An International ICT R&D Journal Sponsored by ZTE Corporation

ISSN 1673-5188
CN 34-1294/ TN
CODEN ZCTOAK

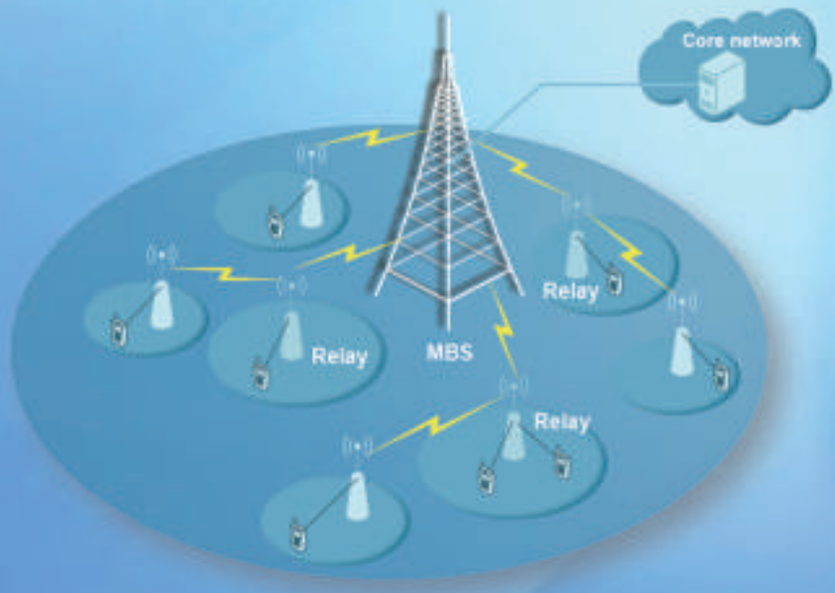
ZTE COMMUNICATIONS

中兴通讯技术(英文版)

tech.zte.com.cn

June 2018, Vol. 16 No. 2

SPECIAL TOPIC:
Ultra-Dense Networking Architectures and Technologies for 5G



9 771673 518185 06

ZTE中兴 ZTE COMMUNICATIONS 中兴通讯技术(英文版)

VOLUME 16 NUMBER 2 JUNE 2018

ZTE Communications Editorial Board

CONTENTS

Special Topic: Ultra-Dense Networking Architectures and Technologies for 5G

CONTENTS

ZTE COMMUNICATIONS
Vol. 16 No. 2 (Issue 62)
Quarterly
First English Issue Published in 2003

Supervised by:
Anhui Science and Technology Department

Sponsored by:
Anhui Science and Technology Information Research Institute;
Magazine House of ZTE Communications

Published and Circulated (Home and Abroad) by:
Magazine House of ZTE Communications

Staff Members:
Editor-in-Chief: CHEN Jie
Executive Associate Editor-in-Chief: HUANG Xinming
Editor-in-Charge: ZHU Li
Editors: XU Ye and LU Dan
Producer: YU Gang
Circulation Executive: WANG Pingping
Assistant: WANG Kun

Editorial Correspondence:
Add: 12F Kaixuan Building,
329 Jinzhai Road,
Hefei 230061, China
Tel: +86-551-65533356
Fax: +86-551-65850139
Email: magazine@zte.com.cn

Printed by:
Hefei Tiancai Color Printing Company

Publication Date:
June 25, 2018

Publication Licenses:
ISSN 1673-5188
CN 34-1294/ TN

Annual Subscription:
RMB 80

DOI: 10.3969/j.issn.1673-5188.2018.02.001
http://kns.cnki.net/kcms/detail/34.1294.TN.20180507.1433.002.html, published online May 7, 2018

Editorial

Ultra-Dense Networking Architectures and Technologies for 5G

► Guest Editors

Victor C. M. Leung is a professor of electrical and computer engineering and holder of the TELUS Mobility Research Chair at the University of British Columbia (UBS), Canada. He has co-authored more than 1000 technical papers in the areas of wireless networks and mobile systems, in addition to 37 book chapters and 12 book titles. He is Fellow of the Royal Society of Canada, the Canadian Academy of Engineering and the Engineering Institute of Canada. He is also a Fellow of IEEE.

ZHANG Haijun is currently a full professor at University of Science and Technology Beijing, China. He was a Postdoctoral Research Fellow at Department of Electrical and Computer Engineering, the University of British Columbia (UBC), Canada. He serves as editors of *IEEE Transactions on Communications* and *IEEE 5G Tech Focus* and leading guest editors for *IEEE Communications Magazine* and *IEEE TETC*. He received the IEEE ComSoc Young Author Best Paper Award in 2017. He is a Senior Member of IEEE.

With the continuous enrichment of mobile communication application scenarios in the future, the traditional macro-cellular-based mobile communication network architecture will be difficult to meet the explosive growth in demand for communications services. A promising solution is the deployment of ultra dense networks (UDNs) comprising flexibly deployed low-power small base stations (BSs), such as microcell BSs, picocell BSs and femtocell BSs. In 5G, ultra dense networks that are deployed with low-cost and low-power small cells are expected to enhance the overall performance of the network in terms of energy efficiency and load balancing. The essence of ultra dense cell deployment is to shorten the physical distance between the transmitter and the receiver, so as to improve the performance of the system.

For this feature topic, academic and industrial researchers have been invited to discuss technical challenges and recent results related to future mobile networks employing UDNs. After a rigorous review process, five papers have been selected for inclusion of this feature topic.

In the first article, LI Tongxin et al. discuss the intrinsic features and potential benefits of unmanned aerial vehicles (UAVs) and introduce the architecture of a multi-layer heterogeneous wireless network (MHetNet), in which traditional wireless network is assisted by UAVs. Then, an explicit discussion on the factors that limit the performance of MHetNet is presented. Simulations show that the altitude of UAV is a limiting factor that should be optimized to improve the spatial throughput (ST) of MHetNet.

The second article by JIN Yaqi et al. proposes a multi-QoS guaranteed resource allocation algorithm for multi-services based on opportunity cost. This algorithm can achieve a well-done balance between user satisfaction and system fairness. The authors first formulate a unified utility function with effective capacity to describe the multi-QoS metrics of different services. Then they introduce the theory of opportunity cost in economy to form the concept of opportunity cost applying. Finally, the simulation results show that the algorithm can achieve superior overall user satisfaction.

In the third article, FENG Hong et al. propose a wireless backhaul algorithm to find an effective backhaul method for the densely-deployed small base stations (SBSs) and to maximize the energy efficiency of the system. They put forward adaptive backhaul methods of indirect and direct modes. At the same time, the algorithm allocates network resources, including the power of SBSs and system bandwidth, to solve the serious interference problem in UDNs. Simulation results show that the proposed wireless backhaul algorithm has desired performance to achieve higher energy efficiency with a required data rate.

There are transport networks with different performance that varies from high transport latency to low transport latency in the real deployment; in order to cater for these various types of transport networks and realize multi-vendor centralized unit and distributed unit (CU-DU) operation, the radio access network (RAN) architecture in New Radio (NR) is split into two kinds of entities, i.e., CU and DU. In the

Editorial

Victor C. M. Leung and ZHANG Haijun

fourth article, GAO Yin et al. introduce the general aspects of CU-DU split architecture, including the split method, interface function, mobility scenarios and other CU-DU related issues.

In the fifth article, GAO Tengshuang et al. propose a linear interpolation method for time-division duplex coordinated multiple point transmission (TDD-CoMP) systems to estimate the uplink channel at the receiver, which can reduce the channel difference caused by time delay and decrease the probability of codeword mismatch between both sides. Moreover, to mitigate severe inter-cell interference and increase the coverage and throughput of cell-edge users in UDN, a two-codebook scheme is used to strengthen cooperation between base stations (BSs). Simulations show that the proposed scheme can significantly improve the link performance compared to the global precoding scheme.

This special issue covers the network architectures, key technologies, challenges, and methodologies of UDN, and it gathers the researchers of the related areas to analysis the future development of UDN in detail.

Call for Papers

ZTE Communications Special Issue on

Data Intelligence

The new ear of AI is brought about by three converging forces: the advance of AI algorithms, the availability of big data, and the increasing popularity of high performance computing platforms. Data-driven intelligence, or data intelligence, is a new form of AI technologies that leverages the power of big data. It is becoming an extremely active research area with broad area of applications such as computer vision, medial and healthy, intelligent transportation system, multimedia system, and social network. With the huge volume of data available in various domains, big data brings opportunities to boost the performance of artificial intelligent system with advanced machine learning especially deep learning techniques. On the other hand, it also presents unprecedented challenges to manage and exploit big data for a variety of applications. This special issue seeks original articles describing development, relevant trends, challenges, and current practices in the field of big data and artificial intelligence. Position papers, and case studies are also welcome.

Appropriate topics include, but are not limited to,

- Computer vision with big data
- Big medial data
- Big transportation data
- Deep learning for big data
- Applications of big data intelligence
- Semantic of heterogeneous data

Guest Editors

- XU Cheng-zhong, Wayne State University (USA)
- QIAO Yu, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences (China)

Important Dates

- Submission Due: August 1, 2018
- Review and Final Decision Due: September 15, 2018
- Final Manuscript Due: October 1, 2018
- Publication Date: December 25, 2018

Manuscript Preparation

Manuscripts must be typed in English and submitted electronically in MS Word (or compatible) format. The word length is approximately 3000 to 8000, and no more than 8 figures or tables should be included. Authors are requested to submit mathematical material and graphics in an editable format.

Online Submission

Please submit your manuscript through the online submission system of the journal: https://mc03.manuscriptcentral.com/ztecom

DOI: 10.3969/j.issn.1673-5188.2018.02.002
http://kns.cnki.net/kcms/detail/34.1294.TN.20180517.1012.002.html, published online May 17, 2018

UAV Assisted Heterogeneous Wireless Networks: Potentials and Challenges

LI Tongxin, SHENG Min, LYU Ruiling, LIU Junyu, and LI Jiandong
(State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an, Shanxi 710071, China)

Abstract

By fully exploiting the spatial resources, unmanned aerial vehicles (UAVs) are expected to serve as an efficient complementary to terrestrial wireless communication system to provide enhanced coverage and reliable connectivity to ground users. With the growing deployment of units such as small cell base stations (BSs), however, the incurred severe interference may hinder the potential benefits of the integration of UAVs. In this paper, we first discuss the intrinsic features and potential benefits of UAVs and introduce the architecture of multi-layer heterogeneous wireless network (MHetNet), in which traditional wireless network is assisted by UAVs. Then, an explicit discussion on the factors that limit the performance of MHetNet is presented, including the UAV topology, UAV density, and spectrum sharing of UAV and terrestrial networks. We use simulation results to investigate the performance of MHetNet in terms of spatial throughput (ST). It is shown that, together with the densities of UAV and terrestrial networks, the altitude of UAV is a limiting factor that should be optimized to improve the ST of MHetNet.

