Data-driven C-RAN optimization exploiting traffic and mobility dynamics of mobile users

Longbiao Chen, Thi-Mai-Trang Nguyen, Dingqi Yang, Michele Nogueira, Cheng Wang, Daqing Zhang

Data-Driven C-RAN Optimization Exploiting Traffic and Mobility Dynamics of Mobile Users

Longbiao Chen, *Member, IEEE*, Thi-Mai-Trang Nguyen, *Member, IEEE*, Dingqi Yang, *Member, IEEE*, Michele Nogueira, *Member, IEEE*, Cheng Wang, *Member, IEEE*, and Daqing Zhang, *Fellow, IEEE*

Abstract—The surging traffic volumes and dynamic user mobility patterns pose great challenges for cellular network operators to reduce operational costs and ensure service quality. Cloud-radio access network (C-RAN) aims to address these issues by handling traffic and mobility in a centralized manner, separating baseband units (BBUs) from base stations (RRHs) and sharing BBUs in a pool. The key problem in C-RAN optimization is to dynamically allocate BBUs and map them to RRHs under cost and quality constraints, since real-world traffic and mobility are difficult to predict, and there are enormous numbers of candidate RRH-BBU mapping schemes. In this work, we propose a data-driven framework for C-RAN optimization. First, we propose a deep-learning-based Multivariate long short term memory (MuLSTM) model to capture the spatiotemporal patterns of traffic and mobility for accurate prediction. Second, we formulate RRH-BBU mapping with cost and quality objectives as a set partitioning problem, and propose a resource-constrained label-propagation (RCLP) algorithm to solve it. We show that the greedy RCLP algorithm is monotone suboptimal with worst-case approximation guarantee to optimal. Evaluations with real-world datasets from Ivory Coast and Senegal show that our framework achieves a BBU utilization above 85.2 percent, with over 82.3 percent of mobility events handled with high quality, outperforming the traditional and the state-of-the-art baselines.

Index Terms—Cellular network, C-RAN optimization, deep learning, big data analytics

1 Introduction

In recent decades, the number of mobile subscriptions is growing rapidly at 6 percent year-on-year, reaching 7.9 billion at the end of 2018 [1]. Correspondingly, the network traffic volume has grown 18-fold over the past five years [2] as smartphones and Internet-of-Things (IoT) devices become increasingly popular. To cope with the fast growing mobile subscribers and the surging traffic demand, network operators are deploying more and more base stations to expand their network coverage [1], and adding more powerful processing units to increase their network capacity [3]. However, as network scale and capacity grow, the capital expenditure (CAPEX) and operating expenditure (OPEX) are becoming increasingly high [4]. Meanwhile, the *service quality* of the network, such as handover and roaming delay, has become increasingly difficult to ensure as various sizes of base stations (e.g., pico-cells, micro-cells, and macro-cells) and different generations of technologies (e.g., LTE, UMTS, and GSM) co-exist in the network [5]. Therefore, designing *cost-effective and quality-aware* network architectures is now a great necessity for network operation and research [4].

Cloud Radio Access Network (C-RAN) [6] is a promising solution to address the above-mentioned challenges. To reduce maintenance cost and enable cooperation between base stations, in C-RAN, a traditional base station is split into two components: the *Remote Radio Head (RRH)* for radio communication with mobile devices, and the *Baseband Unit (BBU)* for signal and data processing [5]. The BBUs are then detached from the RRHs and hosted in centralized *BBU pools* [7]. The RRHs and BBU pools are usually connected via high speed optical fiber [8]. For example, Fig. 1 illustrates a C-RAN structure consisting of six RRHs and one BBU pool.

By adopting the C-RAN architecture, cost-effectiveness and service quality can be improved compared with the traditional radio access network architectures [5]. We exploit the example in Fig. 1 to elaborate the benefit of C-RAN. On the one hand, since multiple RRHs can be connected to one BBU and share the processing capacity (RRH #1 and #2 share BBU #1), the utilization rate of the BBUs is increased, thus improving the cost-effectiveness of the network. On the other hand, when two RRHs are connected to the same BBU (RRH #1 and #2), the handover events between them can be handled directly inside the BBU (BBU #1), which greatly reduces the handover delay and improve the quality of service. Such a seamless handover experience is of key importance in 5G networks to support direct video streaming and real-time IoT applications [9]. With the above-mentioned benefits, C-RAN is foreseen as a promising green and soft technologies in 5G networks [10].

- *L. Chen and C. Wang is with the Fujian Key Laboratory of Sensing and Computing for Smart Cities (SCSC), School of Informatics, Xiamen University, Xiamen 361005, China. E-mail: {longbiaochen, cwang}@xmu.edu.cn.*
- *T. Nguyen is with Laboratoire d'Informatique de Paris 6 (LIP6), Sorbonne Université, CNRS, 7606 Paris, France. E-mail: thi-mai-trang.nguyen@lip6.fr.*
- *D. Yang is with eXascale Infolab, University of Fribourg, 1700 Fribourg, Switzerland. E-mail: dingqi@exascale.info.*
- *M. Nogueira is with the Federal University of Paraná, Curitiba 80060-000, Brazil. E-mail: michele.nogueira@ufpr.br.*
- *D. Zhang is with Institut Mines-Télécom, Télécom SudParis, CNRS 5157 Paris, France. E-mail: daqing.zhang@telecom-sudparis.eu.*

Manuscript received 15 May 2019; revised 14 Jan. 2020; accepted 23 Jan. 2020. Date of publication 4 Feb. 2020; date of current version 2 Apr. 2021. (Corresponding author: Thi-Mai-Trang Nguyen.)
Digital Object Identifier no. 10.1109/TMC.2020.2971470

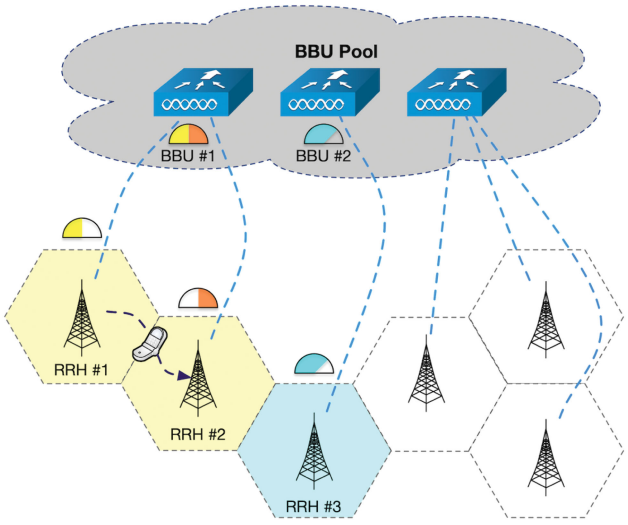


Fig. 1. An illustrative example of a C-RAN architecture consisting of six RRHs and one BBU pool. RRH traffic volume is represented in semicircle, and the mobile device between two RRHs denotes user handover between them. We note that RRH #1 and #2 are connected to BBU #1 to share its processing capacity and thus increasing its utilization rate. Moreover, the handover events between RRH #1 and #2 can be handled directly inside BBU #1 to reduces handover delay.

In order to fully unlock the power of the C-RAN architecture, one of the key problem is to design proper *mapping schemes* between RRHs and BBUs, so as to maximize the utilization rate (i.e., reduce cost) and minimize the handover delay (i.e., improve quality) for the entire network [8], [11]. To this end, a cost-effective and quality-aware RRH-BBU mapping scheme should partition the set of RRHs in the network into several *clusters* and allocates BBUs from the pool to the clusters, so that (1) the aggregated traffic volume generated in each cluster is close to the capacity of the BBU allocated to the cluster, and (2) the handover events are handled within the clusters and processed internally by the corresponding BBUs to the maximal extent. However, designing such RRH-BBU mapping schemes is not trivial, since the traffic demands and handover events among the RRHs are highly *dynamic*, and the number of possible mapping schemes is *enormous*. Specifically, the following challenges need to be addressed:

1) *How to accurately model RRH traffic volume and handover count?* In order to dynamically design RRH-BBU mapping schemes for a future period of time, we need to foresee the traffic volume and handover count in the network beforehand. However, due to the dynamic nature of user activity and mobility, the traffic volume and handover count among the RRHs can vary significantly, depending on the temporal contexts (e.g., weekdays or weekends) and spatial functions (e.g., residential areas or business districts). For example, during weekday working hours, the RRHs located in business districts and transit hubs usually observe higher traffic volume and more handover count than in other cases. Existing work on network optimization usually employ probability models with ideal assumptions (e.g., Poisson process) to *simulate* traffic patterns and handover events [7], [8], [11], which may not be able to capture the complicated traffic and handover dynamics in real-world. Therefore, we need an effective approach to accurately predict traffic and handover dynamics.
2) *How to effectively design RRH-BBU mapping schemes?* Given the predicted RRH traffic volume and handover count for a future period of time, there are potentially enormous numbers of schemes to cluster these RRHs and to allocate BBUs to these clusters. Since the traffic volume and handover count may vary significantly under different contexts, the mapping schemes need to be updated dynamically. Moreover, the *global resource constraints* of the BBU pools, such as the pool capacity and the BBU size (e.g., CPU and memory specifications), should be taken into consideration during the search of the mapping schemes. Existing work with exhaustive search quickly becomes intractable as network scale grows [8], while competitive optimization methods such as borrow-and-lend [12] and swarm optimization [13] approaches suffer from switching overhead and nonlinear time complexity [14]. Therefore, we need an effective algorithm to design RRH-BBU mapping schemes with cost and quality objectives under resource constraints.

Fortunately, with the emergence of ubiquitous sensing, communication and computing diagrams [15], a massive number of *cellular network big data*, such as call detail records (CDRs), have been generated, providing researchers with new opportunities to understand the mobile user dynamics [16]. The knowledge discovered from these big data can be used to guide the optimization of cellular networks [17]. In this work, aiming at achieving the cost and quality objectives, we propose a *data-driven C-RAN optimization* framework to address the above-mentioned research challenges. Particularly, we first propose a *deep learning model* to accurately predict network traffic volume and handover count, and then propose an greedy optimization algorithm to design RRH-BBU mapping schemes with cost and quality objectives. The *main contributions* of this work are:

- We propose a novel data-driven approach to dynamically optimize operation cost and service quality for the C-RAN architecture. By analyzing the traffic and mobility patterns from real-world network big data, we are able to dynamically optimize RRH-BBU mapping schemes for demand-responsive C-RAN.
- We design a two-phase framework to design dynamic RRH-BBU mapping schemes based on the accurate prediction of traffic volume and handover count. In the first phase, we extract traffic volume and handover count from large-scale call detail records, and capture their spatiotemporal dynamics leveraging tensor models. We then propose a *deep-learning-based* Multivariate Long Short Term Memory (MuLSTM) model to accurately model and predict the traffic volume and handover count. In the second phase, we model the entire cellular network as a *weighted graph*, taking into consideration the traffic and handover as graph nodes and link weights, respectively. We then formulate the RRH-BBU mapping problem with cost and quality objectives as a

set partition problem, and propose a Resource-Constrained Label-Propagation (RCLP) algorithm to find the robust approximation to the optimal mapping schemes under pool resource constraints.
- We evaluate the performance of our framework on two large-scale, real-world call detail record datasets collected by Orange S.A. in Ivory Coast and Senegal. Results show that our framework effectively achieves a BBU utilization rate above 85.2 percent and an cluster handover rate above 82.3 percent, which consistently outperforms the traditional RAN architectures and other state-of-the-art baselines.

