAN: the Road toward Green RAN. Tech. rep.. China Mobile Research Institute, Beijing, China.
Checko, A., Christiansen, H.L., Yan, Y., Scolari, L., Kardaras, G., Berger, M.S., Dittmann, L., 2015. Cloud RAN for mobile networks - a technology overview. IEEE Communications Surveys Tutorials 17 (1), 405–426.
Checko, A., Avramova, A.P., Berger, M.S., Christiansen, H.L., 2016. Evaluating C-RAN fronthaul functional splits in terms of network level energy and cost savings. J. Commun. Network. 18 (2), 162–172.
Chen, C., Zhang, D., Li, N., Zhou, Z.-H., 2014. B-planner: planning bidirectional night bus routes using large-scale taxi GPS traces. IEEE Trans. Intell. Transport. Syst. 15 (4), 1451–1465.
Chen, L., Zhang, D., Ma, X., Wang, L., Li, S., Wu, Z., Pan, G., 2015. Container port performance measurement and comparison leveraging ship gps traces and maritime open data. IEEE Trans. Intell. Transport. Syst. 17 (5), 1227–1242.
Chen, L., Yang, D., Zhang, D., Wang, C., Li, J., Nguyen, T.-M.-T., 2018. Deep mobile traffic forecast and complementary base station clustering for C-RAN optimization. J. Netw. Comput. Appl. 121, 59–69 ([CCF-C, JCR-II]).
Cisco, 2015. Fog Computing and the Internet of Things: Extend the Cloud to where the Things Are. Tech. rep., USA.
Cisco, 2016. Global Mobile Data Traffic Forecast Update, 2016–2021. Tech. rep., Cisco, San Jose, CA, USA.
Dao, N.-N., Lee, J., Vu, D.-N., Paek, J., Kim, J., Cho, S., Chung, K.-S., Keum, C., 2017. Adaptive resource balancing for serviceability maximization in fog radio access networks. IEEE Access 5, 14548–14559.
de Montjoye, Y.-A., Smoreda, Z., Trinquart, R., Ziemlicki, C., Blondel, V.D., 2014. D4D-Senegal: the Second Mobile Phone Data for Development Challenge arXiv:1407.4885 [physics].
Demestichas, P., Georgakopoulos, A., Karvounas, D., Tsagkaris, K., Stavroulaki, V., Lu, J., Xiong, C., Yao, J., 2013. 5G on the horizon: key challenges for the radio-access network. IEEE Veh. Technol. Mag. 8 (3), 47–53.
Diaby, M., 2010. Linear programming formulation of the set partitioning problem. Int. J. Operational Research Int. J. Operational Research 8, 399–427.
Dinh, T.H.L., Kaneko, M., Boukhatem, L., 2019. Energy-efficient user association and beamforming for 5g fog radio access networks. In: 2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC). IEEE, pp. 1–6.
Ericsson, 2019. Ericsson Mobility Report 2019. Tech. rep., Stockholm, Sweden.
Fortunato, S., 2010. Community detection in graphs. Phys. Rep. 486 (3), 75–174.
Furno, A., Naboulsi, D., Stanica, R., Fiore, M., 2016. Mobile demand profiling for cellular cognitive networking. IEEE Trans. Mobile Comput. (99), 772–786.
Gandotra, P., Jha, R.K., 2017. A survey on green communication and security challenges in 5G wireless communication networks. J. Netw. Comput. Appl. 96, 39–61.
Gao, X., Huang, X., Tang, Y., Shao, Z., Yang, Y., 2020. Data-Driven Bandit Learning for Proactive Cache Placement in Fog-Assisted IoT Systems arXiv:2008.00196 [cs, eess] ArXiv: 2008.00196.
Giust, F., Verin, G., Antevski, K., Chou, J., Fang, Y., Featherstone, W., Fontes, F., Frydman, D., Li, A., Manzalini, A., et al., 2018. Mec deployments in 4g and evolution towards 5g. ETSI White Paper 24, 1–24.
Guo, B., Wang, Z., Yu, Z., Wang, Y., Yen, N.Y., Huang, R., Zhou, X., 2015. Mobile crowd sensing and computing: the review of an emerging human-powered sensing paradigm. ACM Comput. Surv. 48 (1), 1–31.
Hasan, Z., Boostanimehr, H., Bhargava, V.K., 2011. Green cellular networks: a survey, some research issues and challenges. IEEE Communications Surveys Tutorials 13 (4), 524–540.