Keywords

unmanned aerial vehicles; heterogeneous network; ultra-dense network

1 Introduction

Providing massive device connectivity, enormous network capacity and mega user experienced data rates is one of the aggressive targets of the fifth generation (5G) wireless communications systems. In particular, it is forecasted that the requirement of global mobile data traffic in 2030 will show a 20,000 fold increase compared to that in 2020, and device connections will reach 100 billion [1]. Among the appealing approaches to achieve the ambitious goals, network densification with the deployment of heterogeneous infrastructures has been shown to be the one with the greatest potential [2]. Especially, a growing number of small cells, such as picocells and femtocells, have been deployed to provide high-speed service and boost network capacity. In consequence, spectrum resources could be more effectively exploited and network capacity be significantly enhanced.

While it is reported that network densification with heterogeneous deployments could lead to tremendous network capacity gain [3], [4], an ultra-dense deployment of small cells is not a cost-effective strategy due to capital expense (CAPEX) and operating expense (OPEX) issues. In particular, the quality of service (QoS) can hardly be met with insufficient deployment of small cells. At the same time, recent developments in unmanned aerial vehicles (UAVs) bring forward the idea of using UAVs for coverage extension and capacity enhancements [5]. Due to high mobility and low cost, it could serve as an efficient and flexible complementary to terrestrial heterogeneous wireless networks (HetNet), especially when users' behavior, density, and requirements keep rapidly changing in time and space [6]. The formed new architecture, termed multi-layer heterogeneous wireless network (MHetNet), is promising to provide better terrestrial coverage, enhanced network capacity, and scalable network architecture. For instance, UAVs are capable of providing wireless connectivity to users when the existing terrestrial networks fail to operate or satisfy the demand of wireless connections [7].

However, constantly increasing the density of terrestrial base stations (BSs) or UAVs would not always improve network capacity. Consensus has been recently reached in academia that over-deployment of small cell BSs would incur unexpected and overwhelming interference as well, which conversely results in high transmission outage and degraded user experience [8], [9]. Worse still, it is shown that network capacity would degenerate to be zero with the growing deployment of small cell BSs in ultra-dense networks [10], [11]. Therefore, overwhelming interference caused by deploying UAVs into ultra-dense heterogeneous networks would hinder the application of UAVs, which might conceal the merits and further wors-

en the performance of current terrestrial heterogeneous networks. Therefore, the integration of UAVs into terrestrial network should consider such factors as the density of BSs and the altitude of UAVs.

In this article, we discuss the characteristics of MHetNet by presenting a comparison of traditional terrestrial HetNet and MHetNet in Section 2, aiming to highlight the potential benefits of MHetNet. Afterward, the challenges posed by integrating UAVs into HetNet are elaborated in Section 3, followed by simulation results, which are presented to demonstrate the pros and cons of MHetNet in Section 4. Finally, a conclusion is drawn in Section 5.

2 Multi-Layer Heterogeneous Wireless Networks

In this section, we introduce the architecture of MHetNet by comparing the inherent features of MHetNet with traditional terrestrial wireless networks. Following that, the potential benefits of MHetNet are highlighted.

It can be seen from **Fig. 1** that, with the aid of UAVs, terrestrial HetNet could offload traffic to UAVs especially when user density is large (e.g., the left part in Fig. 1). In consequence, the users who fail to get service from terrestrial HetNet would be alternatively served through connecting to UAVs. The right part of Fig. 1 shows that the UAV network can provide additional coverage for ground users who are severely blocked by buildings. Therefore, the QoS of the ground users could be enhanced with the integration of UAVs in MHetNet. Due to the inherent features of MHetNets, there exist a number of differences between MHetNet and the traditional HetNet, the details of which are discussed in the following.

2.1 Dominant Features of MHetNet

MHetNet has good mobility. For traditional terrestrial wireless networks, in which access points such as BSs are basically deployed in a fixed manner, the mobility of access points is not supported. This means that limited users could be served by each access point even when techniques like cell splitting and merging are applied. In MHetNet, however, UAVs could move randomly or be organized as a swarm. Therefore, compared to traditional HetNet, MHetNet is capable of attaining rapidly-changing user demand [12].

Topology in traditional terrestrial wireless network is basically determined. Even considering the high- mobility vehicle-to-vehicle (V2V) scenarios, the network topology in successive time instants is almost identical. In contrast, topology variation is more straightforward and easier in MHetNet. If serving as access points, UAVs could adaptively adjust positions according to user demand. On the other hand, due to power limitation and malfunction, the UAV positions would be changed frequently and UAV links would form and vanish repeatedly. The topology change of UAV network would significantly influence the performance of terrestrial network. For instance, assuming that UAV access points move rapidly, the terrestrial users would be handed over to other terrestrial BSs or UAV access points, which may lead to substantial overhead. In addition, if no UAV access points serve as the backups and all the traffic is transferred to terrestrial BSs, traffic congestion and overload would be incurred.

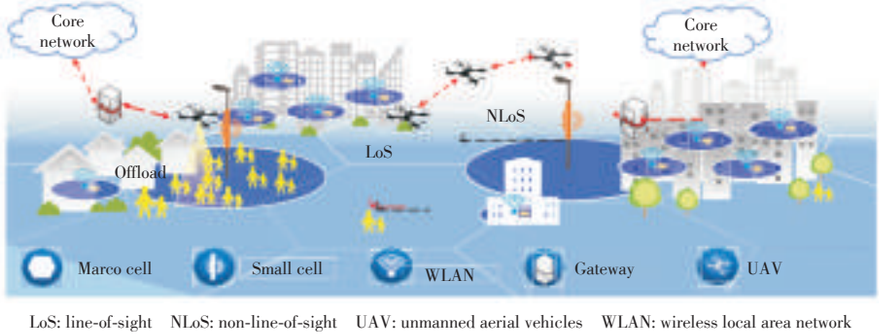
Since low-altitude UAVs could operate over a few hundred of meters, the air-to-ground channel is significantly different from the terrestrial channel. For instance, there are basically less obstructions between a UAV and a ground terminal. In consequence, line-of-sight (LoS) paths are more likely to exist for a UAV-terminal or UAV-BS link. Besides, directional antennas instead of omni - directional antennas are basically equipped on UAVs. In this case, 3D channel modeling would be more suitable for UAV network.

2.2 Potential Benefits of MHetNet

Based on the aforementioned features, we elaborate the potential benefits of MHetNet in detail as follows.

The MHetNet provides more LoS connections. Thanks to the mobility nature of UAVs, LoS paths are more likely to appear in the air-to-ground links, which would lead to smaller pathloss.In consequence, data transmission over the LoS paths could be accomplished with lower transmit power and failure, thereby improving the spectrum and energy efficiency. As well, it is worth noting that the existence of LoS paths is dependent on a number of factors, for instance, the altitude of UAVs.

The MHetNet can offload the traffic of terrestrial access network. The traffic in terrestrial HetNet significantly limits the increase of spectral efficiency especially when the number of users within one small cell is large. In consequence, a proportion of users would be temporarily blocked due to the high traffic load, processing capabilities and backhaul on small cell BSs. Moreover, dense deployment of



LoS: line-of-sight NLoS: non-line-of-sight UAV: unmanned aerial vehicles WLAN: wireless local area network

▲Figure 1. The integration of a multi-layer heterogeneous wireless network.

small cells may not be favorable, since it is indicated that over-densification of small cells would lead to a notable degradation of network capacity [13], [14]. In this case, UAVs can serve as an alternative to assist terrestrial access networks. The blocked users could connect to UAVs, which are capable of providing more LoS paths to ground users and better coverage. As a consequence, the terrestrial traffic could be better balanced and more users could be served.

The CAPEX and OPEX could be significantly reduced. Albeit it is cost-efficient to deploy small cells, the expenditures due to maintenance, operation and backhaul are 10–20 folds greater. Therefore, the tradeoff between the deployment of small cells and the resulting CAPEX and OPEX has been under long consideration in both academia and industry. Especially, considering a flash demand scenario, e.g., concert or open gathering, small cell BSs could only be active for a relative short period, whereas the cost for deployment and cooling would be considerable. Instead, it is more cost-efficient to deploy UAVs in such scenarios, since UAVs could be feasibly deployed to serve as access points only when there is requirement.

The MHetNet has a quick response to rapidly changing user demand. Since user demand may be rapidly changed and greatly differ in different geographic regions, the traffic of terrestrial HetNet is not balanced. Under this circumstance, the UAVs, which could be quickly deployed, can be used as mobile relays in moving overloaded scenarios (e.g., a parade) [15]. With the aid of UAVs, the blocked users could take advantage of unused resources in neighboring cells in priority.

Real-time optimization is feasible for UAVs. Due to fixed deployment, it is difficult for terrestrial wireless networks to deliver reliable service for users in severe shadowing or interference scenarios. However, UAVs are capable of optimizing the topology, altitude and connectivity to the ground users in real time, which has recently received extensive attention from academia [16], [17]. In [16], authors investigate the optimization of UAV altitude so as to maximize the coverage for users. Especially, it is shown that the optimal altitude is critically dependent on the statistical parameters of the underlying environment and pathloss. In addition, the impact of spectrum sharing between UAV and terrestrial network is studied and optimized in [17].

3 Fundamental Challenges of UAV-Assisted MHetNet

While the integration of UAVs into terrestrial wireless network may bring a number of potential benefits, critical issues and challenges remain to be settled before these potential benefits could be readily harvested. As earlier noted, the mobility nature of UAVs would render the topology of UAV network highly time-varying, which results in great difficulty in optimization. In addition, the integration of UAVs may result in additional interference due to the limitation of available spectrum resources. If not properly handled, the incurred interference would degrade the performance of MHetNet as well. In the following, we discuss several key challenges in detail.

3.1 Deployment of UAV Access Points

When UAVs are integrated into terrestrial network, the density of terrestrial BSs plays an important role in influencing the performance of the MHetNet. When the terrestrial BSs are insufficiently deployed, the integration of UAVs could enhance coverage and provide services to more users. Accordingly, the traffic of terrestrial network could be effectively offloaded to the UAV network. In contrast, when the terrestrial network is fully densified, few UAVs should be deployed. The reason is that, for users that are connected to small cell BSs, cross-tier interference from UAVs would become more severe. Hence, the demerits caused by the cross- tier interference overwhelms the benefits of spectrum reuse gain and offloading gain. The detail will be discussed later in Section 4.

3.2 Optimization of UAV Access Points

When UAVs serve as access points, the optimization of UAV access points is dependent on the terrestrial network, such as the topology and transmit power. Furthermore, the factors such as altitude, topology and power optimization. should be considered for the optimization of UAV. For instance, the increase of UAV altitude is likely to lead to an increasing number of LOS paths. Accordingly, the number of users that are served by UAVs would be significantly increased. Nevertheless, the cross-layer interference suffered by terrestrial terminals would be increased as well. Therefore, there exists an optimal altitude for UAVs, under which system performance in MHetNet could be optimized.