The rest of this article is organized as follows. We begin by reviewing the related works in Section 2. After introducing the preliminaries about C-RAN, we present an overview of the proposed framework in Section 3. We then detail the two phases of the framework, i.e., RRH traffic and handover prediction in Section 4, and dynamic RRH-BBU mapping in Section 5. Extensive evaluation results are presented in Section 6 to verify the performance of the proposed framework. Finally, we conclude this work and chart the future directions in Section 7.

2 Related Work

In this section, we first present an overview of the cloud radio access networks, and then review the existing works on C-RAN optimization and network big data analytics.

2.1 Cloud Radio Access Network Optimization

Cellular network operators and researchers are continuously seeking for optimal solutions to provide stable telecommunication, high speed data rate, and high services quality to their users [6]. C-RAN is targeted by worldwide cellular network operators as a promising solution to address these challenges [5]. In 2010, IBM proposed wireless network cloud (WNC) [18], which 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 [18]. In 2011, China Mobile Research Institute envisioned a cloud-based RAN architecture to provide broadband Internet access to wireless customers with low bit-cost, high spectral and energy efficiency [6]. For a comprehensive technology survey on C-RAN, the reader is referred to [5]. In the literature, various topics about C-RAN optimization have been studied. In resource allocation optimization, Ha *et al.* [19] proposed a network slicing framework for OFDMA-based C-RAN shared by multiple operators. To improve spectral efficiency, Sun *et al.* [20] proposed a distributed optimization approach for uplink device-to-device-enabled C-RAN. In sum-rate maximization, Pan *et al.* [21] studied efficient approaches for ultra-dense TDD C-RAN with imperfect CSI. In this work, we focus on the optimization of costs and quality in C-RAN.

2.1.1 Network Traffic Responsiveness

One of the key vision in C-RAN is to provide flexible and configurable data processing capacity according to the traffic demands [6], [9]. In [17], such a vision is coined as a *cognitive networking diagram*. To this end, cooperations among RRHs are necessary to cope with the vibrations in traffic demands [5]. For example, Bhaumik *et al.* [22] proposed CloudIQ, a framework for partitioning a set of RRHs into groups and process the signals in a shared data center, which was able to save up to 19 percent of the computing resources for a probability of failure of one in 100 million. Namba *et al.* [23] proposed an C-RAN architecture that can dynamically change the cooperation schemes of RRHs in response to traffic demand, which proves to reduce the number of BBUs by 47 percent compared with the static assignment. However, these existing works are usually based on predefined models with ideal traffic assumptions. For example, Poisson processes are usually employed to model the patterns of phone calls [8], [24]. These models usually require specific parameters for each RRH, which may not be able to accurate characterize the patterns in real-world networks. Furno *et al.* took a first step in the direction of RRH traffic profiling leveraging a data-driven approach [16], [17]. By exploiting traffic analytics algorithms for large-scale real-world cellular network datasets, they were able to characterize the network demand patterns in different areas in an automated manner. This work inspires us to propose a data-driven approach to capture network traffic dynamics.

2.1.2 User Mobility Awareness

Another key issue in the design and implementation of C-RAN lies in user mobility [25], [26]. One of the important objectives in 5G is to improve the quality of cellular service, with handover events nearly invisible to the mobile users. To this end, the RRHs in a network need to be able to cooperate with each other to seamlessly transfer user contexts, forward network resources, and assign cellular channels [27]. This raises an important problem of foreseeing the mobile user mobility dynamics in next few hours. Traditionally, user mobility is usually ideally modeled with specific assumptions, e.g., random walk variables with specific moving speed and diameters [27]. These assumptions ignore the spatiotemporal variations and dynamics of user mobility, which might be inaccurate to foresee the user movement in a future period of time. Recently, researchers have sought to data-driven user mobility modeling and prediction for C-RAN optimization. In [28], the authors proposed a machine learning framework with echo state networks (ESNs) to predict each mobile user's mobility pattern for effective content caching in the cloud. [26] proposed an online algorithm to optimize the mapping between BBUs and RRHs in C-RAN based on a time-varying graph. Our work differs from the literatures in the following two aspects, (1) we simultaneously optimize handover cost and BBU utilization in the proposed RRH-BBU mapping algorithm, while [26] only focused on reducing handover costs, and (2) we propose a deep learning-based approach to accurately predict traffic and mobility dynamics, while [26] adopted an online strategy with simple prediction.

2.2 Network Big Data Analytics

A massive number of cellular datasets have been available for academic research and industrial analytics [3]. They can be collected either from operators' infrastructures [29], [30], [31], or by leveraging mobile crowdsensing paradigms [32], [33]

with user participation. For example, Telecom Italia [31] has released a large-scale call detail records dataset containing two-months of calls, SMSs and network traffic data from the city of Milan and the province of Trentino, Italy. Orange S.A. [29] has also granted access to researchers participating in their Data for Development (D4D) challenges the access to a large-scale anonymized call detail records dataset, which consists of phone calls and SMS exchanges between five million of Orange's customers in Ivory Coast in half a year. These network big data have been analyzed in the literature to retrieve interesting and informative knowledge [3], [34]. The following two categories of data analytics methodologies are most relevant to this work.

2.2.1 Spatiotemporal Prediction

Autoregressive Integrated Moving Average (ARIMA) models have been widely used to fit a time series data and to predict its future variation [35]. However, ARIMA models are usually used to model *single* variables (e.g., one RRH). When dealing with spatiotemporal data in C-RAN, multiple time series denoting a set of correlated RRHs need to be modeled simultaneously, which poses great challenge for vanilla ARIMA models. Ntalampiras *et al.* [36] proposed a improved method to select time series from stongly correlated regions, and feed them together to ARIMA models to improve prediction accuracy.

Artificial Neural Networks (ANNs) are emerging for modeling spatiotemporal data [37], [37]. A typical implementation is by leveraging a sliding-window-based technique, which can be named *windowed-ANN*, or WANN [38]. Specifically, WANN slices a time series into several equal-length windows, and feeds these windows into an ANN model as *features*. The *output* of the model is the prediction of the future value of the series. WANN models have been applied to analyze temporal patterns in various domains [37]. However, WANN models are incapable of capturing the *temporal dependence* between different time step in the input time series window. In fact, the elements in a window is treated equally as input features and thus the *sequential order* of the elements is ignored. Hence, the WANN model can make fluctuating and inconsistent forecasts.

Recently, spatiotemporal deep-learning framework has been used in IP and transportation network traffic prediction [39] and human motion and behavior recognition [40]. For a survey about deep-learning-based prediction model for spatiotemporal data, the reader is referred to [41]. One of the relevant existing works is by Zhang *et al.* [42]. The authors proposed a double Convolutional Long Short-Term Memory Network (ConvLSTM) architecture to make accurate long-term prediction. We note that training such a complicated model is time-consuming and requires high-end GPU acceleration. Wang *et al.* [43] also propose a spatiotemporal deep-learning approach for cellular network traffic prediction. They incorporated an autoencoder model for spatial modeling and an LSTM model for temporal modeling. In [28], the authors build two ESNs to model network content distribution and user mobility patterns. Compared to the proposed MuLSTM approach, ESNs has the following limitations. (1) ESN simplifies the network training process by directly connecting the input signal to a random and non-trainable RNN (the reservoir) [44]. However, we argue that such a simplified model is not capable of capturing the highly dynamic traffic demands and mobility patterns in the hidden layers. (2) ESN is sensitive to parameters, the selection of parameters in ESNs require experience and insight to achieve a good performance in many tasks [44]. (3) The proposed MuLSTM model train the traffic and mobility layers simultaneously to incorporate the correlations and dependencies, which is very difficult to implement for ESNs due to inconsistent variable dimensions.

2.2.2 Clustering and Mapping

In network data analytics, clustering is a very important and useful technique for *discovering patterns* from a wide range of spatial regions [17], and for *reducing fluctuations* in individual spatial areas [45]. In [46], Naboulsi *et al.* proposed a framework to identify a set of clusters of call profiles, and classify the network usages accordingly. Similarly, Cici *et al.* [47] proposed a spectral method to cluster area units with similar activity patterns and validated the results with external municipal and social data sources. Furno *et al.* [17] proposed to cluster the traffic demand in the temporal dimension, by adopting a hierarchical clustering method on the city-wide traffic snapshots.

In C-RAN, mapping RRH clusters to BBU pools is not trivial, since the clusters also need to meet some explicit and implicit constraints, including the geographic distance of the cluster, the global constraints on the resource blocks available, etc [5]. Such a problem has been identified as *set partitioning problem* [48], [49], and its complexity is proven to be NP-hard [50]. Therefore, exhaustively searching for every possible mapping scheme is computationally intractable as the network scale increases [8].

Instead of exhaustive search, Chen *et al.* [12] proposed a borrow-and-lend approach to dynamic switch RRHs from busy BBUs to neighboring candidates. However, such an ad-hoc switching mechanism introduces extra control overhead and may lead to redundant switching [12]. In [51], RRH-BBU mapping is formulated as a bin packing problem and solved with integer linear programming algorithms. However, this formulation assumes unified BBU capacities (fixed-size bins), while in practice we need to deal with BBUs with various capacity levels (as defined in Equation (3)). In [52] and [13], a particle swarm optimization-based algorithm is adopted to find optimal RRH-BBU mapping schemes, but the stochastic optimization process has nonlinear time complexity and may result in sub-optimal solutions as network scales grows [52]. Some other work adopt distributed methods such as coalitional games, where the players (RRHs) decide to form or leave the coalitions (BBUs) based on the transfer order [53]. However, it is not trivial to define a proper transfer order to guarantee a strict utility improvement under the geographic and resource constraints in the centralized BBU pool [14].

3 Preliminaries and Framework

3.1 Preliminaries

Definition 1 (Remote Radio Head). *an RRH is the radio transceiver placed in a base station site to facilitate wireless*

communication between user devices and the network [54]. We define an RRH r as a 3-tuples

$$r = < label, lat, lng >, \quad (1)$$

where $label$ is the label used to identify the RRH, and lat and lng are the corresponding latitude and longitude coordinates of the RRH.

Definition 2 (RRH Traffic Volume). *in this work, we refer to the term* traffic volume *as the quantity of radio resource units [8] consumed in the RRH for communication during a period of time, which can be the total duration of calls, the overall volume of Internet data, etc. Particularly, we denote the traffic volume of RRH r_i during time span t as $f(r_i, t)$.*

Definition 3 (RRH Handover Count). *in this work, we refer to the term* handover count *as the quantity of users moving between a pair of two RRHs during a period of time. Particularly, we denote the handover count between RRH r_i and RRH r_j during time span t as $h(r_i, r_j, t)$.*

Definition 4 Baseband Unit. *a BBU is a device providing baseband processing functionalities for RRHs, such as such as time multiplexing, encapsulation, and compression [5], [54]. Specifically, we define a BBU as a 3-tuple*

$$b = < label, pool, cluster >, \quad (2)$$

where $label$ is the label to identify a BBU instance, $pool$ is the BBU pool where the BBU is allocated, and $cluster$ is the RRH cluster where the BBU is assigned to.

Definition 5 (BBU Capacity). *in the C-RAN architecture, BBUs are usually implemented as virtual machine instances with specific sizes of computing resources, including CPU, memory, and storage [5]. Consequently, the BBU capacity can be classified into a set of* discrete *levels, e.g.,* LARGE, MEDIAN, *and* SMALL. *Specifically, we define the set of BBU capacity level as*

$$\mathbb{L} = \{l_1, \ldots, l_{N_l}\}, \quad (3)$$

where N_l is the number of capacity levels. Correspondingly, we denote the capacity level of BBU b_k as $l(b_k) \in \mathbb{L}$.