I, C.L., Rowell, C., Han, S., Xu, Z., Li, G., Pan, Z., 2014. Toward green and soft: a 5G perspective. IEEE Commun. Mag. 52 (2), 66–73.
J. Research, 2011. Mobile Operator Business Models: Challenges, Opportunities & Adaptive Strategies 2011-2016. Tech. rep.. Juniper Research, New York.
Karp, R.M., 1972. Reducibility Among Combinatorial Problems. Springer US, Boston, MA, pp. 85–103.
Klonoff, D.C., 2017. Fog Computing and Edge Computing Architectures for Processing Data from Diabetes Devices Connected to the Medical Internet of Things.
Kolda, T., Bader, B., 2009. Tensor decompositions and applications. SIAM Rev. 51 (3), 455–500.
Kumar, P., Zaidi, N., Choudhur, T., 2016. Fog Computing: Common Security Issues and Proposed Countermeasures, vol. 5.
Leskovec, J., Rajaraman, A., Ullman, J.D., 2014. Mining of Massive Datasets. Cambridge University Press.
Li, T., Magurawalage, C.S., Wang, K., Xu, K., Yang, K., Wang, H., 2017. On efficient offloading control in cloud radio access network with mobile edge computing. In: 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE, Atlanta, GA, USA, pp. 2258–2263. https://doi.org/10.1109/ICDCS.2017.24.
Lin, Y., Shao, L., Zhu, Z., Wang, Q., Sabhikhi, R.K., 2010. Wireless network cloud: architecture and system requirements. IBM J. Res. Dev. 54 (1), 4:1–4:12.
Liu, Liang, Yang, Feng, Wang, Richard, Shi, Zhenning, Stidwell, A., Gu, Daqing, 2012. Analysis of handover performance improvement in cloud-RAN architecture. In: 7th International Conference on Communications and Networking in China, pp. 850–855.
Lu, K., Rong, B., Kota, S.L., Liu, G., Wang, X., 2013. Next generation cognitive cellular networks: spectrum sharing and trading [Guest editorial]. IEEE Wireless Communications 20 (2), 10–11.
Luo, H., Zhao, H., Yin, S., 2018. Data-driven design of fog-computing-aided process monitoring system for large-scale industrial processes. IEEE Transactions on Industrial Informatics 14 (10), 4631–4641.
Mandlekar, V.G., Mahale, V., Sancheti, S.S., Rais, M.S., 2014. Survey on fog computing mitigating data theft attacks in cloud. Int. J. Innov. Res. Comput. Sci. Technol 2 (6), 13–16.
Mao, Y., You, C., Zhang, J., Huang, K., Letaief, K.B., 2017. A survey on mobile edge computing: the communication perspective. IEEE Communications Surveys Tutorials 19 (4), 2322–2358. Fourthquarter.
Munaretto, A., Fonseca, M., 2007. Routing and quality of service support for mobile ad hoc networks. Comput. Network. 51 (11), 3142–3156.
Nagelkerke, N.J., et al., 1991. A note on a general definition of the coefficient of determination. Biometrika 78 (3), 691–692.
Nemhauser, G.L., Wolsey, L.A., 1988. Integer Programming and Combinatorial Optimization, vol. 20.
Newman, M.E.J., 2004. Fast algorithm for detecting community structure in networks. Phys. Rev. 69 (6), 066133.
Newman, M.E.J., Girvan, M., 2004. Finding and evaluating community structure in networks. Phys. Rev. 69 (2), 026113.
Ni, J., Zhang, K., Lin, X., Shen, X.S., 2017. Securing fog computing for internet of things applications: challenges and solutions. IEEE Communications Surveys & Tutorials 20 (1), 601–628.
Ostroumova Prokhorenkova, L., Prałat, P., Raigorodskii, A., 2016. Modularity of complex networks models. In: Algorithms and Models for the Web Graph. Springer International Publishing, pp. 115–126.
Pang, A.-C., Chung, W.-H., Chiu, T.-C., Zhang, J., 2017. Latency-driven cooperative task computing in multi-user fog-radio access networks. In: 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE, pp. 615–624.
Park, S.-H., Simeone, O., Shamai Shitz, S., 2016. Joint optimization of cloud and edge processing for fog radio access networks. IEEE Trans. Wireless Commun. 15 (11), 7621–7632.