The topology of UAV network rapidly changes with the number of UAVs and the relative positions of the UAVs. As a consequence, traditional routing protocols are not always efficient, which may result in user session interruption. A slight move of a single UAV may result in substantive change of the whole network, especially when UAVs are densely deployed. Moreover, when one UAV is disconnected, another UAV should be properly selected as a substitute to minimize the overhead due to topological changes.

Onboard energy optimization is also a critical issue to be settled. For instance, the movement of UAVs should be care- fully controlled by taking into account the energy consumption associated with every maneuver. Moreover, ascending of UAVs is basically energy - intensive. Therefore, excessive frequent changes of UAV altitude should be avoided.

3.3 Spectrum Sharing of Terrestrial and UAV Networks

Spectrum sharing of terrestrial and UAV networks can be generally classified into two categories, namely, reuse mode and dedicated mode. In reuse mode, spectrum resources are

identically allocated to terrestrial terminals and UAVs, thereby improving the reuse of available spectrum resources. However, cross-layer interference would exist between terrestrial terminals and UAVs. If not properly controlled, the induced interference may ruin the potential of the integration of UAVs. Therefore, how to mitigate the cross-layer interference, e.g., via tuning the UAV altitude, topology and transmit power, etc., is challenging in the reuse mode.

To avoid the cross-layer interference, the dedicated mode serves as an alternative especially when considering dense network deployment scenarios. In particular, non-overlapped spectrum resources are allocated to terrestrial terminals and UAVs, respectively. Apparently, the allocation of spectrum resources is a critical issue, which is to be well considered in this mode. Specifically, spectrum allocation is dependent on the parameters such as the densities of terrestrial and UAV networks, the demand of users, which connect to terrestrial BSs and UAVs, the deployment of terrestrial network, and the topology of UAV network. Moreover, it should be noted that reuse and dedicated modes should be dynamically configured to further optimize the performance of MHetNet.

3.4 Backhaul of UAV Network

Backhaul is one of the dominant factors that limit the performance of MHetNet when UAVs are applied as access points. Different from errestrial networks, in which wired backhaul is available, wireless backhaul is the only choice for UAV network. Accordingly, UAVs could connect to ground gateways through multiple hops or the backhaul link could be directly established via the connection to satellites. For either approach, however, delay would be the major concern, which impacts the performance of UAV network, since the delay would be basically increased with the number of relays if connecting ground gateway and real-time service could hardly be provided if connecting to satellites. For the above reasons, the design of backhaul should be fully taken into account when devising the architecture of MHetNet.

4 Simulation Results

In this section, we present simulation results to further investigate the impact of parameters, including densities of small cell BSs, UAVs and users, UAV altitude and LoS transmissions, on the performance of MHetNets. In particular, we adopt spatial throughput (ST) as the performance metric, which is defined by

$$ST=\mu\mathbb{P}(SINR>\tau)\log_2(1+\tau),[bits/(s\cdot Hz\cdot m^2)], \quad (1)$$

where μ denotes the density of active links, τ denotes the signal-to-noise-and-interference ratio (SINR) threshold and $\mathbb{P}(SINR>\tau)$ denotes the success probability of data transmissions. By definition, ST could capture the number of bits that are successfully conveyed over unit time, spectrum and area. Therefore, ST is an important indicator to network capacity.

Simulations are conducted using Matlab and the results are generated over 1 million Monte Carlo trials. In each trial, small cell BSs, UAVs and users are distributed in a 3-dimension space, the 2-dimension coordinates of which follow three independent homogeneous Poisson Point Processes (PPPs) with densities λ_{BS}, λ_{UAV} and λ_U, respectively. Unless otherwise stated, the altitudes of small cell BSs, UAVs, and users are set as $h_{BS}=3$ m, $h_{UAV}=11.5$ m and $h_U=1.5$ m, respectively. In consequence, the antenna height difference (AHD) between BSs and users and the AHD between UAVs and users are $\Delta h_{BS}=1.5$ m and $\Delta h_{UAV}=10$ m, respectively.

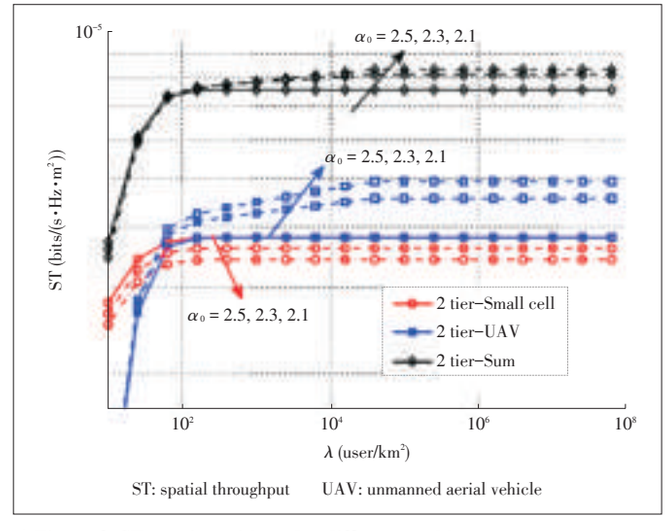
Serving as access points, UAVs reuse the available spectrum resources with terrestrial small cells to serve users with unlimited-capacity backhaul. Full spectrum reuse is considered for small cell BSs and UAVs such that each small cell BS or UAV could serve one user at one time. For user association, each user is first connected to the geometrically closest small cell BS. If the intended BS is connected by more than one user, it would randomly select one user to serve in each time slot and offload the other users to UAVs.

Dual-slope pathloss model (DSPM) is applied to comprehensively characterize the variation of pathloss with transmission distance. In particular, DSPM is defined by

$$l_2\left(\{\alpha_n\}_{n=0}^{1};x\right)=K_n x^{-\alpha_n},R_n\leqslant x<R_{n+1}, \quad (2)$$

where $K_0=1$, $K_1=R_1^{\alpha_1-\alpha_0}$, $R_0=0$ and $R_2=\infty$. In simulations, we set $\alpha_0=2.5, \alpha_1=4$ and $R_1=20$ m for DSPM. In addition, transmit power of each small cell BS is set to be 23 dBm and transmit power of UAV is 30 dBm.

We first investigate the impact of LoS transmission on the performance of two-layer network. In **Fig. 2**, we plot ST as a



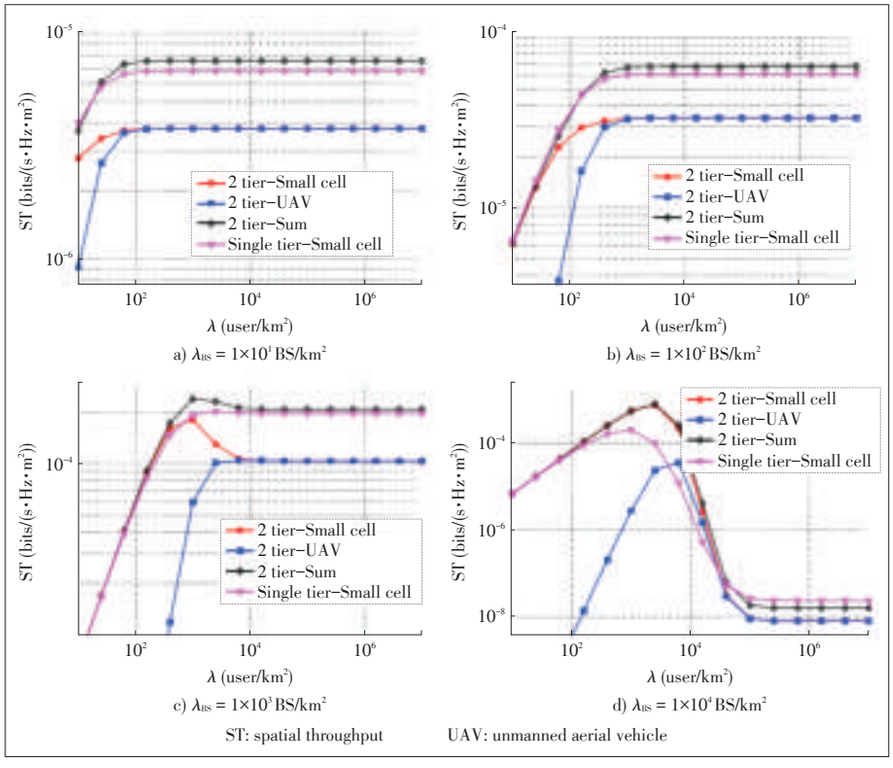
▲**Figure 2. ST varying with under different pathloss exponent α_0. For the system settings, we set $\lambda_{BS}=10$ BS/km^2 and $\lambda_{UAV}=5$ BS/km^2.**

function of user density under different pathloss exponents α_0 in the two-layer network. It is worth noting that a smaller α_0 means that the power loss over LoS paths is smaller and vice versa. It is observed from Fig. 2 that the ST of small cell network monotonously decreases with α_0, while the STs of UAV network and two-layer network would increase with α_0. The reason is that the channel gain over LOS paths inversely increase with α_0 and consequently the desired signal power of UAV-to-user pairs would be enhanced as α_0 grows. Meanwhile, for users that are connected to terrestrial small cell BSs, cross-tier interference from UAVs would become more severe. In consequence, the ST of small cell networks would decrease accordingly.

We then evaluate the performance of multi-layer networks under different BS and UAV densities. In particular, we plot ST as a function of user density under different BS and UAV densities in **Fig. 3**. To validate the benefit of integrating UAVs into terrestrial networks, we evaluate the ST in single-layer small cell networks for comparison as well. It is shown from Figs. 3a and 3b that, although the integration of UAVs would potentially degrade the ST of terrestrial small cell networks due to the generated cross-layer interference, the ST of the two-layer network is greater than that of the single-layer network. The reason is that the traffic of terrestrial network could be effectively offloaded to the UAV network.

However, as the densities of small cell BSs and UAVs grow, the benefits start to diminish. It can be seen from Fig. 3c that the ST of the two-layer network is almost identical to that of single-layer network. Worse still, if the densities of small cell BSs and UAVs further increase, the performance of two-layer network is significantly degraded especially when user density is large (Fig. 3d). The performance degradation is primarily due to the cross-tier interference in the two-layer network. Specifically, when small cell BSs and UAVs are densely deployed, the demerits caused by the cross-tier interference overwhelms the benefits of spectrum reuse gain and offloading gain.