Definition 6 (BBU Pool). *in the C-RAN architecture, a BBU pool is a cloud-based data center with low-cost and high-speed interconnect network, a real-time virtualization platform with dynamic shared resource allocation and management, and a general-purpose baseband processing platform with multiple BBUs [5], [6]. For a city-scale network, one or more BBU pools can be implemented and connected to RRHs via high-speed optical fiber. Specifically, we denote a BBU pool as a set of BBUs*

$$\mathbb{B} = \{b_1, b_2, \ldots, b_k\}. \quad (4)$$

In this work, we consider a C-RAN architecture with one centralized BBU pool for a city-wide cellular network. The the fronthaul transmission latency between each RRH and the BBU pool is considered as constant and not impacted by the RRH-BBU mapping scheme. The benefits of adopting such a centralized pool are three-fold. First, the deployment cost and energy consumption can be greatly reduced by employing data center virtualization technologies [5]. Second, the handover handing and contents offloading among RRHs can be processed internally in the pool, which significantly reduces delays and increases throughput [5]. Third, the network upgrades and hardware maintenance are easy to conduct just in one place, without the need of labor-consuming on-site work.

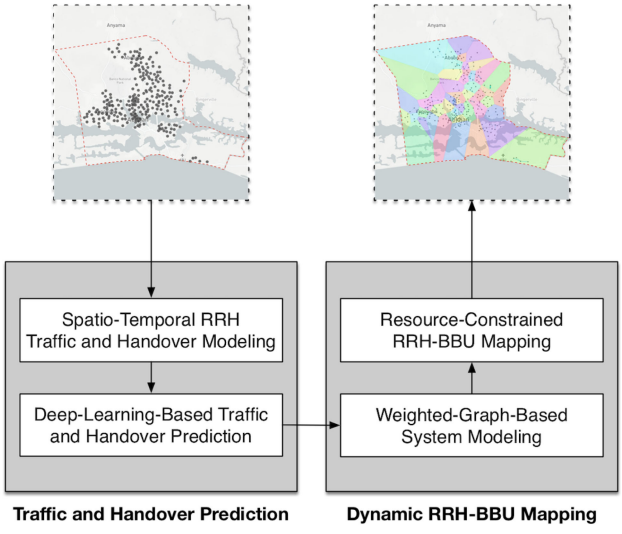


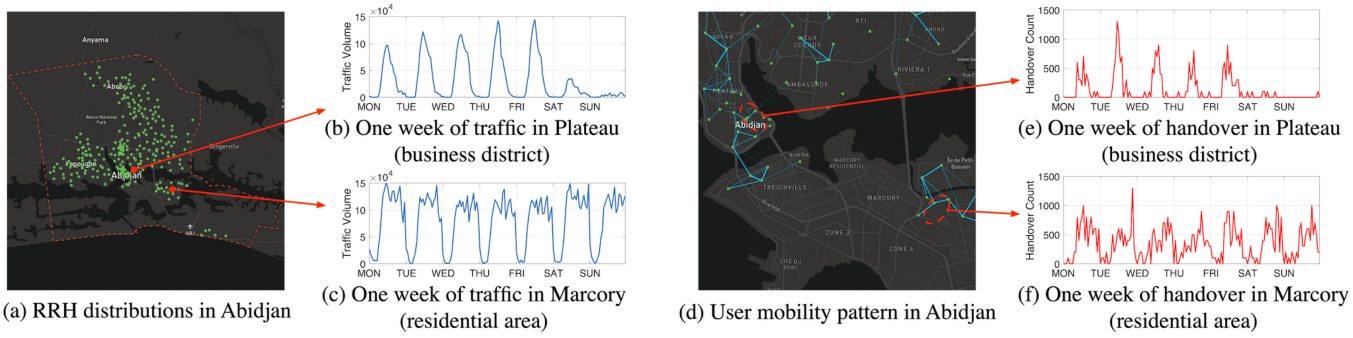
Fig. 2. Framework overview.

3.2 Framework Overview

As presented in Fig. 2, we propose a two-phase framework to accurately predict RRH traffic volume and handover count based on historical data, and then dynamically design RRH-BBU mapping schemes under constraints for C-RAN optimization. In the *traffic and handover prediction* phase, we first model the traffic and handover dynamics among RRHs with spatial and temporal dimensions, and then propose a deep-learning-based approach to predict the traffic volume and handover count simultaneously for a future period of time. In the *dynamic RRH-BBU mapping* phase, we first model the network with predicted RRH traffic volume and handover count as a weighted graph, and identify the resource constraints from the BBU pools. We then propose a resource-constrained RRH-BBU mapping algorithm to find the robust approximation to the optimal solution under pool resource constraints.

4 RRH Traffic and Handover Prediction

In this phase, our objective is to accurately predict the RRH traffic volume and handover count in a future period of time, hences we can design the RRH-BBU mapping scheme in the next phase. However, this is not trivial due to the highly dynamic nature of social activity and human mobility. On the one hand, the RRH traffic and user mobility behaviors may vary significantly under different *temporal* contexts. On the other hand, the *spatial* function of an area may have strong impacts on the traffic and mobility patterns of the RRHs located in that area. Fig. 3 shows an example of the traffic and mobility dynamics in Abidjan, Ivory Coast during one week. In the business district (Plateau), we can observe different traffic (Fig. 3b) and mobility (Fig. 3e) patterns between weekdays and weekends. Meanwhile, the traffic and mobility patterns in the residential area (Marcory) exhibit quite different patterns.



(a) RRH distributions in Abidjan (b) One week of traffic in Plateau (business district) (c) One week of traffic in Marcory (residential area) (d) User mobility pattern in Abidjan (e) One week of handover in Plateau (business district) (f) One week of handover in Marcory (residential area)

Fig. 3. The RRH traffic and user mobility dynamics in Abidjan, Ivory Coast during a sample week (01/09/2012–01/15/2012). In map (a), each green dot corresponds to an RRH. In map (c), each blue link denotes user mobility between RRHs pairs. Wider link corresponds to higher handover count.

In a word, the real-world traffic and mobility patterns demonstrate *temporal dependency* and *spatial correlation*. However, traditional prediction methods usually model each RRH traffic as single time series, and model the user mobility as static graphs [55], which fails to integrate the spatial and temporal dynamics in a unified model, and hinders the accurate prediction of RRH traffic and handover. Therefore, we propose a deep-learning based approach to model the spatial and temporal dynamics as a multivariate Long Short Term Memory neural network for accurate prediction.

4.1 Call Detail Records Dataset

In cellular networks, call detail records (CDR) are data that document the details of phone calls, text exchanges, or other telecommunication transaction that pass through the network infrastructures [34]. CDR data contain rich information about social activity and human mobility, providing opportunities to optimize network infrastructures, such as reducing operation cost and improving service quality. In this paper, we exploit two real-world large-scale anonymized CDR datasets released by Orange S.A. via the Data for Development (D4D) Challenge.[1] The datasets are collected from Orange customers from Ivory Coast for half-a-year, and Senegal in one year, respectively. The datasets consists of the following information:

- *RRH Attributes*: the RRH labels and geographic coordinates.
- *RRH Communication*: the number and durations of phone calls and SMS exchanges between RRHs in the network on an hourly basis.
- *User Attributes*: the anonymized user labels which are shuffled every two weeks for privacy concerns.
- *User Mobility*: the user mobility trajectories among RRHs in the network with precise time and RRH information.

Based on the datasets, we aggregate the communication and mobility events by RRH, and perform data cleansing process to extract the RRH traffic volume and RRH handover count on hourly basis, respectively. More details about the datasets are presented in the evaluation section.

4.2 Spatiotemporal Traffic-Handover Modeling

To capture the spatiotemporal dynamics of the RRH traffic and handover, we construct two *tensors* [56] to model the traffic volume generated in each RRH and the handover counts observed among each RRH pair, respectively. Specifically, given a network with N_r RRHs and the corresponding CDR data observed in N_t time spans, the *RRH traffic tensor* and *RRH handover tensor* are defined as follows.

RRH Traffic Tensor: we build a tensor $\mathcal{F} \in \mathbb{R}^{N_r \times N_t}$ with two dimensions to model the RRH traffic volume, where $\mathcal{F}(r_i, t)$ corresponds to the total incoming and outgoing communication traffic volume of RRH r_i during time span t $(i = 1, \ldots, N_r, t = 1, \ldots, N_t)$. We note that based on different scenarios, the definition of traffic may vary, such as the total duration of calls, the number of messages, and the overall volume of Internet data. For example, Figs. 3b and 3c visualize two typical traffic patterns extracted from two specific RRHs in $\mathcal{F}$.

RRH Handover Tensor: we build a tensor $\mathcal{H} \in \mathbb{R}^{N_r \times N_r \times N_t}$ with three dimensions to model the RRH handover counts, where $\mathcal{H}(r_i, r_j, t)$ corresponds to the total count to handover events between RRH r_i and RRH r_j during time span t. We consider the case of symmetric modeling where $\mathcal{H}(r_i, r_j, t) = \mathcal{H}(r_j, r_i, t)$. As an example, Figs. 3e and 3f visualize two typical handover patterns extracted from two specific pairs of RRHs in $\mathcal{H}$.

4.3 Deep-Learning-Based Traffic-Handover Prediction

Deep learning approaches have been widely applied to capture the spatial and temporal dynamics of urban traffic and human mobility [24], [24]. Particularly, Recurrent Neural Networks (RNNs) are proposed for time series modeling and prediction [37]. Built upon the traditional neural network architecture, an RNN features *recurrent connections* among internal nodes that add a state to the network architecture, and thus allowing it to learn and harness the temporal dependency in the time series [57]. Unfortunately, training an RNN effectively is technically challenging due to the *vanishing or exploding gradient problem* [58], i.e., the weights in the training procedure quickly became so small as to have no effect (vanishing gradients) or so large as to result in very large changes (exploding gradients). To overcome this problem, researchers proposed the Long Short-Term Memory Network (LSTM) model [59], which introduces the concepts of memory cells and forget gates to generate consistent data flow between the layers of the network and keep the weights stable.

1. http://www.d4d.orange.com/

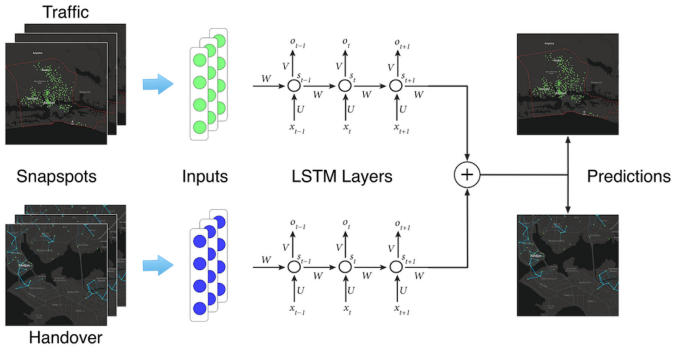


Fig. 4. The MuLSTM model for RRH traffic and handover prediction.

The MuLSTM Model: in this work, we use LSTM networks to effectively learn the temporal dependency of RRH traffic patterns and handover patterns from historical data. To further model the *spatial correlation* among RRHs in the network, we propose a multivariate Long Short Term Memory Network architecture to simultaneously integrate all the RRHs in a unified model. Specifically, each RRH traffic is regarded as an input variable to a shared LSTM model, while each RRH handover pair is regarded as an input variable to another shared LSTM model. The two LSTM models accept the multivariate inputs and are trained jointly. Fig. 4 shows the overview of the proposed MuLSTM model. We elaborate on the technical details as follows.