Peng, M., Wang, C., Lau, V., Poor, H.V., 2015. Fronthaul-constrained cloud radio access networks: insights and challenges. IEEE Wireless Communications 22 (2), 152–160.
Peng, M., Yan, S., Zhang, K., Wang, C., 2016. Fog-computing-based radio access networks: issues and challenges. IEEE Network 30 (4), 46–53.
Porambage, P., Okwuibe, J., Liyanage, M., Ylianttila, M., Taleb, T., 2018. Survey on multi-access edge computing for internet of things realization. IEEE Communications Surveys & Tutorials 20 (4), 2961–2991.
Raghavan, U.N., Albert, R., Kumara, S., 2007. Near linear time algorithm to detect community structures in large-scale networks. Phys. Rev. 76 (3), 036106.
Saharan, K., Kumar, A., 2015. Fog in comparison to cloud: a survey. Int. J. Comput. Appl. 122 (3).

Santoyo-González, A., Cervelló-Pastor, C., 2018. Latency-aware cost optimization of the service infrastructure placement in 5G networks. J. Netw. Comput. Appl. 114, 29–37.
Sengupta, A., Tandon, R., Simeone, O., 2017. Fog-aided wireless networks for content delivery: fundamental latency tradeoffs. IEEE Trans. Inf. Theor. 63 (10), 6650–6678.
Shi, W., Cao, J., Zhang, Q., Li, Y., Xu, L., 2016. Edge computing: vision and challenges. IEEE internet of things journal 3 (5), 637–646.
Shih, Y.-Y., Chung, W.-H., Pang, A.-C., Chiu, T.-C., Wei, H.-Y., 2017. Enabling low-latency applications in fog-radio access networks. IEEE Network 31 (1), 52–58.
Sigwele, T., Hu, Y.F., Ali, M., Hou, J., Susanto, M., Fitriawan, H., 2018. Intelligent and energy efficient mobile smartphone gateway for healthcare smart devices based on 5g. In: 2018 IEEE Global Communications Conference (GLOBECOM). IEEE, pp. 1–7.
Taleb, H., Helou, M.E., Khawam, K., Lahoud, S., Martin, S., 2017. Centralized and distributed RRH clustering in cloud radio access networks. In: 2017 IEEE Symposium on Computers and Communications (ISCC'17), pp. 1091–1097.
Tan, M., Wang, B., Wu, Z., Wang, J., Pan, G., 2016. Weakly supervised metric learning for traffic sign recognition in a LIDAR-equipped vehicle. IEEE Trans. Intell. Transport. Syst. 17 (5), 1415–1427.
Tandon, R., Simeone, O., 2016. Harnessing cloud and edge synergies: toward an information theory of fog radio access networks. IEEE Commun. Mag. 54 (8), 44–50.
Tian, F., Zhang, P., Yan, Z., 2017. A survey on C-RAN security. IEEE Access 5, 13372–13386.
Tinini, R.I., Reis, L.C.M., Batista, D.M., Figueiredo, G.B., Tornatore, M., Mukherjee, B., 2017. Optimal placement of virtualized BBU processing in hybrid cloud-fog RAN over TWDM-PON. In: GLOBECOM 2017 - 2017 IEEE Global Communications Conference, pp. 1–6.
Tinini, R.I., Batista, D.M., Figueiredo, G.B., Tornatore, M., Mukherjee, B., 2019. Low-latency and energy-efficient BBU placement and VPON formation in virtualized cloud-fog RAN. IEEE/OSA Journal of Optical Communications and Networking 11 (4), B37–B48.
Tran, T.X., Hajisami, A., Pandey, P., Pompili, D., 2017. Collaborative mobile edge computing in 5g networks: new paradigms, scenarios, and challenges. IEEE Commun. Mag. 55 (4), 54–61.
Tse, D., Viswanath, P., 2005. Fundamentals of Wireless Communication. Cambridge University Press.
Tu, S., Waqas, M., Rehman, S.U., Aamir, M., Rehman, O.U., Jianbiao, Z., Chang, C.-C., 2018. Security in fog computing: a novel technique to tackle an impersonation attack. IEEE Access 6, 74993–75001.
Tu, S., Waqas, M., Meng, Y., Rehman, S.U., Ahmad, I., Koubaa, A., Halim, Z., Hanif, M., Chang, C.-C., Shi, C., 2020. Mobile fog computing security: a user-oriented smart attack defense strategy based on dql. Comput. Commun. 160, 790–798.