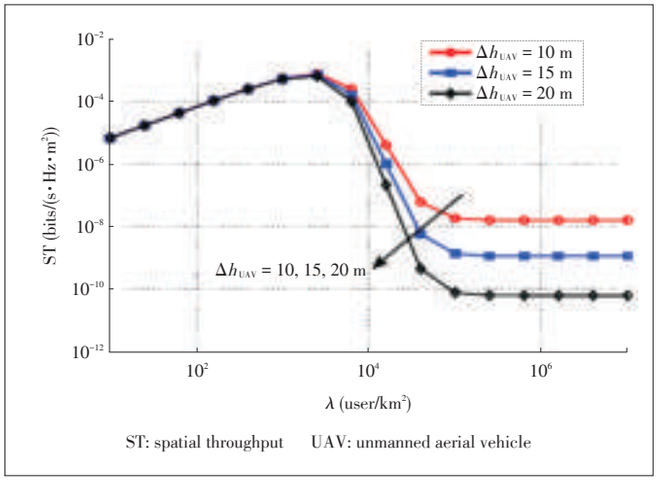
Afterward, we evaluate the impact of UAV altitude on network ST considering dense deployment of network infrastructures, i.e., small cell BSs and UAVs. In particular, **Fig. 4** shows the ST of two-layer network as a function of user density under different Δh_{UAV}. It can be seen that, the network STs under different Δh_{UAV} almost overlap when user density is small. In this case, terrestrial small cell BSs are sufficient to serve all ground users such that a small number of users are offloaded to UAV network. When user density further increases, it is obvious that network ST would decrease with Δh_{UAV}. The reason is that the increase of UAV altitude may result in a higher probability of LoS paths between UAVs and ground users. Although the desired signal power would be accordingly enhanced, the introduced cross-layer interference may be more severe. For this reason, there exists a critical UAV altitude in two-layer network, under which network ST could be maximized. Especially, we obtain the critical altitude of UAVs under different densities of BSs and UAVs in **Table 1**. It can be seen that the critical altitude inversely increases with the BS density. If the altitude of UAVs is greater than the critical altitude, the ST of two-layer network would be degraded. In other words, the critical altitude could serve as a upper bound, under which the integration of UAVs is beneficial to network ST in two-layer network.



▲**Figure 3. ST varying with user density under different BS and UAV densities. In simulations, the UAV density is set as $\lambda_{UAV} = 0.5\lambda_{BS}$.**

5 Conclusion

In this article, we have introduced the architecture of MHetNet, in which UAVs are applied to assist the traditional terrestrial wireless networks to provide better user experience and system performance. After discussing the potential bene- fits of MHetNet, an overview of the key issues and challenges brought by the integration of UAVs has been provided. Aided by simulation results, we have shown that the optimal UAV altitude should be decreased with the density of terrestrial small cell BSs so as to improve the MHet-



▲Figure 4. ST as a function of user density under different Δh_{UAV}. (For system settings, we set $\lambda_{BS} = 1 \times 10^4$ BS/km^2 and $\lambda_{UAV} = 5 \times 10^3$ BS/km^2.)

▼Table 1. Critical altitude with BS density

BS density (/km^2)	Critical altitude (m)
1×10^2	45.2
1×10^3	22.3
1×10^4	10.2
1×10^5	3.9

BS: base station

Net performance. In summary, although UAVs could serve as a promising complementary to existing terrestrial wireless network, it is imperative to investigate and fully exploit the characteristics of UAVs, thereby meeting the ambitious goals of massive connectivity and enormous capacity in the future wireless networks.

References

[1] I.-G. P. Group, "5G vision and requirements," Tech. Rep., Dec. 2015.
[2] Z. Chen and M. Kountouris, "Cache-enabled small cell networks with local user interest correlation," in *IEEE International Workshop on Signal Processing Advances in Wireless Communications*, Stockholm, Sweden, Jul. 2015, pp. 680–684.
[3] H. Zhang, Y. Dong, J. Cheng, M. J. Hossain, and V. C. M. Leung, "Fronthauling for 5G LTE-U ultra dense cloud small cell networks," *IEEE Wireless Communications.*, vol. 23, no. 6, pp. 48–53, Dec. 2016. doi: 10.1109/MWC.2016.1600066WC.
[4] W. Webb, *ArrayComm*. London, U.K.: Ofcom, 2007.
[5] V. Sharma, M. Bennis, and R. Kumar, "UAV-assisted heterogeneous networks for capacity enhancement," *IEEE Communications Letters*, vol. 20, no. 6, pp. 1207–1210, 2016. doi: 10.1109/LCOMM.2016.2553103.
[6] K. P. Valavanis, "Applications of intelligent control to engineering systems," in *Honour of Dr. G. J. Vachtsevanos*. Berlin/Heidelberg, Germany: Springer, 2009.
[7] K. Miranda, A. Molinaro, and T. Razafindralambo, "A survey on rapidly deployable solutions for post-disaster networks," *IEEE Press*, vol. 54, pp. 117–123, Apr. 2016. doi: 10.1109/MCOM.2016.7452275.
[8] M. Ding, P. Wang, D. López-Pérez, G. Mao, and Z. Lin, "Performance impact of LoS and NLoS transmissions in dense cellular networks," *IEEE Transations on Wireless Communications*, vol. 15, no. 3, pp. 2365–2380, Mar. 2016. doi: 10.1109/TWC.2015.2503391.
[9] X. Zhang and J. G. Andrews, "Downlink cellular network analysis with multi-slope path loss models," *IEEE Transactions on Communications*, vol. 63, no. 5, pp. 1881–1894, May 2015. doi: 10.1109/TCOMM.2015.2413412.
[10] J. Liu, M. Sheng, L. Liu, and J. Li, "Interference management in ultra-dense networks: Challenges and approaches," *IEEE Network*, vol. 31, no. 6, pp. 70–77, Nov. 2017. doi: 10.1109/MNET.2017.1700052.
[11] J. Liu, M. Sheng, L. Liu, and J. Li, "Effect of densification on cellular network performance with bounded pathloss model," *IEEE Communications Letter*, vol. 21, no. 2, pp. 346–349, Feb. 2017. doi: 10.1109/LCOMM.2016.2615298.
[12] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Communications Surveys Tutorials*, vol. 18, no. 2, pp. 1123–1152, Second quarter 2016. doi: 10.1109/COMST.2015.2495297.
[13] D. López-Pérez, M. Ding, H. Claussen, and A. H. Jafari, "Towards 1 Gbps/UE in cellular systems: understanding ultra-dense small cell deployments," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2078–2101, Fourth quarter 2015. doi: 10.1109/COMST.2015.2439636.
[14] J. Liu, M. Sheng, L. Liu, and J. Li, "Network densification in 5G: From the short-range communications perspective," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 96–102, Dec. 2017. doi: 10.1109/MCOM.2017.1700487.
[15] S. Rohde and C. Wietfeld, "Interference aware positioning of aerial relays for cell overload and outage compensation," in *IEEE VTC-Fall*, Quebec City, Canada, Sept. 2012, pp. 1–5. doi: 10.1109/VTCFall.2012.6399121.
[16] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Communications Letter*, vol. 3, no. 6, pp. 569–572, Dec. 2014. doi: 10.1109/LWC.2014.2342736.
[17] C. Zhang and W. Zhang, "Spectrum sharing for drone networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 1, pp. 136–144, 2017. doi: 10.1109/JSAC.2016.2633040.

Manuscript received: 2018-01-14

Biographies

LI Tongxin (txli@stu.xidian.edu.cn) received the B.Eng. degree in communication and information systems from Xidian University, China in 2016. She is currently pursuing the M.A.Sc. degree with the State Key Laboratory of Integrated Service Networks, Institute of Information and Science, Xidian University. Her research interests include performance evaluation for UAV assisted cellular network and small cell caching in ultra-dense wireless networks.

SHENG Min (msheng@mail.xidian.edu.cn) received the M.S. and Ph.D. degrees in communication and information systems from Xidian University, China in 1997 and 2000, respectively. She is currently a full professor with the State Key Laboratory of Integrated Service Networks, Institute of Information Science, Xidian University. Her general research interests include mobile ad hoc networks, 5G mobile communication systems, and satellite communications networks. She was awarded as Changjiang Scholar by Ministry of Education, China.

LYU Ruiling (rllv@stu.xidian.edu.cn) received the B.Eng. degree from Inner Mongolia University, China in 2017. She is currently pursuing the M.A.Sc. degree with the State Key Laboratory of Integrated Service Networks, Institute of Information and Science, Xidian University. Her research interests include interference management and performance evaluation for UAV assisted cellular network, as well as ultra-dense wireless networks.

LIU Junyu (junyuliu@xidian.edu.cn) received the B.Eng. and Ph.D. degrees in communication and information systems from Xidian University, China in 2007 and 2016, respectively. He is currently a lecturer/postdoc with the State Key Laboratory of Integrated Service Networks, Institute of Information and Science, Xidian University. His research interests include interference management and performance evaluation of wireless heterogeneous networks and ultra-dense wireless networks.

LI Jiandong (jdli@mail.xidian.edu.cn) received the B.E., M.S. and Ph.D. degrees in communications engineering from Xidian University, China in 1982, 1985 and 1991, respectively. He has been a faculty member of the School of Telecommunications Engineering, Xidian University since 1985, and now serves as the Vice President of Xidian University. He was awarded as Distinguished Young Researcher from NSFC and Changjiang Scholar by Ministry of Education, China. His major research interests include wireless communication theory, cognitive radio and signal processing.

DOI: 10.3969/j.issn.1673-5188.2018.02.003
http://kns.cnki.net/kcms/detail/34.1294.TN.20180518.1657.002.html, published online May 18, 2018

Multi-QoS Guaranteed Resource Allocation for Multi-Services Based on Opportunity Costs

JIN Yaqi, XU Xiaodong, and TAO Xiaofeng
(Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract

To meet the booming development of diversified services and new applications in the future, the fifth-generation mobile communication system (5G) has arisen. Resources are increasingly scarce in the dynamic time-varying of 5G networks. Allocating resources effectively and ensuring quality of service (QoS) requirements of multi-services come to be a research focus. In this paper, we utilize effective capacity to build a utility function with multi-QoS metrics, including rate, delay bound and packet loss ratio. Taking advantage of opportunity cost (OC), we also propose a multi-QoS guaranteed resource allocation algorithm for multi-services to consider the future condition of system. In the algorithm, according to different business characteristics and the theory of OC, we propose different selection conditions for QoS users and best effort (BE) users to choose more reasonable resources. Finally, simulation results show that our proposed algorithm achieves superior system utility and relatively better fairness in multi-service scenarios.

Keywords

utility function; effective capacity; opportunity cost; QoS guaranteed; resource allocation

1 Introduction

The rapid development of the mobile Internet has driven the demand of users for higher speed, larger data traffic and more intensive network coverage. Therefore, the fifth-generation of mobile communication system (5G) has emerged. With the explosive growth of business demands in 5G, more services which have diverse quality of service (QoS) requirements appear. Due to the scarcity of resources in mobile communication systems, the importance of resource allocation in system performance is decisive. The conflict between limited wireless resources and the ever-increasing QoS needs has become increasingly acute. Therefore, we call for a resource allocation strategy that can improve overall network performance as well as support high quality multi-services.