Snapshots: we generate two sets of consecutive *traffic and handover snapshots* based on the traffic tensor $\mathcal{F}$ and the handover tensor $\mathcal{H}$, respectively. A snapshot is a slice of the tensor along the time axis, which corresponds to the traffic or handover observations among all RRHs during one hour, and can be denoted as $F_t = \mathcal{F}(:,t)$ or $H_t = \mathcal{H}(:,:,t)$, respectively. Consequently, a set of consecutive traffic snapshots can then be represented as $\{F_t, F_{t-1}, \ldots, F_{t-N_s}\}$, and a set of consecutive RRH handover snapshots as $\{H_t, H_{t-1}, \ldots, H_{t-N_s}\}$, where N_s is the number of snapshots in the set.

Inputs: we extract the appropriate inputs for the LSTM layers based on the snapshots. For traffic, we simply stack N_r RRH traffic observations in each snapshot to form an input vector, and select N_s snapshots as the look-back time steps [57]. For handover, since there are $N_r \times N_r$ handover pairs, directly constructing an input vector with such high dimension will be computational impossible for the LSTMs. In fact, many of the RRH pairs do not observe meaningful handover counts since they are geographically distant from each other. Therefore, we adopt a hypothesis-based method to select RRH pairs with *statistical significant handover counts*. Based on our observation from the dataset, a significant handover count series exhibits large variations (i.e., over-dispersion) [60], thus we make a hypothesis that the handover count of an RRH pair follows the negative binomial distribution [60]. We test each RRH pair again this hypothesis and remove failure pairs. In this way, we obtain N_h pairs of RRHs with significant handover counts. We stack the pairs in each snapshot to form an input vector, and use N_s look-back time steps for the LSTM layers.

LSTM Layers: we build two LSTM neural networks for the traffic and handover inputs, respectively. The traffic LSTM accepts an input of N_r traffic variables with N_s time steps, while the handover LSTM accepts an input of N_h handover pairs with N_s time steps, respectively. As illustrated in Fig. 4, for each time step, the hidden unit s_t in the network computes its current activation o_t as a nonlinear function of both the current input weights U and the weights from the previous state W. In this way, the networks are able to keep a memory of the previous perception and use the knowledge for current decision making.

Predictions: the LSTM layers output the RRH traffic volume and handover count for the next time step as predictions. In order to exploit the correlation between RRH traffic and handover, we aggregate the outputs via an addition neural unit, and train the the two LSTMs jointly using the Backpropagation Through Time (BPTT) algorithm [57] for multiple iterations. We run our prediction algorithm in an online manner, i.e., at the end of each time step t, we make a new prediction for the traffic and handover of $t+1$. We construct a tensor $\hat{\mathcal{F}} \in \mathbb{R}^{N_r \times N_t}$ to store the traffic prediction, and a tensor $\hat{\mathcal{H}} \in \mathbb{R}^{N_r \times N_r \times N_t}$ to store the handover prediction, respectively. The prediction results are then used in the next phase for RRH-BBU mapping.

5 Dynamic RRH-BBU Mapping

In this phase, given the RRH traffic and handover predictions as well as the BBU pool constraints, our objective is to design an optimal RRH-BBU mapping scheme that maximizes BBU utilization rate and minimizes RRH handover overhead. Such a problem has been identified as *set partitioning problem* [48], [49], and its complexity is proven to be NP-hard [50]. Therefore, exhaustively searching for every possible mapping scheme is computationally intractable as the network scale increases [8]. In order to address these challenges, we propose a resource-constrained RRH-BBU mapping approach based on weighted-graph model and label propagation algorithm. We first introduce the system model the problem formulation, and then propose an algorithm to find robust approximations to the optimal RRH-BBU mapping schemes under resource-constraints.

5.1 System Model

Based on the above-mentioned definitions, we model a cellular network as an undirected, weighted graph $G = (V, E)$, where $V = \{v_1, \ldots, v_{N_r}\}$ is the set of graph *nodes* denoting the N_r RRHs, and E is the set of graph *links* corresponding to the significant handover pairs (as defined in the previous section) among RRHs. We consider our dynamic RRH-BBU mapping problem in an *online* manner, i.e., at the end of time span t, we make decision of the RRH-BBU mapping scheme for the next time span $t+1$. To this end, we need to update the graph dynamically in each time span.

Graph Weights: we initialize the above-mentioned graph at $t+1$ as $G(t+1)$ with the traffic and handover predictions. Specifically, we define the *weight of node* $|v_i| = \hat{\mathcal{F}}(i, t+1)$, which corresponds to the traffic volume of the RRH i in time span $t+1$. Similarly, we define the *weight of link* $|e_{i,j}| = \hat{\mathcal{H}}(i, j, t+1)$, which is the handover count between RRH i and j in time span $t+1$. We note that if there is no predicted handover count between RRH i and j, then the link weight is set to 0, and we remove the corresponding link in $G(t+1)$. We consider the case of symmetric link weights ($|e_{i,j}| = |e_{j,i}|$) with no loops ($|e_{i,i}| = 0$).

Constraints: we model the resource constraints in the BBU pool according to the available BBU capacity. Since BBUs in the pools are implemented as virtual machine instances with specific sizes of computing resources, their capacities can be classified into a set of discrete levels. For example, we can denote a set of BBU capacity level as $\Phi = \{PICO, SMALL, MEDIAN, LARGE, \ldots\}$, each corresponding to a specific computing resource configuration. The capacity of an allocated BBU b_k shall be in one of the capacity levels, i.e., $\phi(b_k) \in \Phi$. Note that we do not explicitly constrain the number of available BBU with specific capacity level, since large-capacity BBUs (virtual machines) can be allocated by merging two or more small-capacity BBUs, and vice versa. Instead, we constrain the overall *capacity limit* of a BBU pool to be $\mathcal{O}$, since the capacity of a BBU pool is usually fixed once it is deployed. We study the impacts of different capacity level and capacity limit combinations in the evaluation section.

BBU Utilization Rate: once a BBU b_k is allocated to an RRH or a cluster of RRHs c_k in the time span $t+1$, its utilization rate can be calculated as

$$U(c_k) = \frac{\sum_{v_i \in c_k} |v_i|}{\phi(b_k)} \quad \in [0,1], \tag{5}$$

where $\{v_i\}$ is the set of graph nodes corresponding to the cluster of RRHs mapped to the BBU, and $|v_i|$ is the traffic volume of the RRH i in time span $t+1$. BBU utilization rate is one of the key objectives in optimizing our RRH-BBU mapping scheme. Since the BBU capacity $\phi(b_k)$ is discrete and the traffic generated in a cluster is continuous, it is important to ensure that the aggregated traffic volume in the cluster is close to the corresponding BBU capacity. Note that in order to avoid BBU processing latency due to traffic congestion [8], we do not allocate BBU for clusters whose aggregated traffic volume is larger than the maximum available BBU capacity in the pool. Thus, we constrain $U(c_k) \in [0,1]$ to avoid cluster traffic overflow.

Cluster Handover Rate: the other key objective in RRH-BBU mapping optimization is to maximize the extent to which the handover events are processed within a BBU (i.e., the corresponding RRHs are in a cluster). Inspired by the influence maximization model in social networks [61], we derive the *cluster handover rate* of cluster c_k as

$$W(c_k) = \frac{\sum_{v_i \in c_k} \sum_{v_j \in V} |e_{i,j}|}{2|E|} \quad \in [0,1], \tag{6}$$

where $\{e_{i,j}\}$ is the handover count between an RRH in the cluster and any other RRH in the graph reachable by it. $|E|$ corresponds to the handover count between all RRH pairs.

5.2 Problem Formulation

With the above-mentioned system model, we now present the problem formulation for the BBU-RRH mapping problem with the objectives of maximizing BBU pool utilization and minimizing handover costs. We argue that these two objectives do not conflict with each other in our problem. Based on data observation, user mobility patterns exhibit strong *locality*, i.e., user handover events are frequently observed among neighboring RRHs in a period of time. Meanwhile, user traffic demands in these neighboring RRHs (forming a community) tend to be *complementary*. For example, during rush hours, when users move from residential area A to neighboring transit hub B, the traffic volumes of A and B are complementary to each other as the aggregated volume keeps stable.

Specifically, given the graph representation $G(t+1)$ of a set of RRHs with the corresponding traffic and handover prediction, as well as the BBU pool resource constraints, our objective is to partition the graph into a set of N_k disjoint clusters $\mathbb{C} = \{c_1, \ldots, c_{N_k}\}$, and map each cluster c_k to a BBU b_k in the BBU pool $\mathbb{B} = \{b_1, b_2, \ldots, b_k\}$, with the following objective function and constraints:

$$(\mathbf{P1}): \quad \underset{\mathbb{C}}{\text{maximize}} \quad U(\mathbb{C}) + W(\mathbb{C}) \tag{7}$$

$$= \underset{\mathbb{C}}{\text{maximize}} \quad \frac{1}{N_k}\left(\sum_{k=1}^{N_k} U(c_k) + \sum_{k=1}^{N_k} W(c_k)\right) \tag{8}$$

$$= \underset{\mathbb{C}}{\text{maximize}} \quad \frac{1}{N_k}\left(\sum_{k=1}^{N_k} \frac{\sum_{v_i \in c_k} |v_i|}{\phi(b_k)} + \sum_{k=1}^{N_k} \frac{\sum_{v_i \in c_k} \sum_{v_j \in V} |e_{i,j}|}{2|E|}\right). \tag{9}$$

Subject to

$$(\mathbf{C1}): \quad \cup_{\forall c_k \in \mathbb{C}} = V \quad \text{and} \quad \cap_{\forall C_k \in \mathbb{P}} = \emptyset \tag{10}$$

$$(\mathbf{C2}): \quad U(c_k) \in [0,1] \tag{11}$$

$$(\mathbf{C3}): \quad \phi(b_k) \in \Phi \tag{12}$$

$$(\mathbf{C4}): \quad \sum \phi(b_k) \leq \mathcal{O}. \tag{13}$$

In this problem formulation, constraint **C1** ensures that the clusters form a complete disjoint partition of the graph. Constraints **C2** is posed to avoid large clusters with aggregated traffic volume higher than the maximum available BBU capacity. Constraints **C3–C4** make sure that the allocated BBU capacity can only be discrete values specified by the pool configuration, and their overall capacity can not exceed the resource limit $\mathcal{O}$.

5.3 Proposed Algorithm

The problem **P1** is indeed a graph partitioning problem (GPP) [62], which has been proved to be an NP-hard problem [49], [62]. To tackle this difficult problem, we resort to a fast heuristics approximation algorithm named label propagation (LP) [63]. The basic idea of label propagation is to initialize each node in the graph as a cluster, and iteratively assign a node to its neighboring cluster based on a *gain* function [63]. However, directly applying an label propagation algorithm to our problem may not be adequate, since we also need to impose the resource constraints from the BBU pool, including available BBU capacity levels and pool capacity limit. Therefore, we propose a Resource-Constrained Label Propagation algorithm to solve this problem. We elaborate on the details as follows.

Algorithm: as shown in Algorithm 1, the RCLP algorithm is initialized by assigning each node in the graph to a unique cluster label. In each iteration, we randomly populate a list of node labels L and traverse the list to update the cluster label of each node. The label update process is as follows. First, we remove the node from its current cluster, and find the set of adjacent clusters to the current node. Then, we compute the gain for adding the current node to the adjacent clusters, and assign it with the label of the cluster with the highest gain.[2] We mark the the node as *moved* among clusters if its new cluster label is different from the old one.

After finishing iterations over the node list, we evaluate whether the allocated resources of the resultant cluster partition are within the capacity limit of the BBU pool. If not, we reset the cluster labels and restart the optimization procedure. As the convergence speed of such a greedy algorithm is difficult to estimate, we set a maximum iteration number max_iter to stop the algorithm. At the end of each iteration, we decide whether to perform another iteration or finish the algorithm based on the following stop criteria: (1) the user specified maximum iteration number max_iter is reached, or (2) none of the nodes are moved among clusters.