Wang, L., Zhang, D., Wang, Y., Chen, C., Han, X., M'hamed, A., 2016. Sparse mobile crowdsensing: challenges and opportunities. IEEE Commun. Mag. 54 (7), 161–167.
Wang, J., Wang, Y., Zhang, D., Wang, L., Chen, C., Lee, J.W., He, Y., 2017. Real-time and generic queue time estimation based on mobile crowdsensing. Front. Comput. Sci. 11 (1), 49–60.
Xiang, H., Yan, S., Peng, M., 2020. A realization of fog-ran slicing via deep reinforcement learning. IEEE Trans. Wireless Commun. 19 (4), 2515–2527.
Yang, M., Li, Y., Jin, D., Su, L., Ma, S., Zeng, L., 2013. OpenRAN: a software-defined ran architecture via virtualization. In: Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM. ACM, New York, NY, USA, pp. 549–550.
Yang, D., Zhang, D., Chen, L., Qu, B., 2015. NationTelescope: monitoring and visualizing large-scale collective behavior in LBSNs. J. Netw. Comput. Appl. 55, 170–180.
Yi, S., Qin, Z., Li, Q., 2015. Security and Privacy Issues of Fog Computing: A Survey, vol. 10.
Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., Kong, J., Jue, J.P., 2019. All one needs to know about fog computing and related edge computing paradigms: a complete survey. J. Syst. Architect. 98, 289–330.
Zhang, H., Qiu, Y., Chu, X., Long, K., Leung, V.C.M., 2017a. Fog radio access networks: mobility management, interference mitigation, and resource optimization. IEEE Wireless Communications 24 (6), 120–127.
Zhang, M., Fu, H., Li, Y., Chen, S., 2017b. Understanding urban dynamics from massive mobile traffic data. PP (99). IEEE Transactions on Big Data, 1–1.
Zhao, Z., Feng, C., Yang, H.H., Luo, X., 2020. Federated-learning-enabled intelligent fog radio access networks: fundamental theory, key techniques, and future trends. IEEE Wireless Communications 27 (2), 22–28.
Zheng, K., Yang, Z., Zhang, K., Chatzimisios, P., Yang, K., Xiang, W., 2016. Big data-driven optimization for mobile networks toward 5G. IEEE Network 30 (1), 44–51.

**Longbiao Chen** is an assistant professor with Fujian Key Laboratory of Sensing and Computing for Smart City, Xiamen University, China. He re-ceived a Ph.D. degree in computer science from Sorbonne University, France in 2018, and a Ph.D. degree in computer science from Zhejiang University, China in 2016. His research interests are mobile computing, big data ana-lytics, and ubiquitous computing. Dr. Chen has published over 30 papers in top-tier journals and conferences, and received the 2015 and 2016 UBICOMP Honorable Mention Awards.

**Zhihan Jiang** received the B.E. and M.E. degrees in computer science and technology from Xiamen University, China, in 2018 and 2021, respectively. She is currently pursuing the Ph. D. degree in The University of Hong Kong.

**Dingqi Yang** is a senior researcher in the Department of Computer Sci-ence, University of Fribourg, Switzerland. He received his Ph.D. in Com-puter Science from Pierre and Marie Curie University (Paris VI) and Institut Mines-TELECOM/ TELECOM SudParis, where he won both the Doctorate Award and the Institut Mines-TELECOM Press Mention in 2015. His research interests lie in big social media data analytics, ubiquitous computing and smart city applications.

**Cheng Wang** received the Ph.D. degree in information and communication engineering from the National University of Defense Technology, Changsha, China, in 2002. He is currently a Professor with and the Associate Dean of the School of Information Science and Technology, Xiamen University, Xiamen, China. He has authored more than 80 papers. His research interests include remote sensing image processing, mobile LiDAR data analysis, and multisensor fusion.

**Thi-Mai-Trang Nguyen** is an associate professor at University Pierre and Marie Curie (Paris 6) and doing research at Laboratoire d'Informatique de Paris 6 (LIP6), France. She received the PhD Degree in Computer Sci-ence from University of Paris 6, France, in 2003. The PhD thesis was co-supervised and carried-out at Ecole Nationale Superieure des Telecommunications (ENST-Paris). From 2004 to 2006, She was post-doctoral researcher at France Telecom in Rennes, France and at University of Lausanne, Switzer-land. Her research interests include network architecture, network protocol design, and network data analytics.