Researches on resource allocation for QoS guaranteed under multi-service hybrid scenarios have increasingly become a research hotspot. The authors of [1] solved the distributed-resource allocation problems in 5G cellular systems. They utilized the concepts of stable matching, factor-graph-based message passing, and distributed auctions. The authors of [2] investigated the problem of power allocation and sub-channel assignment in heterogeneous small cell network. They considered cross-tier interference mitigation, energy harvesting and incomplete channel state information. In [3], the optimal objective function proposed in the wireless resource allocation algorithm is introduced. The time delay is taken into as a constraint to guarantee QoS. The size of the delay constraint value can be set according to the priority and quality of service requirements. In [4], the researchers proposed a heterogeneous QoS-driven resource allocation scheme by the multiple input multiple output-orthogonal frequency-division multiple access (MIMO-OFDMA) based relaying scheme. Given the heterogeneous statistical QoS constraints, the authors derived the effective capacity under developed optimal power-allocation policies. The authors of [5] proposed a QoS scheduling strategy for multi-users and multi-services. They considered the service type, channel quality, buffer size and fairness based on carrier aggregation.

In this paper, we consider multi-services whose QoS metrics include bandwidth requirements, as well as delay, packet loss ratio, etc. [6]. Therefore, we firstly analyze different typical services' business characteristics. To consider the variety of QoS metrics, we combine the theory of effective capacity (EC) with unified utility function to well characterize multiple QoS constraints [7]–[8]. EC is the maximum constant data rate that a given service rate can support subject to a QoS exponent [9].

This paper is supported by the National Science and Technology Major Project under Grant No. 2016ZX03001009-003 and the Nature and Science Foundation of China under Grants Nos. 61471068 and 111 Project of China B16006.

Besides, in order to consider the dynamic time-varying conditions of 5G ultra-dense networks, the paper introduces the opportunity cost (OC) model. In economics, the original concept of OC is the maximum benefit that can be gained from other uses after the resource is put into a particular use [10]–[11]. As the theory of OC evolves and expands, the opportunity cost is used in not only economics, but also multi-service resources allocation scenarios. According to OC, we propose a multi-service QoS guaranteed resource allocation algorithm, which can improve the system utility while guaranteeing the QoS requirement of multi-users within a long-time frame.

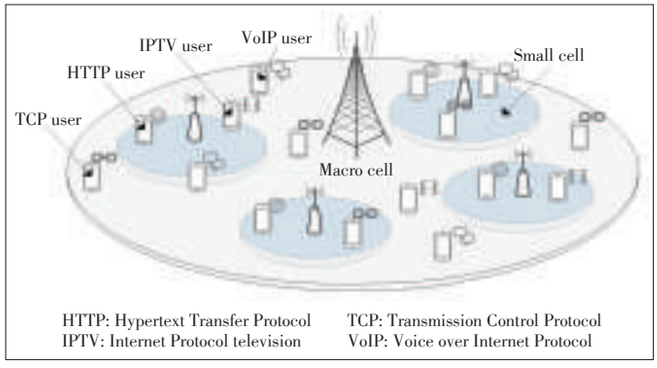
The contributions of this paper can be summarized as follows:

1) Our scheme invites effective capacity based utility function to establish a multi-QoS optimization strategy that put the system utility as the objective function.
2) We introduce opportunity cost to take future network condition into account, which can lead multi-services to make rational choices to conduct resource allocation within the tolerance of delay requirement.
3) The schemes and simulation results show that our algorithm can achieve superior overall user satisfaction and fairness. The correctness of the proposed utility is also evaluated, which could be used as references for related studies.

The rest of this paper is organized as follows. Section 2 provides the system model, EC-based utility function and opportunity cost for multi-service scenarios. Section 3 formulates the multi-service QoS guaranteed resource allocation problem. By comparing opportunity cost and current data rate of QoS users and BE users respectively, we come to the conditions users choose more suitable resources to get better system performance. Simulation results are given and discussed in section 4. Finally, section 5 draws conclusions.

2 System Model

We consider the downlink scenario of multi-service cellular networks as shown in **Fig. 1**. The scenario depicts N base stations (BSs), the power of which is limited by P_n respectively.



▲**Figure 1. The system model of wireless network for multi-QoS mobile services.**

There are multiple users, which are categorized into four classes: conversational class (VoIP), streaming class (IPTV), interactive class (HTTP) and background class (TCP). The number of VoIP, IPTV, HTTP and TCP users are K_1, K_2, K_3, and K_4 respectively, and the sum of the four kind of users is equal to K. The resource pool has M resource blocks (RBs), which we allow each BS to use. We assume that in each scheduling cycle, a RB m can only be assigned to one user k at most. $\beta \in \{0,1\}^{K\times M\times N}$ is the RB assignment matrix, where $\beta_{k,m,n}=1$ indicates the assignment RB and $\beta_{k,m,n}=0$, otherwise.

The Signal to Noise Ratio (SNR) between k-th user and n-th BS on RB m is formulated as follows:

$$SNR_{k,m,n}=P_{k,m,n}\cdot\left|H_{k,m,n}\right|^2/(N_0B), \tag{1}$$

where $P_{k,m,n}$ denotes the transmit power. $\left|H_{k,m,n}\right|^2$ denotes the channel gain. N_0 is the power spectral density of noise and B is the bandwidth of each RB.

Based on Shannon's capacity formula, the rate between the k-th user and the n-th BS on RB m is given by (2):

$$R_{k,m,n}=B\log(1+P_{k,m,n}\cdot C_{k,m,n}), \tag{2}$$

where $C_{k,m,n}=\left|H_{k,m,n}\right|^2/(N_0B)$. The total data rate that user k gets from BS n is expressed as:

$$R_{k,n}=\sum_{m=1}^{M}\beta_{k,m,n}\cdot R_{k,m,n}. \tag{3}$$

2.1 Effective Capacity Based Utility Function

To measure user satisfaction, utility function is proposed. The higher the value of $U(r)$, the higher the user satisfaction. To analyze different service characteristics, the utility curves of four typical classes are introduced in **Fig. 2** [12].

From the research of [13], we use the effective capacity based utility function to consider multi-QoS requirements. The function combines the effective capacity with a uniform utility function, which is expresses as

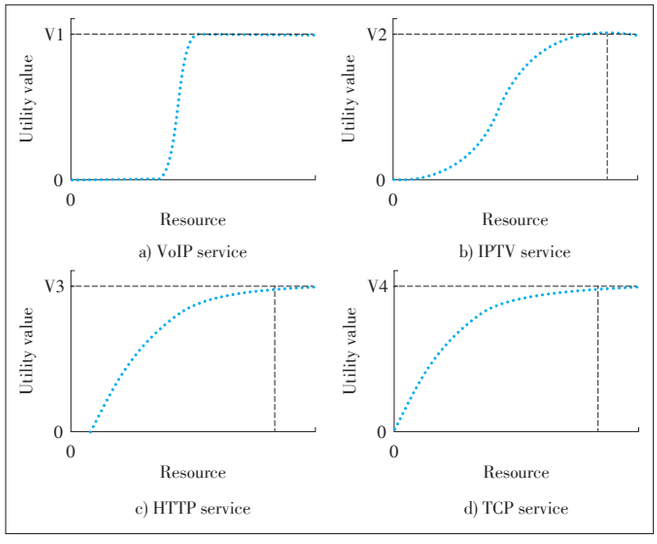
$$U_k(E_{C_k}(\theta_{k,n}))=U_k(P_{k,m,n},\beta_{k,m,n},\theta_{k,n}), \tag{4}$$

where $E_{C_k}(\theta_{k,n})$ is the effective capacity with the parameter of QoS exponent θ.

The utility function $U(r)$ can be set by different parameters to form different curves adapted to different typical classes [8].

$$U(r)=\frac{1}{A+Be^{-C(r-d)}}+D, \tag{5}$$

where r is the resource parameter. Parameters A, B and D mainly affect the range of the utility function. C affects the



▲Figure 2. The utility function trend of four classes.

slope of the curve. And d is the inflexion of the function.

The effective capacity is expressed as following:

$$E_C(\theta) = -\frac{1}{\theta T}\log(E[e^{-\theta T R[i]}]), \tag{6}$$

where $R[i]$ represents the instantaneous channel capacity during i-th frame. T represents the frame duration.

Moreover, the probability that the delay bound $D_{\max}$ must be below a certain packet loss ratio ε is expressed as (7):

$$\Pr\{D(\infty) > D_{\max}\} \approx e^{-\theta(\lambda)\cdot D_{\max}} \leqslant \varepsilon, \tag{7}$$

where λ represents the service arriving rate.

2.2 Opportunity Cost

Multi-service resource allocation provides corresponding service quality assurance for different user terminals under the condition of limited resources. By allocating available resources such as subcarriers and power properly, we use resource allocation to maximize user satisfaction, system throughput and the efficiency of resource utilization. If a user selects a base station's resources, it will lose the opportunity to access other base stations. If a base station chooses a user to provide service, it will lose opportunity to allocate resources to other users. In summary, the scenario of multi-service resource allocation meets the three preconditions of opportunity cost: scarcity, diversity and decision rationality of resources.

In this paper, we divide users into QoS users and BE users to research multi-service resource allocation scheme. We assume that choices are made rationally by users aiming to maximize system utility.

For QoS users, when the EC of a QoS user is greater than the expected effective bandwidth, increasing the rate will not greatly improves the utility. However, for BE users, as user data rate continues to increase, their utility will continue to improve. Therefore, if QoS users select the current resource, they may lose the better suited resource when the delay is tolerable. It is also possible that the waiting time exceeds the delay limit so that a serious decline in channel quality causes calls drop.

For BE users, if they choose the current resource, they may lose the larger expected transmission rate that can be obtained when its latency is tolerable. If BE users drop the connection, it is also possible to drop the call. Therefore, from the above resource allocation, the user needs to make a choice to select or abandon a certain resource.

We come to the definition of opportunity cost for resource allocation as follows: the opportunity cost of a user is defined as the expected value of the service transmission rate that the user chooses to wait within the delay allowed by the service.

By introducing the concept of opportunity cost, we can not only allocate according to the current resource situation, but also consider the future changes of the system and maximize the utility value of the system through the rational choice of users.

3 Multi-Service QoS Guaranteed Resource Allocation Based on Opportunity Cost

In this section, we study the multi-service QoS guaranteed resource allocation algorithm based on opportunity cost aiming to improve system utility.

Taking advantage of opportunity cost, taking the opportunity cost into consideration can consider the advantages of the future situation of the system and put forward a more reasonable resource allocation algorithm.

3.1 Problem Formulation

To optimize multi-service resource allocation algorithm, we aim to maximize system utility, that is, the sum utility function of each user. The objective function is shown in the following formula:

$$\max \sum_k U_k(P_{k.m.n}, \beta_{k.m.n}, \theta_{k,n}). \tag{8}$$

In this paper, we suppose users are sorted by the delay bound $D_{\max,k}$ from the smallest to the largest. It is easy to find that user sequence meets $\{k \in K_1, k \in K_2, k \in K_3, k \in K_4\}$. Meanwhile, the channel transmission capabilities of BSs are sorted from the largest to the smallest. And the corresponding channel transmission capabilities of the base stations are $\{E_{C_n} | n = 1,2,...,N\}$. Without loss of generality, we suppose $E_{C_1} \geqslant E_{C_2} \geqslant ... \geqslant E_{C_N}$.