Algorithm 1. The RCLP Algorithm

Input: Graph $G(t+1) = (V, E)$, pool capacity limit $\mathcal{O}$, maximum iteration number max_iter
Output: Cluster labels L for nodes in the graph
▷ `cluster label assignment`
Initialize: $L \leftarrow 1, \ldots, N$
while $(iter < max\_iter) \wedge (move > 0)$ **do**
  ▷ `random permutation of nodes`
  $rand\_perm(V)$;
  $move \leftarrow 0$;
  for $i \leftarrow 1$ **to** N_r **do**
    ▷ `remove current node from its cluster`
    $old\_label \leftarrow L(v_i)$;
    $L(v_i) \leftarrow null$;
    ▷ `select adjacent clusters`
    $\mathbb{C}_{v_i} = get\_adjacent\_clusters(v_i, G, L)$;
    $max\_gain \leftarrow 0$;
    for $c \in \mathbb{C}_{v_i}$ **do**
      ▷ `find cluster with highest gain`
      $gain \leftarrow compute\_gain(v_i, c)$;
      if $gain \geq max\_gain$ **then**
        $new\_label \leftarrow L(c)$;
        $max\_gain \leftarrow gain$;
      end
    end
    ▷ `update current node label`
    $L(v_i) \leftarrow new\_label$
    if $old\_label \neq new\_label$ **then**
      $move \leftarrow 1$;
    end
  end
  ▷ `reset labels if capacity limit exceeded`
  if $allocated\_capacity(L) > \mathcal{O}$ **then**
    $L \leftarrow 1, \ldots, N$
  end
end

2. If two clusters yield the same gain, we randomly choose one.

Gain Function: the gain function is used to determine whether a node should be added to an adjacent cluster, and it shall take into consideration the improvement in both BBU utilization and handover performance. To this end, we first design the *utilization gain* of adding node v_i to cluster c_k as

$$gain\_u(v_i, c_k) = \max \Gamma(|\{v_i\} \cup c_k|, l), \quad l \in \mathbb{L}, \tag{14}$$

where $\Gamma(|\{v_i\} \cup c_k|, l) =$

$$\begin{cases} \frac{|\{v_i\} \cup c_k|}{l}, & \text{if} |\{v_i\} \cup c_k| \leq l \quad (15) \\ -log(\frac{|\{v_i\} \cup c_k|}{l}), & \text{if} |\{v_i\} \cup c_k| > l \quad (16) \end{cases}.$$

The rationale is like this: suppose we add v_i to c_k to form a candidate cluster $\{v_i\} \cup c_k$, we try to allocate BBUs with different capacity levels $l \in \mathbb{L}$ to the cluster. If the aggregated traffic of the candidate cluster does not exceed the BBU capacity, we calculate its utilization rate as Equation 15. Otherwise, we punish the candidate cluster with a *log* function (Equation 16) to avoid forming a cluster that no BBU can handle. Finally, we assign the maximum possible utilization rate to the candidate cluster with Equation (14).

Then, we define the *handover gain* of adding node v_i to cluster c_k as

$$gain\_h(v_i, c_k) = \frac{2\sum_{v_k \in c_k} |e_{i,k}| + \sum_{v_{k_i} \in c_k} \sum_{v_{k_j} \in c_k} |e_{k_i,k_j}|}{2|E|}, \tag{17}$$

which is a measurement of how strong the nodes in the new cluster $\{v_i\} \cup c_k$ are connected to each other. Finally, we define the *gain function* as the combination of both the utilization gain and the handover gain

$$gain(v_i, c_k) = \lambda gain\_u(v_i, c_k) + (1 - \lambda) gain\_h(v_i, c_k), \tag{18}$$

where $\lambda \in (0, 1)$ controls the weight of the utilization and handover gains. In practice, the operators can adjust λ to obtain different RRH-BBU mapping scheme with different BBU utilization and RRH handover optimization objectives.

Convergence: the RCLP algorithm is said to have converged if any further execution of the algorithm yields the same state [64]. Despite the simplicity of label propagation algorithms, there has been very little formal analysis of its convergence, since the variations of graph structures may lead to complicated behaviors of such a greedy algorithm [64]. In our problem, the graph structure can be identified as a clustered Erdös-Rényi graph, and [64] have proved that a label propagation algorithm can correctly and quickly identifies its community structure.

Time Complexity: for each iteration of the RCLP algorithm, it first takes $O(|V|)$ steps for node permutation, and then processes all the links to compute the gain function, taking $O(|E|)$ steps in the worst case. In total, the time complexity of our algorithm is $O(|E|)$. As the handover pairs are quite sparse ($O(|E|) \approx O(|V|)$), the complexity can be nearly $O(|V|)$.

Optimality: RCLP is a greedy algorithm and may not necessarily obtain the optimal solutions [64]. Fortunately, the objective function of problem **P1** is a monotone and

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
Dataset Period	20 weeks 12/05/2011 -04/22/2012	50 weeks 01/07/2013 -12/22/2013
Average call duration	5.18 minutes	6.82 minutes
Handover per hour	78,662	113,082

submodular set function.[3] According to the theory of submodular function maximization [65], such a greedy algorithm can achieve a worst-case approximation bound of $(1 - e^{-1})$ to the optimal. To further improve the robustness of approximation, we exploit a Monte Carlo method [66] to approach the optimal RRH-BBU mapping scheme. Specifically, for each RRH-BBU mapping task, we simultaneously ran the RCLP algorithms on N_w distributed workers of a computation cluster, and repeated each task for Γ_w times. Finally, we collected the formed clustering schemes and select the one with the highest frequency.

6 Evaluation

6.1 Dataset Description

We exploit two large-scale anonymized datasets released by Orange S.A. in the D4D challenges [29], [30]. Specifically, we extract two city-scale datasets for *Abidjan* and *Dakar*, the two largest cities in Ivory Coast and Senegal, respectively. We perform data cleansing to remove missing and incomplete data. Particularly, we exclude base stations with no traffic or handover records, and compile two datasets containing the base station positions, call durations, and handover counts. The details of these two datasets are listed in Table 1.

We assume the C-RAN architecture is deployed in the two cities during the data collection. Specifically, the RRHs are placed in the base station sites, and the centralized BBU pools are deployed and connected to the RRHs via high speed optical fiber. We quantify the RRH traffic based on the aggregated radio resource units [8] allocated to the phone calls, which is proportional to the total call durations in each hour.[4] Due to privacy concerns, the mobility data is randomly sampled from a portion of Orange customers (1 percent for Ivory Coast and 3.33 percent for Senegal, respectively)[29], [30], therefore we estimate the actual handover count by multiplying the sample rate.

6.2 Evaluation on Prediction Accuracy

Evaluation Plan: we use 70 percent of the datasets for the model training, and the remaining 30 percent for testing. For each part, we use the first N_s time steps (hours) as input and predict for the next one time step (hour). We then use the training set to train the MuLSTM model. Since the traffic and handover patterns are quite different during weekdays and weekends, we separately train a weekday and a weekend model using the corresponding datasets, respectively. We implement the LSTM layers using an encoder-decoder architecture. Specifically, the encoder layer L_1 contains $N_{encoder}$ memory units, which accepts as input an array of traffic or handover vectors of N_s time steps, and outputs an encoded sequence for the decoder. The decoder contains $N_{decoder}$ memory units, which accepts as input the encoded sequence and outputs the traffic or handover forecast.

Model Training: we use the popular Tensorflow[5] library for constructing our deep-learning model. We train the MuLSTM model for N_{iter} iterations to ensure that the network learns the potential temporal and spatial structures of the traffic and handover patterns. Based on a series of empirical experiments, we choose the optimal $N_s = 12$ hours, $N_{encoder} = N_{decoder} = 32$, and $N_{iter} = 10,000$. The model is trained on a 64-bit server with an NIVIDA GeForce GTX 1080 graphic card and 16 GB of RAM. Each training iteration takes about 1.5 seconds and the whole process takes 4.2 hours.

Evaluation Metrics: for the model testing phase, we use the trained MuLSTM model to predict the city-wide traffic volume and handover count at the beginning of each hour, and compare the results with the ground truth data. For RRH traffic prediction, we compare the predicted traffic snapshot $\hat{\mathcal{F}}(:,t)$ with the ground truth data $\mathcal{F}(:,t)$ in the test set of size N_{test}, and calculate the MAPE for each snapshot

$$MAPE_f = \frac{1}{N_{test}} \sum_{t=1}^{N_{test}} \left| \frac{\mathcal{F}(:,t) - \hat{\mathcal{F}}(:,t)}{\mathcal{F}(:,t)} \right| \times 100\%. \quad (19)$$

Similarly, for RRH handover prediction, we compare the predicted handover snapshot $\hat{\mathcal{H}}(:,:,t)$ with the ground truth data $\mathcal{H}(:,:,t)$ in the test set of size N_{test}, and calculate the Mean Absolute Error (MAE) for each snapshot

$$MAPE_h = \frac{1}{N_{test}} \sum_{t=1}^{N_{test}} \left| \frac{\mathcal{H}(:,:,t) - \hat{\mathcal{H}}(:,:,t)}{\mathcal{H}(:,:,t)} \right| \times 100\%. \quad (20)$$

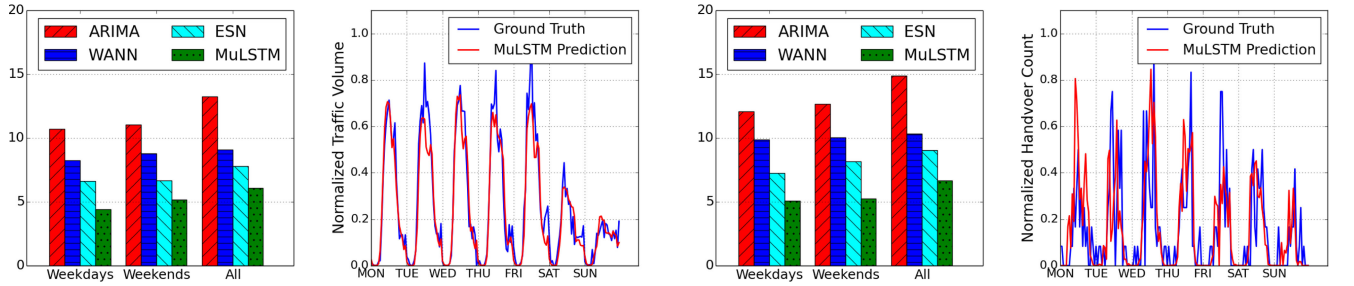
Baseline Methods: we design the following baselines for comparison. The training and validation settings are the same as the proposed method.

- *ARIMA*: this baseline models the traffic of each RRH as a time series, and uses the traditional ARIMA model [35] for traffic prediction. Similarly, it models each significant handover sequence as a time series, and builds individual ARIMA models for handover prediction, respectively.
- *WANN*: this baseline models the RRHs in the network as a whole, and adopts the same architecture as the proposed MuLSTM model except that the predictors are implemented using Windowed-ANN structure [38]. The WANN layers do not have an internal temporal state and thus are not able to

3. For the proof of submodularity, please refer to Appendix I, available as the online supplemental material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TMC.2020.2971470.

4. We note that if fine-grained network traffic data, such as video stream, are available, our solution can easily adapt to the optimization task with regard to each specific traffic type.

5. https://www.tensorflow.org



(a) RRH traffic prediction errors. (b) Downtown traffic example. (c) RRH handover prediction errors. (d) Downtown handover example.

Fig. 5. Evaluation results of RRH traffic and handover prediction. (a) and (c) show the traffic and handover prediction errors of the baselines and the proposed MuLSTM method, respectively. (b) and (d) demonstrate illustrative examples of the traffic patterns and prediction results from a downtown RRH in Abidjan in one week (from 01/09/2012 to 01/15/2012).

model the temporal dependencies among different time steps.

- *ESN*: this baseline models the traffic and mobility dynamics with two echo state networks [44], respectively. Based on similar techniques as [28], we build an ESN with $N_{et} = 3,000$ hidden units (i.e., the reservoir size) for traffic prediction, and $N_{em} = 2,500$ hidden units for mobility prediction.

Evaluation Results: Fig. 5 shows the results of traffic and handover prediction using the baseline and proposed methods, respectively. Each method is evaluated on weekdays, weekends, and all days. In RRH traffic prediction, the proposed MuLSTM method achieves an MAPE of 6.08 percent for all days, which is much lower than the MAPE of ARIMA (13.23 percent) and WANN (9.08 percent) methods. The possible reason is that the ARIMA method models the temporal dependency of RRH traffic, but it fails to capture the correlations among RRHs. Meanwhile, the WANN method models the RRH correlations in the ANN layers, but it is not able to capture the temporal dynamics of RRH traffic. In contrast, the proposed MuLSTM method models the temporal dynamics and spatial correlations simultaneously to achieve lower prediction errors. We also note that the ESN baseline achieves a worse MAPE (7.78 percent) than the proposed method, validating the limitations of the reservoir-based hidden layers in ESN and the benefit of joint training of MuLSTM.

Furthermore, by separately training different predictive models for weekdays and weekends, the prediction errors can be reduced for the ARIMA, ESN, and MuLSTM methods, since the temporal patterns can be modeled in a fine-grained manner for these methods. Similar conclusions can be made for handover prediction with these methods.

We show two illustrative examples of RRH traffic and handover predictions using the proposed MuLSTM models (without weekday and weekend separation) in Figs. 5b and 5d, respectively. The example RRH is located in Plateau, the downtown area of Abidjan, Ivory Coast. We can see that the proposed methods successfully learn the dynamic weekday and weekend patterns in both traffic and handover dynamics, and make accurate predictions based on the temporal and spatial factors.

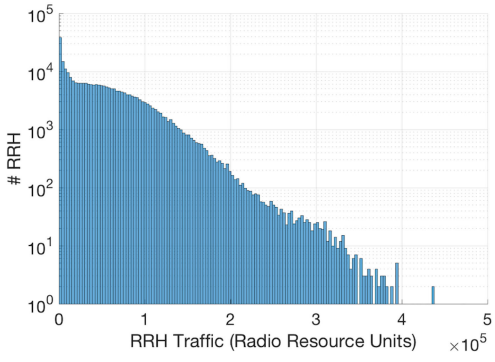


Fig. 6. The histogram of RRH traffic in the training set of Abidjan, measured in radio resource units.

6.3 Evaluation on RRH-BBU Mapping

Because it is difficult to deploy real-world C-RAN networks in the two cities, in this work, we evaluate the effectiveness of the proposed RRH-BBU mapping algorithm via several key metrics. Particularly, we run the mapping algorithms for each hour in the test set and calculate the statistical multiplex gains for comparison.

Parameter Selection: the most important parameter in the RRH-BBU mapping phase is the *BBU size* in the BBU pool. Since BBUs are implemented as virtual machines, their sizes are usually discrete values corresponding to predefined VM configurations (e.g., $PICO, SMALL, MEDIUM, LARGE$). However, the radio resource units occupied by RRHs are continuous. For example, Fig. 6 shows the histogram of the radio resource units of all the RRHs in the training set of Abidjan, which ranges from 1×10^5 to 4×10^5 radio resource units. The desired BBU size needs to accommodate the demands of radio resource units occupied by both single RRH and RRH clusters. Based on previous studies [8], [14] and empirical experiments, we design the BBU size category as a discrete set as follows:

$$\Phi = \{1RU, 2RU, 4RU, 8RU, \ldots\}, \tag{21}$$

where $1RU = 10^5$ radio resource units in this example. In this way, an RRH that occupies 1.5×10^5 radio resource units can allocate a BBU of size $2RU$, while a cluster of RRHs with an aggregated radio resource units of 10.5×10^5 can allocate a BBU of size $16RU$, respectively.

We run the RRH-BBU mapping algorithms on a distributed cluster system via Matlab Parallel Computing Toolbox.[6] We allocate $N_w = 16$ distributed workers and

6. https://www.mathworks.com/products/parallel-computing

TABLE 2
Evaluation Results of the RRH-BBU Mapping Methods

Methods	Abidjan		Dakar	
	BBU Utilization Rate	Inner-BBU Handover Rate	BBU Utilization Rate	Inner-BBU Handover Rate
DIRECT-MAP	58.7%	0%	49.8%	0%
STATIC-MAP	77.9%	35.4%	73.4%	33.2%
UTIL-RCLP	99.3%	1.64%	99.2%	0.77%
HAND-RCLP	60.5%	85.8%	62.1%	83.5%
COALITION	81.8%	80.2%	81.3%	77.9%
DUAL-RCLP (Proposed)	**86.5%**	**85.1%**	**85.2%**	**82.3%**

repeated each task for $\Gamma_w = 10^3$ times. Each RCLP task takes about 0.1 seconds, and finding an RRH-BBU mapping scheme takes 100 seconds.

Evaluation Metrics: for an RRH-BBU mapping scheme that partition the RRHs into a set of N_k disjoint clusters $\mathbb{C} = \{c_1, \ldots, c_{N_k}\}$ and map each cluster c_k to a BBU b_k in the BBU pool $\mathbb{B} = \{b_1, b_2, \ldots, b_k\}$, we evaluate its statistical multiplex gains from the following two aspects.

To evaluate the improvement of BBU utilization, we define the *average utilization rate* of the BBU pool based on Equation (5) as

$$U(\mathbb{C}) = \frac{1}{N_k} \sum_{k=1}^{N_k} U(c_k). \tag{22}$$

Similarly, to evaluate the improvement of handover quality, we define the *cluster handover rate* of the RRH clusters based on Equation (6) as

$$W(\mathbb{C}) = \frac{1}{N_k} \sum_{k=1}^{N_k} W(c_k). \tag{23}$$

Baseline Methods: we design the following baselines to compare with the proposed method.

- *DIRECT-MAP*: this baseline directly maps each RRH to a BBU with maximal utilization rate based on the pool constraints. We note that this method is widely adopted in the traditional RAN architecture [5].
- *STATIC-MAP*: this baseline first clusters RRHs based on the complementarity of their daily traffic profiles, and then statically map clusters to BBUs using the algorithm in **Chen_2017b**. The cluster handover rate is not optimized, and the daily traffic profile is generated based on the average over a long term of observation for each RRH. This baseline is similar to the method proposed in [22].
- *UTIL-RCLP*: this baseline finds mapping schemes that maximize the BBU utilization rate without considering the cluster handover rate. This baseline is similar to the method proposed in [23].
- *HAND-RCLP*: similarly, this baseline finds mapping schemes that maximize the cluster handover rate without considering the BBU utilization rate.
- *COALITION*: this baseline adopts the similar coalitional game approach of [53]. Specifically, we regard each BBU as a coalition and each RRH as a player who transfers among these coalitions. We define the transfer order as follows: for any user (RRH) v_i in a coalition (BBU) c_j, $\{v_i\} \cup c_j$ is preferred over another coalition $\{v_i\} \cup c_k$ (i.e., $c_j i c_k$) if $gain(v_i, c_j) > gain(v_i, c_k)$. Finally, a local optimal solution with Nash-stable partition are solved [53].

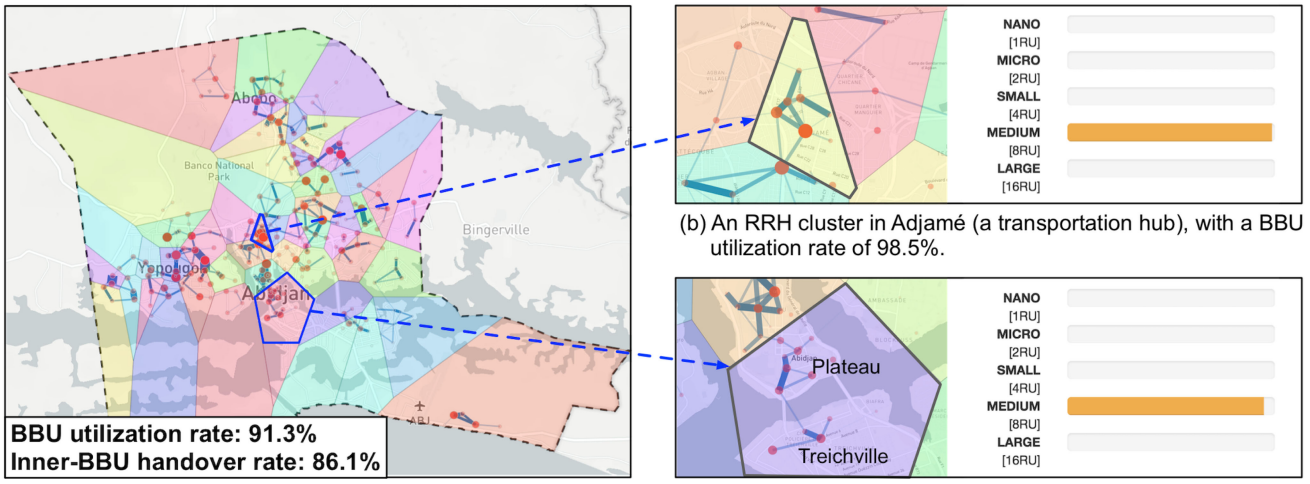
Correspondingly, we name the proposed method as *DUAL-RCLP*, which simultaneously optimizes the BBU utilization rate and the cluster handover rate.

Evaluation Results: Table 2 shows the results of BBU utilization rate and cluster handover rate on the test set using the baseline and proposed methods, respectively. The *DIRECT-MAP* baseline achieves the lowest performance, since each RRH is allocated a BBU without resource sharing and handover optimization. The *STATIC-MAP* shows moderate performance improvements via static clustering, and the *COALIATION* baseline further improves the mapping performance. Note that the *UTIL-RCLP* method achieves relatively high BBU utilization rate (above 99.2 percent), but fails to arrange RRHs with frequent handover events into clusters. In contrast, the *HAND-RCLP* method finds clusters with high cluster handover rate (above 83.5 percent), but these clusters do not utilize the allocated BBUs efficiently (with a utilization rate below 62.1 percent). The proposed *DUAL-RCLP* method achieves a BBU utilization rate above 85.2 percent and an cluster handover rate above 82.3 percent in both cities, validating the effectiveness of our method in finding cost-effective and quality-aware mapping schemes.

6.4 Case Studies

In order to further evaluate the effectiveness of our framework, we conduct a series of case studies in Abidjan and Dakar, respectively. In each case study, we showcase the traffic and handover snapshot in a specific scenario, and present the RRH-BBU mapping results on the map.

Abidjan Rush Hour: we select a typical weekday morning rush hour (9:00–10:00, 04/10/2012) in Abidjan from the test set for a case study. Fig. 7a shows the RRH traffic and handover patterns during the rush hour, where larger dots denote RRHs with higher traffic volume, and thicker lines correspond to more handover events observed between the two corresponding RRHs. We also visualize the RRH-BBU mapping scheme using a Voronoi diagram [67] in Fig. 7a, where each polygon corresponds to a RRH cluster. We can see that during the morning rush hour, the network traffic of the city are mainly generated from the residential areas, the business districts, and the transportation hubs. Correspondingly, the handover events are frequently observed in these areas. Our framework successfully find an RRH-BBU scheme with an average BBU utilization rate of 91.3 percent and an RRH internal handover rate of 86.1 percent.