Assume the user's service duration has an exponential distribution with parameter τ [14]. The distribution function of the service duration T of each user in the base station is as following,

$$F(t) = P(T \leqslant t) = \begin{cases} 1 - e^{-\frac{t}{\tau}}, t > 0. \\ 0, \quad \text{other} \end{cases} \tag{9}$$

For a full-loaded BS, the probability that the BS will have free resources, that is, at least one user ends the service leaving within the duration t, can be expressed as

$$P(t)=1-P(T_1>t,T_2>t,...,T_K>t). \tag{10}$$

Supposing the service duration of user is mutual independent of each other, the probability is written as

$$P(t)=1-(e^{-t/\tau})^K=1-e^{-Kt/\tau}. \tag{11}$$

From the above analysis, the probability that the $BS-i=1,2,...,N$ will have free resources is $1-e^{-K_i t/\tau}$ respectively, where K_i represents the number of users served by the i-th BS and $K_0=0$.

Assuming that the resource is pre-assigned without considering the future condition of the system, the power pre-allocated to the user is $P^{'}_{k,m,n}$. And if the user occupies the RB $m^{'}$ on the BS $n^{'}$, it is denoted as $\beta^{'}_{k,m,n}=1$. For QoS users, we suppose the allocated data rates all satisfy $E_c(P^{'}_{k,m,n},\beta^{'}_{k,m,n},\theta_{k,n})\geqslant E_b^{\,k},k\in K_1,K_2$. Effective bandwidth E_b is defined as the minimum service rate to meet the QoS requirements [15] According to the utility characteristic of QoS users from Fig. 2, when the effective capacity of QoS users is larger than their effective bandwidth, increasing the rate does not greatly improve their utility. Therefore, it is better for QoS user k to obtain desired data rates smaller but meets its QoS requirement during its waiting time t which is less than the delay tolerance range $D_{\max,k}$. Therefore, the resources can be assigned to other users, so as to enhance the overall effectiveness of the system and the system fairness. The condition QoS users choose to wait as shown in (12):

$$E_b^{\,k}\leqslant E_{QoS}[E_C(\theta)]\leqslant E_C(P^{'}_{k,m,n},\beta^{'}_{k,m,n},\theta_{k,n}). \tag{12}$$

Therefore, QoS users only consider the base station whose transmission capacity is lower than $n^{'}$ to save more resources. We can conclude the expected transmission rate that can be obtained in the waiting time t for QoS users in (13):

$$E_{QoS}[E_C(\theta)]=\sum_{i=n^{'}+1}^{N}(1-e^{-\frac{K_i t}{\tau}})e^{-\sum_{s=i+1}^{N}\frac{K_s t}{\tau}}\cdot E_c(P_{k,m,i},\beta_{k,m,i},\theta_{k,i}). \tag{13}$$

What is different from QoS users, when BE user is provided more resources, is that BE user satisfaction can still be improved. Therefore, if the opportunity cost of BE user k after the waiting time t is greater than the pre-allocated resources, BE users may choose to wait to access a higher transfer rate, as shown in the following equation:

$$E_{BE}[E_C(\theta)]\geqslant E_C(P^{'}_{k,m,n},\beta^{'}_{k,m,n},\theta_{k,n}). \tag{14}$$

Unlike QoS users, BE users only consider base stations with transmission capability higher than $n^{'}$ in order to obtain more data rates to improve their utility. The expected transmission rate that can be obtained in the waiting time t is as follows:

$$E_{BE}[E_C(\theta)]=\sum_{i=1}^{n^{'}-1}(1-e^{-\frac{K_i t}{\tau}})e^{-\sum_{s=1}^{i}\frac{K_{s-1} t}{\tau}}\cdot E_c(P_{k,m,i},\beta_{k,m,i},\theta_{k,i}). \tag{15}$$

If there are multiple users waiting in line for a certain BS service, the waiting queue length is set as j_i (the user is ranked j_i) and $j_0=0$. The probability that the user can get service within the tolerable waiting time range is equal to the probability that at least j_i users leaving within the time frame. It can be seen that the time interval for the user to leave the system obeys the negative exponential distribution of parameter $\lambda_i=K_i/\tau$.

Therefore, the number of users leaving the system in time obeys Poisson distribution. The probability of the user getting the service is converted to (16):

$$P(t_1+t_2+...+t_j<t)=P(k\geqslant j)=1-\sum_{k=0}^{j}\frac{\lambda^k}{k!}e^{-\lambda}. \tag{16}$$

Thus, the expected transmission rate that QoS users can wait for access is:

$$E_{QoS}[E_C(\theta)]=\sum_{i=n^{'}+1}^{N}(1-\sum_{k=0}^{j_i}\frac{\lambda_i^k}{k!}e^{-\lambda_i})\{\prod_{s=i+1}^{N}(\sum_{k=0}^{j_s}\frac{\lambda_s^k}{k!}e^{-\lambda_s})\}\cdot E_c(P_{k,m,i},\beta_{k,m,i},\theta_{k,i}). \tag{17}$$

The condition QoS users choose to wait is shown as following:

$$E_b^{\,k}\leqslant\sum_{i=n^{'}+1}^{N}(1-\sum_{k=0}^{j_i}\frac{\lambda_i^k}{k!}e^{-\lambda_i})\{\prod_{s=i+1}^{N}(\sum_{k=0}^{j_s}\frac{\lambda_s^k}{k!}e^{-\lambda_s})\}\cdot E_c(P_{k,m,i},\beta_{k,m,i},\theta_{k,i})\leqslant E_c(P^{'}_{k,m,n},\beta^{'}_{k,m,n},\theta_{k,n}). \tag{18}$$

The expected transmission rate that a BE user can wait for access is:

$$E_{BE}[E_C(\theta)]=\sum_{i=1}^{n^{'}-1}(1-\sum_{k=0}^{j_i}\frac{\lambda_i^k}{k!}e^{-\lambda_i})\{\prod_{s=1}^{i}(\sum_{k=0}^{j_{s-1}}\frac{\lambda_{s-1}^k}{k!}e^{-\lambda_{s-1}})\}\cdot E_c(P_{k,m,i},\beta_{k,m,i},\theta_{k,i}). \tag{19}$$

The condition BE users choose to wait is shown as following:

$$\sum_{i=1}^{n^{'}-1}(1-\sum_{k=0}^{j_i}\frac{\lambda_i^k}{k!}e^{-\lambda_i})\{\prod_{s=1}^{i}(\sum_{k=0}^{j_{s-1}}\frac{\lambda_{s-1}^k}{k!}e^{-\lambda_{s-1}})\}\cdot E_c(P_{k,m,i},\beta_{k,m,i},\theta_{k,i})\geqslant E_C(P^{'}_{k,m,n},\beta^{'}_{k,m,n},\theta_{k,n}). \tag{20}$$

In addition, users need to tolerate delay during the waiting of the queuing. And each user has the maximum delay limit $D_{\max,k}$, beyond which the user's service quality will be affected. Therefore, the duration a user chooses to wait should be less than its maximum delay limit. When user k queues for the service of a selected base station n, we assume the length of waiting queue is j_n. The probability distribution of the user's

service duration obeys the exponential distribution with parameter τ. According to the characteristic of exponential distribution, the expected duration is τ. So the user's queue waiting delay is $j_n\tau$. Therefore, j_n needs to satisfy the following formula:

$$j_n \leqslant [\frac{D_{\max,k}}{\tau}], \tag{21}$$

where $[x]$ represents an integer not greater than x.

3.2 Problem Solution

Based on the above analysis, a multi-service QoS guaranteed resource allocation optimization model based on opportunity cost is formed as follows:

$$\begin{aligned}
&\max \sum_k U_k(P_{k,m,n},\beta_{k,m,n},\theta_{k,n}),\\
&\text{s.t. 1)}\ E_b^k \leqslant \sum_{i=n+1}^{N}(1-\sum_{k=0}^{j_i}\frac{\lambda_i^k}{k!}e^{-\lambda_i})\{\prod_{s=i+1}^{N}(\sum_{k=0}^{j_s}\frac{\lambda_s^k}{k!}e^{-\lambda_s})\}\cdot E_c(P_{k,m,i},\beta_{k,m,i},\theta_{k,i}) \leqslant\\
&\quad E_c(P'_{k,m,n},\beta'_{k,m,n},\theta_{k,n}), k\in K_1,K_2,\\
&2)\ \sum_{i=1}^{n'-1}(1-\sum_{k=0}^{j_i}\frac{\lambda_i^k}{k!}e^{-\lambda_i})\{\prod_{s=1}^{i}(\sum_{k=0}^{j_{s-1}}\frac{\lambda_{s-1}^k}{k!}e^{-\lambda_{s-1}})\}\cdot E_c(P_{k,m,i},\beta_{k,m,i},\theta_{k,i}) \geqslant\\
&\quad E_C(P'_{k,m,n},\beta'_{k,m,n},\theta_{k,n}), k\in K_3,K_4,\\
&3)\ j_n \leqslant [\frac{D_{\max,k}}{\tau}],\ n=1,...,N,\\
&4)\ \sum_k\sum_n P_{k,m,n} \leqslant P_n,\\
&5)\ \sum_k\sum_n \beta_{k,m,n} \leqslant 1, \beta_{k,m,n}\in\{0,1\}.
\end{aligned} \tag{22}$$

The algorithm flow is designed as follows:

1) First, we calculate the channel transmission capacity of each base station and sort it from high to low, numbered 1, ... ,n, ... N. The users are sorted according to their maximum delay limit from low to high, numbered as 1, ... ,k, ... K. Then when $k=1,...,K_1+K_2$, step 2 is performed for a QoS user; when $k=K_1+K_2+1,...,K$, step 3 is performed for a BE user.

2) When $k=1,...,K_1+K_2$, we calculate the opportunity cost of QoS users. The algorithm finds the value of n that satisfies the limit conditions 1) and 3) in (22). To make the system utility higher, set $n^*=\arg\max\{n\}$, $\beta^*_{k,m,n}=1$, $j_{n^*}+1$. If not, then make $k+1$.

3) When $k=K_1+K_2+1,...,K$, we calculate the opportunity cost of BE users. And then the algorithm finds the value of n that satisfies the limit conditions 2) and 3) in (22) To make the system utility higher, set $n^*=\arg\max\{n\}$, $\beta^*_{k,m,n}=1$, $j_{n^*}+1$. If not, then make $k+1$.

4) After traversing all the users, the resource allocation is completed. And the power matrix and the allocation matrix are updated when each n' is taken.