(a) An RRH-BBU mapping scheme in Abidjan during the morning rush hour (04/10/2012 9:00–10:00).

(b) An RRH cluster in Adjamé (a transportation hub), with a BBU utilization rate of 98.5%.

(c) An RRH cluster in Plateau (a business district) and Treichville (a residential area), with a BBU utilization rate of 94.9%.

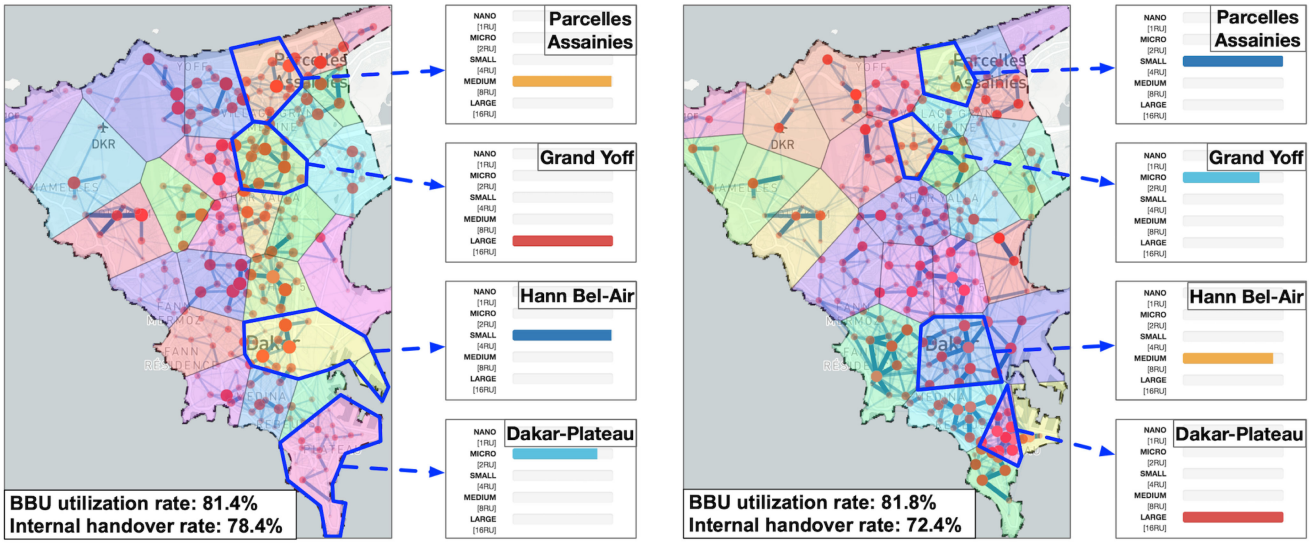
Fig. 7. A case study of the RRH-BBU mapping results during a typical morning rush hour in Abidjan.

Fig. 7b shows the traffic and handover patterns of a cluster in Adjamé, a transportation hub of Abidjan. Adjamé has several important bus stations from where buses serve the greater Abidjan area as well as all of Ivory Coast. In the morning rush hour, large crowds of commuters and long distance travelers arrive at and depart from this area, generating significant handover events among the RRHs in this area, as well as large traffic volume. Our method successfully identifies this RRH cluster and assigns a medium-size BBU ($8RU$) to it, achieving a high BBU utilization rate of 98.8 percent, as shown in Fig. 7b.

Fig. 7c shows a hybrid cluster formed by RRHs in Plateau and Treichville. Plateau is the central business district of Abidjan, and Treichville is one of the most populated suburban residential areas in Abidjan. In the morning rush hour, significant traffic volumes are observed in the RRHs of Plateau and Treichville, respectively, most probably generated by the residents, commuters, and workers in these areas. By sharing a medium-size BBU with $8RU$, the cluster of RRHs in these areas achieves a high BBU utilization rate of 94.9 percent. More importantly, the large volume of handover events between Plateau and Treichville during the rush hour can be processed within the BBU, which significant improves handover quality.

Dakar Independence Day: in Dakar, we investigate the RRH-BBU mapping scheme during the morning hours of the 2013 Senegal Independence Day (04/04/2013 10:00–11:00), and compare it with the scheme during the morning hours of a typical weekday (04/11/2013 10:00–11:00, one week later). Fig. 8 shows the RRH traffic and handover patterns on the two days and the RRH-BBU mapping scheme.

In Senegal, the Independence Day is celebrated as a public holiday. In Fig. 8a, we can see that during the morning hours of that day, most traffic and handover events are generated in the central and northern parts of Dakar, which correspond to the city's residential neighborhoods, restaurants,



(a) An RRH-BBU mapping scheme in Dakar during the morning hours of the Independence Day (04/04/2013 10:00–11:00).

(b) An RRH-BBU mapping scheme in Dakar during the morning hours of a typical weekday (04/11/2013 10:00–11:00).

Fig. 8. A case study of the RRH-BBU mapping results during the morning hours in Dakar on two typical days.

and parks, etc. Consequently, our framework identifies these communities and allocates high-capacity BBUs for the corresponding RRH clusters. For example, Fig. 8a shows two RRH clusters in *Parcelles Assainies* and *Grand Yoff*, two of the largest residential neighborhoods in Dakar, as well as the allocated BBU capacity, respectively. In contrast, the southern parts of the city, including *Hann Bel-Air* and *Dakar-Plateau*, are the central industrial, business and administrative districts of Dakar. On the Independence Day, these areas observe relatively fewer user activities due to public holidays. Consequently, our framework tends to form large clusters consisting of many RRHs to reduce handover cost, while allocating BBUs with relatively small capacities since the aggregated traffic volumes are insignificant. For example, Fig. 8a illustrates two clusters in *Hann Bel-Air* (the port and industrial zone) and *Dakar-Plateau* (the business and administrative center) and the allocated BBUs, respectively. These two clusters occupy large geographic areas with many RRHs, however the small and micro size BBUs are already adequate to process the traffic. In this way, our framework achieves an average BBU utilization rate of 81.4 percent and an internal handover rate of 78.4 percent, respectively.

We also present the RRH-BBU mapping scheme in the morning hours of a typical weekday (one week later) for comparison. From Fig. 8b, we can see that during the weekday morning, a large number of RRHs in the southern parts of the city observe significant traffic volume and handover events. Correspondingly, our framework identifies clusters with densely connected RRHs in *Hann Bel-Air* and *Dakar-Plateau*, and allocate BBUs with high capacities for them. In contrast, the clusters formed in the residential areas (e.g., *Parcelles Assainies* and *Grand Yoff*) do not observe significant user activities, and thus the BBUs allocated to them are of lower capacities. Similarly, our framework effectively increase the average BBU utilization rate to 81.8 percent and achieves an internal handover rate of 72.4 percent, respectively.

In summary, by adaptively forming different sizes of clusters and allocating BBUs with adequate capacities, our framework effectively improves the BBU utilization rate and handover performance in the C-RAN architecture.

7 Conclusion

In this work, we propose a data-driven approach for C-RAN optimization, considering both the traffic and mobility dynamics for RRH-BBU mapping. We extract traffic volume and handover count from large-scale CDR datasets, and propose a deep-learning-based model to accurately predict the traffic and handover patterns. We formulate the RRH-BBU mapping with cost and quality objectives as a set partition problem, and propose a heuristic greedy algorithm to effectively find the robust approximation to the optimal schemes under resource constraints. Evaluations on large-scale CDR datasets validate the effectiveness of our framework, which outperforms the traditional RAN architectures and state-of-the-art baselines.

In the future, we plan to evaluate our framework on datasets with richer traffic and mobility information. We also plan to incorporate contextual factors (e.g., social events) to further improve prediction accuracy, and explore mapping algorithms in multi-pool architectures (e.g., Fog-RAN and Mobile Edge Cloud).

Acknowledgments

This work was supported by the NSF of China No. 61802325, NSF of Fujian Province No. 2018J01105, and the China Fundamental Research Funds for the Central Universities No. 20720170040.

References

[1] Ericsson, "Ericsson mobility report 2018," Stockholm, Sweden, Tech. Rep., 2018.
[2] Cisco, "Global mobile data traffic forecast update, 2016–2021," Cisco, San Jose, CA, USA, Tech. Rep., 2016.
[3] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, and W. Xiang, "Big data-driven optimization for mobile networks toward 5G," *IEEE Netw.*, vol. 30, no. 1, pp. 44–51, Jan./Feb. 2016.
[4] J. Research, "Mobile operator business models: Challenges, opportunities & adaptive strategies 2011–2016," Juniper Research, New York, Tech. Rep., 2011.
[5] A. Checko *et al.*, "Cloud RAN for mobile networks—A technology overview," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 405–426, Firstquarter 2015.
[6] C. M. R. Institute, "C-RAN: The road toward Green RAN," China Mobile Research Institute, Beijing, China, Tech. Rep., 2011.
[7] N. Nikaein, "Processing radio access network functions in the cloud: Critical issues and modeling," in *Proc. 6th Int. Workshop Mobile Cloud Comput. Services*, 2015, pp. 36–43.
[8] H. Taleb, M. E. Helou, K. Khawam, S. Lahoud, and S. Martin, "Centralized and distributed RRH clustering in Cloud Radio Access Networks," in *Proc. IEEE Symp. Comput. Commun.*, 2017, pp. 1091–1097.
[9] J. G. Andrews *et al.*, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.
[10] C. L. I, C. Rowell, S. Han, Z. Xu, G. Li, and Z. Pan, "Toward green and soft: A 5G perspective," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 66–73, Feb. 2014.
[11] K. Boulos, M. E. Helou, and S. Lahoud, "RRH clustering in cloud radio access networks," in *Proc. Int. Conf. Appl. Res. Comput. Sci. Eng.*, 2015, pp. 1–6.
[12] Y. S. Chen, W. L. Chiang, and M. C. Shih, "A dynamic BBU-RRH mapping scheme using borrow-and-lend approach in cloud radio access networks," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1632–1643, Jun. 2018.
[13] M. Khan, R. S. Alhumaima, and H. S. Al-Raweshidy, "QoS-aware dynamic RRH allocation in a self-optimized cloud radio access network with RRH proximity constraint," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 3, pp. 730–744, Sep. 2017.
[14] S. C. Zhan and D. Niyato, "A coalition formation game for remote radio head cooperation in cloud radio access network," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 1723–1738, Feb. 2017.
[15] D. Zhang, B. Guo, and Z. Yu, "The emergence of social and community intelligence," *Computer*, vol. 44, no. 7, pp. 21–28, 2011.
[16] A. Furno, M. Fiore, and R. Stanica, "Joint spatial and temporal classification of mobile traffic demands," in *Proc. IEEE Conf. Comput. Commun.*, 2017, pp. 1–9.
[17] A. Furno, D. Naboulsi, R. Stanica, and M. Fiore, "Mobile demand profiling for cellular cognitive networking," *IEEE Trans. Mobile Comput.*, vol. 16, no. 3, pp. 772–786, Mar. 2017.
[18] Y. Lin, L. Shao, Z. Zhu, Q. Wang, and R. K. Sabhikhi, "Wireless network cloud: Architecture and system requirements," *IBM J. Res. Develop.*, vol. 54, no. 1, pp. 4:1–4:12, 2010.
[19] V. N. Ha and L. B. Le, "End-to-end network slicing in virtualized OFDMA-based cloud radio access networks," *IEEE Access*, vol. 5, pp. 18 675–18 691, 2017.
[20] Y. Sun, M. Peng, and H. V. Poor, "A distributed approach to improving spectral efficiency in uplink device-to-device-enabled cloud radio access networks," *IEEE Trans. Commun.*, vol. 66, no. 12, pp. 6511–6526, Dec. 2018.
[21] C. Pan, H. Ren, M. Elkashlan, A. Nallanathan, and L. Hanzo, "Weighted sum-rate maximization for the ultra-dense user-centric TDD C-RAN downlink relying on imperfect CSI," *IEEE Trans. Wireless Commun.*, vol. 18, no. 2, pp. 1182–1198, Feb. 2019.

[22] S. Bhaumik *et al.*, "CloudIQ: A framework for processing base stations in a data center," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 125–136.
[23] S. Namba, T. Matsunaka, T. Warabino, S. Kaneko, and Y. Kishi, "Colony-RAN architecture for future cellular network," in *Proc. Future Netw. Mobile Summit*, 2012, pp. 1–8.
[24] J. Zhang, Y. Ji, X. Xu, H. Li, Y. Zhao, and J. Zhang, "Energy efficient baseband unit aggregation in cloud radio and optical access networks," *J. Opt. Commun. Netw.*, vol. 8, no. 11, pp. 893–901, 2016.
[25] S. Tekinay and B. Jabbari, "Handover and channel assignment in mobile cellular networks," *IEEE Commun. Mag.*, vol. 29, no. 11, pp. 42–46, Nov. 1991.
[26] D. Naboulsi, A. Mermouri, R. Stanica, H. Rivano, and M. Fiore, "On user mobility in dynamic cloud radio access networks," in *Proc. IEEE Conf. Comput. Commun.*, 2018, pp. 1–9.
[27] K. Dimou *et al.*, "Handover within 3GPP LTE: Design principles and performance," in *Proc. IEEE 70th Veh. Technol. Conf. Fall*, 2009, pp. 1–5.
[28] M. Chen, W. Saad, C. Yin, and M. Debbah, "Echo state networks for proactive caching in cloud-based radio access networks with mobile users," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3520–3535, Jun. 2017.
[29] V. D. Blondel *et al.*, "Data for development: The D4D challenge on mobile phone data," 2012, *arXiv:1210.0137*.
[30] Y.-A. de Montjoye, Z. Smoreda, R. Trinquart, C. Ziemlicki, and V. D. Blondel, "D4D-senegal: The second mobile phone data for development challenge," 2014, *arXiv:1407.4885 [physics]*.
[31] G. Barlacchi *et al.*, "A multi-source dataset of urban life in the city of milan and the province of trentino," *Sci. Data*, vol. 2, 2015, Art. no. 150055.
[32] B. Guo *et al.*, "Mobile crowd sensing and computing: The review of an emerging human-powered sensing paradigm," *ACM Comput. Surv.*, vol. 48, no. 1, pp. 1–31, 2015.
[33] L. Wang, D. Zhang, Y. Wang, C. Chen, X. Han, and A. M'hamed, "Sparse mobile crowdsensing: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 161–167, Jul. 2016.
[34] G. M. Weiss, "Data mining in telecommunications," in *Data Mining and Knowledge Discovery Handbook*. Boston, MA, USA: Springer, 2005, pp. 1189–1201.
[35] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*. Hoboken, NJ, USA: Wiley, 2015.
[36] S. Ntalampiras and M. Fiore, "Forecasting mobile service demands for anticipatory MEC," in *Proc. IEEE 19th Int. Symp. "A World Wireless Mobile Multimedia Netw."*, 2018, pp. 14–19.
[37] G. P. Zhang and M. Qi, "Neural network forecasting for seasonal and trend time series," *Eur. J. Oper. Res.*, vol. 160, no. 2, pp. 501–514, 2005.
[38] C. Freeman, R. D. Dony, and S. M. Areibi, "Audio environment classication for hearing aids using artificial neural networks with windowed input," in *Proc. IEEE Symp. Comput. Intell. Image Signal Process.*, 2007, pp. 183–188.
[39] L. Nie, D. Jiang, L. Guo, and S. Yu, "Traffic matrix prediction and estimation based on deep learning in large-scale IP backbone networks," *J. Netw. Comput. Appl.*, vol. 76, pp. 16–22, 2016.
[40] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," *ArXiv e-prints*, vol. 1511, 2015, Art. no. arXiv:1511.05298.
[41] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1655–1661.
[42] C. Zhang and P. Patras, "Long-term mobile traffic forecasting using deep spatio-temporal neural networks," in *Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2018, pp. 231–240.
[43] J. Wang *et al.*, "Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach," in *Proc. IEEE Conf. Comput. Commun.*, 2017, pp. 1–9.
[44] M. Lukoševičius, "A practical guide to applying echo state networks," in *Neural Networks: Tricks of the Trade: Second Edition*. Berlin, Germany: Springer, 2012, pp. 659–686.
[45] L. Chen *et al.*, "Dynamic cluster-based over-demand prediction in bike sharing systems," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, 2016, pp. 841–852.
[46] D. Naboulsi, R. Stanica, and M. Fiore, "Classifying call profiles in large-scale mobile traffic datasets," in *Proc. IEEE Conf. Comput. Commun.*, 2014, pp. 1806–1814.
[47] B. Cici, M. Gjoka, A. Markopoulou, and C. T. Butts, "On the decomposition of cell phone activity patterns and their connection with urban ecology," in *Proc. 16th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2015, pp. 317–326.
[48] E. Balas and M. Padberg, "Set partitioning: A survey," *SIAM Rev.*, vol. 18, no. 4, pp. 710–760, 1976.
[49] R. S. Garfinkel and G. L. Nemhauser, "The set-partitioning problem: Set covering with equality constraints," *Oper. Res.*, vol. 17, no. 5, pp. 848–856, 1969.
[50] K. Hoffman and M. Padberg, "Set covering, packing and partitioning problems," in *Encyclopedia of Optimization*. Boston, MA, USA: Springer, 2001, pp. 2348–2352.
[51] J. Yao and N. Ansari, "QoS-aware joint BBU-RRH mapping and user association in cloud-RANs," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 4, pp. 881–889, Dec. 2018.
[52] M. Khan, Z. H. Fakhri, and H. S. Al-Raweshidy, "Semistatic cell differentiation and integration with dynamic BBU-RRH mapping in cloud radio access network," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 1, pp. 289–303, Mar. 2018.
[53] W. Saad, Z. Han, R. Zheng, M. Debbah, and H. V. Poor, "A college admissions game for uplink user association in wireless small cell networks," in *Proc. IEEE Conf. Comput. Commun.*, 2014, pp. 1096–1104.
[54] J. Wu, Z. Zhang, Y. Hong, and Y. Wen, "Cloud radio access network (C-RAN): A primer," *IEEE Netw.*, vol. 29, no. 1, pp. 35–41, Jan./Feb. 2015.
[55] F. Simini, M. C. González, A. Maritan, and A.-L. Barabási, "A universal model for mobility and migration patterns," *Nature*, vol. 484, no. 7392, pp. 96–100, 2012.
[56] T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.
[57] D. P. Mandic and J. Chambers, *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability*. New York, NY, USA: Wiley, 2001.
[58] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 1310–1318.
[59] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[60] J. M. Ver Hoef and P. L. Boveng, "Quasi-poisson vs. negative binomial regression: How should we model overdispersed count data?" *Ecology*, vol. 88, no. 11, pp. 2766–2772, 2007.
[61] S. Banerjee, M. Jenamani, and D. K. Pratihar, "A survey on influence maximization in a social network," 2018, *arXiv:1808.05502*.
[62] K. Andreev and H. Räcke, "Balanced graph partitioning," in *Proc. 16th Annu. ACM Symp. Parallelism Algorithms Architectures*, 2004, pp. 120–124.
[63] M. Ciglan and K. Nørvåg, "Fast detection of size-constrained communities in large networks," in *Proc. Int. Conf. Web Inf. Syst. Eng.*, 2010, pp. 91–104.
[64] K. Kothapalli, S. V. Pemmaraju, and V. Sardeshmukh, "On the analysis of a label propagation algorithm for community detection," in *Proc. Int. Conf. Distrib. Comput. Netw.*, 2013, vol. 7730, pp. 255–269.
[65] S. Fujishige, *Submodular Functions and Optimization*, vol. 58. Amsterdam, Netherlands: Elsevier, 2005.
[66] C. Z. Mooney, *Monte Carlo Simulation*. Thousand Oaks, CA, USA: SAGE Publications, 1997.
[67] Okabe Atsuyuki, "Spatial tessellations," in *International Encyclopedia of Geography*, Hoboken, NJ, USA: Wiley, 2016.

Longbiao Chen (Member, IEEE) received the PhD degree in computer science from Zhejiang University, China, in 2016 and Sorbonne University, France, in 2018. He is currently an assistant professor with Xiamen University, China. His research interests include mobile computing, big data analytics, crowdsensing and urban computing. He has published more than 30 papers in top-tier journals and conferences, including the *IEEE Transactions on Intelligent Transportation Systems*, *IEEE Transactions on Human-Machine Systems*, *Journal of Network and Computer Applications*, and ACM UBICOMP. He received two UBICOMP Honorable Mention awards, in 2015 and 2016. He is also a technical committee member of ACM SIGSPATIAL China, and serves as the PC members of IJCAI and UIC.

Thi-Mai-Trang Nguyen (Member, IEEE) received the PhD degree in computer science from the University of Paris 6, France, in 2003. She is currently an associate professor at University Pierre and Marie Curie (Paris 6) and doing research at Laboratoire d'Informatique de Paris 6 (LIP6), France. The PhD thesis was co-supervised and carried-out at Ecole Nationale Superieure des Telecommunications (ENST-Paris). From 2004 to 2006, She was a postdoctoral researcher at France Telecom in Rennes, France and at the University of Lausanne, Switzerland. Her research interests include network architecture, network protocol design, and network data analytics.

Dingqi Yang (Member, IEEE) received the BSc degree in instrumentation engineering from Xidian University, Xi'an, China, in 2009 the ME degree in telecommunication network from TELECOM SudParis, in 2011, and the PhD degree from Pierre and Marie Curie University and Institut Mines-TELECOM/TELECOM SudParis, in 2015, where he won both the Doctorate Award and the Institut Mines-TELECOM Press Mention. He is currently a senior researcher in the University of Fribourg, Switzerland. His research interests include big social media data analytics, ubiquitous computing, location based services, and smart city applications.

Michele Nogueira (Member, IEEE) received the doctorate degree in computer science from the University Pierre et Marie Curie—Sorbonne Universités, Laboratoire d'Informatique de Paris VI (LIP6), in 2009. She is currently a professor of computer science at the Federal University of Paraná, Curitiba, Brazil, where she has been since 2010. She was a visiting researcher at Georgia Institute Technology (GeorgiaTech) and a visiting professor at University Paul Sabatier, in 2009 and 2013, respectively. Her research interests include wireless networks, security and dependability. She has been a recipient of Academic Scholarships from Brazilian Government on her undergraduate and graduate years, and of international grants such as from the ACM SIGCOMM Geodiversity program. She is also associate technical editor for the *IEEE Communications Magazine* and the *Journal of Network and Systems Management*.

Cheng Wang (Member, IEEE) received the PhD 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 data analysis, and multisensor fusion.

Daqing Zhang (Fellow, IEEE) received the PhD degree from the University of Rome "La Sapienza", Italy, in 1996. He is currently a full professor at Telecom SudParis, Institut Mines-Telecom, France. He has published more than 200 technical papers in leading conferences and journals, where his work on context model is widely accepted by the pervasive computing, mobile computing and service-oriented computing communities. His research interests include context-aware computing, urban computing, mobile computing, big data analytics, pervasive elderly care, etc.