4 Simulation Results and Analysis

4.1 Simulation Parameters

In this section, simulation results about the utility-based resource allocation for multi-QoS services with EC are given. There are multiple BSs deployed in a coverage area of 2 km×3 km. All BSs connect with the controller. There are total 50 files and the size of each content is a normal random variable with the mean of 30 Mbits. The requests of users follow a Zipf distribution with Zipf parameter g=0.5. Generally, g indicates the degree of skewness of popularity distribution, a larger g means the content requests are more centralized into popular files. Besides, we let the users of the four classes have the same proportion. More details of simulation environment settings and QoS settings are listed in **Tables 1** and **2** respectively. The QoS parameters are set according to 3GPP TS 23.203 [6]. The four classes have different QoS parameters, such as delay bound and packet loss ratio. Besides, VoIP users and IPTV users have effective bandwidth bound, because they are QoS users who have minimum bandwidth requirement.

4.2 Results and Analysis

According to the proposed resource allocation algorithm, QoS users are expected to seek more suitable resources instead of blindly seeking for better resources by comparing opportunity cost with current resources favorably. In this paper, we compare sum utility, system throughput and fairness between the proposed OC algorithm, the max utility algorithm without OC

▼**Table 1. Simulation setting: system parameters**

System parameters	
Number of BSs	7
Number of Subchannels	50
Maximum power of BSs	46 dBm
Carrier Frequency	2 GHz
Bandwidth	10 MHz
Cell average radius	500 m
Pathloss model	$PL=128.1+37.6\log_{10}d, d(km)$
Shadowing standard deviation	8 dB
Shadowing correlation distance	50 m
Fast fading	Rayleigh fading
Noise density	-174 dBm/Hz
Average arriving rate	150 kbit/s

BS: base station

▼**Table 2. Simulation setting: QoS parameters**

Traffic Type	Effective bandwidth (kbit/s)	Delay bound (ms)	Packet loss ratio
VoIP	150	[20, 50]	10^{-2}
IPTV	200	[50, 100]	10^{-3}
HTTP	--	[100, 200]	10^{-6}
TCP	--	[300, 500]	10^{-6}

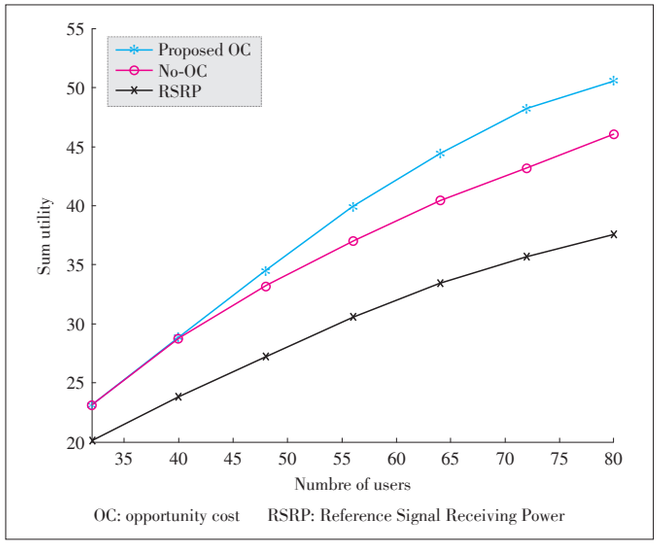
HTTP: Hypertext Transfer Protocol
IPTV: Internet Protocol television
QoS: quality of service
TCP: Transmission Control Protocol
VoIP: Voice over Internet Protocol

(No-OC) and the Reference Signal Receiving Power (RSRP) algorithm.

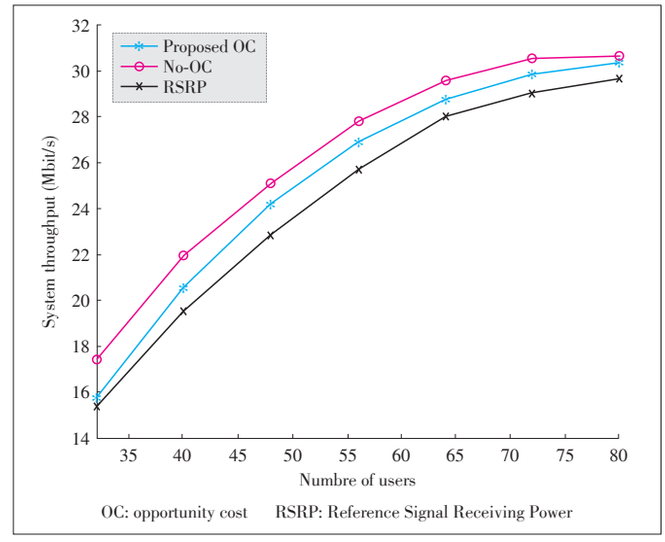
Fig. 3 depicts the sum utility versus number of users. Since the sum utility is the cumulative result of users´ utility, all three algorithms are monotonically increasing. The OC algorithm and the No-OC algorithm always give higher utility values than the RSRP algorithm, because they guarantee the QoS requirements of the more efficient QoS users. When the number of users is small, the OC algorithm has almost the same utility value as the No-OC algorithm. As the number of users increase, the utility value of the OC algorithm is significantly higher than that of the No-OC algorithm. This is because, when the number of users is small, the resources are sufficient, and both strategies can meet the needs of almost all users. However, as the number of users increases, BE users in the No-OC algorithm cannot be met, while the QoS users in the OC algorithm can obtain more reasonable resources based on the opportunity cost to save more resources for the BE users to implement higher system utility.

Fig. 4 shows the system throughput versus the number of users. The throughput of the three algorithms depicts an increasing trend with the increase of the number of users. The proposed OC algorithm is between No-OC and RSRP algorithm. This is due to the OC algorithm sacrificing some of the QoS user data rates. When the rate of QoS users meets their requirements, the utility value will not increase any more. Therefore, the resources allocated by QoS are allocated to BE users by OC algorithm to obtain higher system utility. Therefore, the total system throughput is somewhat lower than the No-OC algorithm.

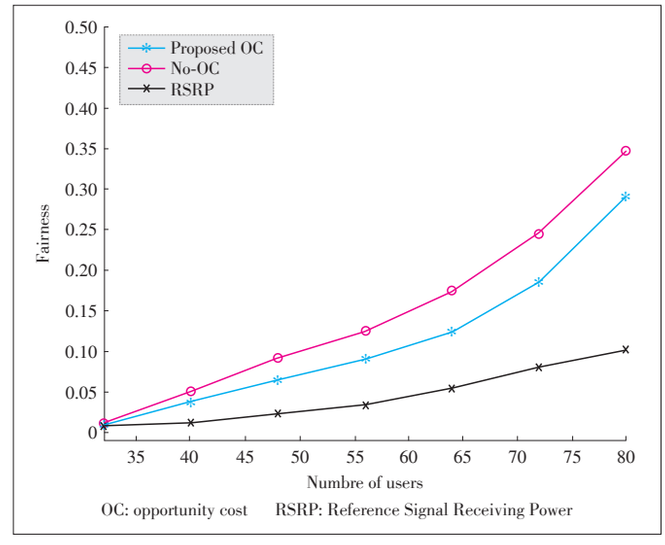
The fairness versus the number of users is shown in **Fig. 5**. The fairness factor is defined as the user-derived data rate normalized variance. The lower the fairness factor indicates, the greater the variance and the more unfair the algorithm are. We



▲Figure 3. Sum utility versus the number of users.



▲Figure 4. System throughput versus the number of users.



▲Figure 5. Fairness versus number of users.

observe that the fairness of OC algorithm is better No-OC algorithm. Because the No-OC algorithm always guarantees the QoS requirements of the QoS users and does not consider the BE users well. The OC algorithm takes advantage of the opportunity cost so that the BE users and the QoS users can make constant calculations and thus have better fairness. For RSRP algorithm, all users are equal without distinguishing different users with different QoS requirements. Therefore, its fairness is better than OC algorithm and No-OC algorithm.

5 Conclusions

In conclusion, the proposed multi-QoS guaranteed resource allocation algorithm for multi-services based on opportunity cost can achieve a well-done balance between user satisfaction

and system fairness. We first formulate a unified utility function with effective capacity, which is able to represent the QoS requirements of delay and packet loss rate, to describe the multi-QoS metrics of different services. Then we invite the theory of opportunity cost in economy to form the concept of opportunity cost applied into the multi-service resource allocation scenario by analyzing the utility characteristics of QoS users and BE users respectively. In the multi-services resource allocation scheme, QoS users and BE users make different preferences rationally to maximize system utility. From business characteristic of different users, if boosting the service rate of QoS users when their lowest data rate is met, system utility will not be effectively increased. Meanwhile, BE users always look for a higher data rate to enhance system utility. Therefore, by calculating opportunity cost, QoS users tend to choose resources that have smaller data rates but meet their QoS requirements within delay limits. And BE users prefer resources that provide higher transmission capacity within delay limits. Finally, the simulation results show that our algorithm can achieve superior overall user satisfaction and the algorithm also balances fairness and throughput in a good way.

References

[1] R. Vannithamby and S. Talwar, "Distributed resource allocation in 5G cellular networks," in *Towards 5G:Applications, Requirements and Candidate Technologies*. New York, USA: Wiley, 2017. doi: 10.1002/9781118979846.ch8.
[2] H. Zhang, J. Du, J. Cheng, et al., "Incomplete CSI based resource optimization in SWIPT enabled heterogeneous networks: a non-cooperative game theoretic approach," *IEEE Transactions on Wireless Communications*, pp. 1882–1892, Mar. 2018. doi: 10.1109/TWC.2017.2786255.
[3] H. Y. Liu, S. Y. Xu, K. S. Kwak, et al., "Geometric programming based distributed resource allocation in ultra dense hetnets," in *IEEE 83rd Vehicular Technology Conference*, Nanjing, China, 2016, pp. 1–5. doi: 10.1109/VTCSpring.2016.7504261.
[4] X. Zhang and J. Q. Wang, "Heterogeneous QoS-driven resource allocation over MIMO-OFDMA based 5G cognitive radio networks," in *IEEE Wireless Communications and Networking Conference*, San Francisco, USA, 2017, pp. 1–6. doi: 10.1109/WCNC.2017.7925876.
[5] Q. L. Wang, Q. X. Zhang, Y. H. Sun, et al., "A QoS-guaranteed radio resource scheduling in multi-user multi-service LTE-A systems with carrier aggregation," in *IEEE 2nd International Conference on Computer and Communications*, Chengdu, China, 2016. pp. 2927–2932. doi: 10.1109/CompComm.2016.7925233.
[6] *Technical Specification Group Services and System Aspects, QoS Concept and Architecture (Release 1999)*, 3GPP TS 23.107 v3.7.0, 2002.
[7] D. P. Wu and R. Negi, "Effective capacity: a wireless channel model for support of quality of service," *IEEE Transactions on Wireless Communications*, vol. 2, no. 4, pp. 630–643, Jul. 2003. doi: 10.1109/TWC.2003.814353.
[8] L. Chen, B. Wang, X. H. Chen, X. Zhang, and D. C. Yang, "Utility-based resource allocation for mixed traffic in wireless networks," in *IEEE Conference on Computer Communications Workshops*, Shanghai, China, 2011, pp. 91–96. doi: 10.1109/INFCOMW.2011.5928944.
[9] S. Ahn, H. Wang, S. Han, and D. Hong, "The effect of multiplexing users in QoS provisioning scheduling," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2575–2581, 2010.
[10] F. M. Xue and W. F. Zhang, "Analysis of opportunity costs," *Market Modernization*, vol. 566, no. 2, pp. 73–73, 2009.
[11] F. Chi, "Research on the meaning, expression and use of opportunity cost," *Journal of Changchun University of Science and Technology*, vol. 6, no. 11, pp. 27–28, 2011.
[12] C. B. Liu, L. Shi, and B. Liu, "Utility-based bandwidth allocation for triple-play services," in *Proc. 4th European Conference on Universal Multiservice Networks*, Toulouse, France, 2007, pp. 327–336. doi: 10.1109/ECUMN.2007.58.
[13] Y. Q. Jin, X. D. Xu, Y. T. Wang, et al., "Multi-QoS mobile services guaranteed resource allocation with effective capacity," in *IEEE 3rd International Conference on Computer and Communications*, Chengdu, China, 2017.
[14] Z. J. Hao, X. D. Xu, and L. J. Li, "System utility based: resource allocation for multi-cell OFDM system," *Journal of China Universities of Posts and Telecommunications*, vol. 17, no. 2, pp. 14–19, 2010.
[15] J. Y. Cao and L. Qiu, "An effective capacity-based hybrid service resource allocation algorithm," *Journal of University of Chinese Academy of Sciences*, vol. 31, no. 11, pp. 685–690, 2014.

Manuscript received: 2018-02-28

Biographies

JIN Yaqi (yqjin@bupt.edu.cn) received the B.S. degree in communication engineering from Harbin Institute of Technology (HIT), China in 2015. She is currently pursuing the M.S. degree in information and communication engineering with Beijing University of Posts and Telecommunications (BUPT), China. Her research interests cover wireless communication, QoS guaranteed techniques, resource management including resource allocation and load balancing. She has published several papers at international conferences.

XU Xiaodong (xuxiaodong@bupt.edu.cn) received his B.S. degree in information and communication engineering and master's degree in communication and information system from Shandong University, China in 2001 and 2004 respectively. He received his Ph.D. degree of circuit and system from Beijing University of Posts and Telecommunications (BUPT), China in 2007. He is currently a professor of BUPT. He has coauthored seven books and more than 120 journal and conference papers. He is also the inventor or co-inventor of 37 granted patents. His research interests cover network architecture, moving network, coordinated multi-point and mobile network virtualization. His reasearch is supported by Beijing Nova Programme on Mobile Networking.

TAO Xiaofeng (taoxf@bupt.edu.cn) received the B.S. degree in electrical engineering from Xi'an Jiaotong University, China in 1993, and the M.S.E.E. and Ph.D. degrees in telecommunication engineering from Beijing University of Posts and Telecommunications (BUPT), China in 1999 and 2002, respectively. He was a visiting professor with Stanford University, USA from 2010 to 2011, was the Chief Architect with the Chinese National FUTURE Fourth-Generation (4G) TDD Working Group from 2003 to 2006, and established the 4G TDD CoMP Trial Network in 2006. He is currently a professor with BUPT and a fellow of the Institution of Engineering and Technology. He is the inventor or co-inventor of 50 patents and the author or co-author of 120 papers in 4G and beyond 4G. He is currently involved in fifth-generation networking technology and mobile network security.

DOI: 10.3969/j.issn.1673-5188.2018.02.004
http://kns.cnki.net/kcms/detail/34.1294.TN.20180508.1129.002.html, published online May 8, 2018

Energy-Efficient Wireless Backhaul Algorithm in Ultra-Dense Networks

FENG Hong, LI Xi, ZHANG Heli, CHEN Shuying, and JI Hong
(Key Laboratory of Universal Wireless Communication, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract

Ultra-dense networks (UDNs) are expected to be applied for the fifth generation wireless system (5G) to meet the requirements of very high throughput density and connections of a massive number of users. Considering the large amount of small base stations (SBSs), how to choose proper backhaul links is an important problem under investigation. In this paper, we propose a wireless backhaul algorithm to find an effective backhaul method for densely-deployed SBSs and to maximize energy efficiency of the system. We put forward adaptive backhaul methods of indirect and direct modes. The SBS can select the direct backhaul which connects to the macro base station (MBS) directly, or the indirect backhaul which selects an idle SBS as a relay based on the backhaul channel condition. The algorithm also allocates network resources, including the power of SBSs and system bandwidth, to solve the serious interference problem in UDN. Finally, the simulation results show that the proposed wireless backhaul algorithm has desired performance to achieve higher energy efficiency with required data rate.

Keywords

UDN; wireless backhaul; energy efficiency; recourse allocation

1 Introduction

The fifth generation wireless system (5G) is predicted to be commercialized by 2020 [1]. In order to meet the expected 1000 times increase in capacity of existing cellular networks, experts believe that technology innovation is necessary [2]. Ultra-dense networks (UDNs) are proposed to improve system throughput by densely deploying small base stations (SBSs) in the network. On the other hand, it brings huge backhaul traffic load, especially for wireless links. It also indicates more energy consumption of the backhaul process for huge traffic.

Energy efficiency is an important indicator of network performance and a challenge for UDN, which has been studied by many researchers. For examples, in [3], the network energy efficiency of some open channels in a small-area network is discussed and evaluated. In [4], the energy efficiency is improved by power allocation and load balance constraints. In [5], the energy efficiency of SBSs and regional energy efficiency can be improved by deploying SBSs and reducing the transmission power of macro base stations (MBSs). Most of the existing studies about UDN focus on optimizing the energy efficiency of the access network. However, the backhaul capacity is also a constraint on the overall network throughput.

With the intensive deployment of the SBSs in UDN, most of the user devices choose to access the SBS rather than MBS in order to obtain a higher data rate, so the number of small cell user equipment (SUE) is much larger than that of macro cell user equipment (MUE). Furthermore, due to the short inter site distance (ISD) between SBSs, the interference environment and resource allocation are quite complex for wireless backhaul transmission. When the limited capability of SBSs and energy consumption during the transmission are considered, how to choose proper backhaul routing is an important problem and still under investigation by researchers. In [6], backhaul constraint is considered as an important constraint for network performance while modeling a heterogeneous network.

Some research in backhaul has been published. In [7], a new wireless backhaul architecture of dense small cell network is proposed, which optimizes the backhaul path selection and wireless backhaul link scheduling. And in [8], the requirement of the backhaul traffic is evaluated, which indicates that the cooperation between MBSs and SBSs could improve the network performance. In [9], the resource management for wireless backhaul is studied to optimize performance of UDN. In [10], a joint optimization mechanism for forward and backhaul links is proposed which takes into account both the transmission power

This work is jointly supported by the National Natural Science Foundation of China under Grant Nos. 61771070 and 61671088 and the National Science and Technology Major Project under Grant No. 2016ZX03001017.

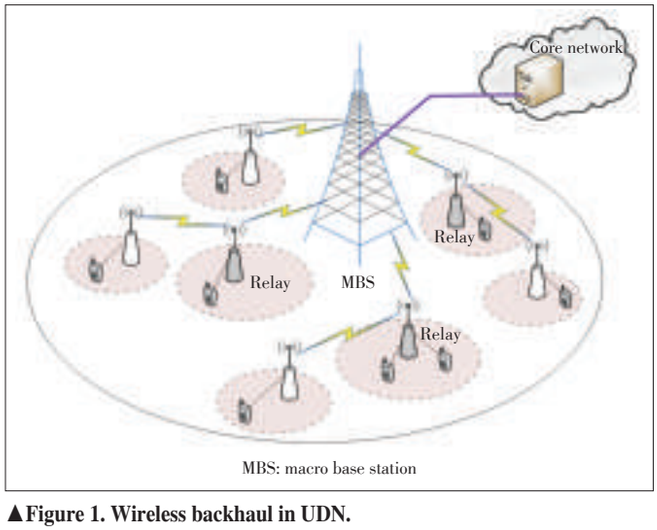
and the backhaul data rate. However, the wireless backhaul transmission with consideration on energy efficiency is still an open issue and needs further investigation.

In this paper, we propose a wireless backhaul algorithm in UDN to maximize the system energy efficiency. The proposed algorithm considers the channel condition, the backhaul rate requirement and the interference factors between SBSs for selecting an appropriate backhaul mode. There are two types of backhaul modes. One is that the SBS connects to the MBS directly, which is denoted as direct backhaul. The other is indirect backhaul in which the backhaul data is sent to the relay base station acted by surrounding free SBSs and then to the MBS. In [11], the advantages of relay in energy efficiency is described. Meanwhile, Combining the content of [3]–[5], the subchannels and transmit power are allocated to the backhaul nodes along the route, to meet the data requirement and enhance the overall network energy efficiency. Simulation results have proved the effectiveness of the proposed algorithm.

2 System Model

We consider a two-tier heterogeneous UDN scenario shown in **Fig. 1**. It contains a MBS and several SBSs densely deployed in the coverage. In this paper, the SBS with user accessing and backhaul requirement is defined as the backhaul small base station (BSBS), and the SBS without user accessing is defined as the idle small base station (ISBS). The MBS communicates with the core network through a fiber link. The BSBS obtains the data information required by the user through the wireless backhaul link with the MBS for the cost of the optical fiber between a lot of SBSs and the MBS is expensive. [12] We focus on planning the wireless backhaul link of the BSBS in a centralized manner. Also, we summarize the data rate requirement required by the BSBS users as the backhaul rate requirement of the BSBS.

The BSBS for possible backhaul routing is shown in **Fig. 2**.



▲**Figure 1. Wireless backhaul in UDN.**

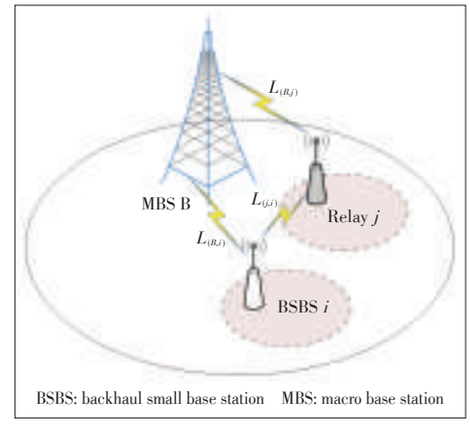


Figure 2.▶ Backhaul link selection for a BSBS.

We assume that there are n sub-channels in the system, denoted by the set K as: $K = \{k_1, k_2, \cdots, k_n\}$. Only one MBS which is denoted as B and a total of z SBSs are in the system. It is assumed that at some points, the number of BSBSs in the system is p, which is expressed as the set I, $I = \{i_1, i_2, ..., i_p\}$. The number of free base stations is q, which is expressed by the set J, $J = \{j_1, j_2, ..., j_q\}$. As shown in Fig. 2, the base station i is a BSBS and the base station j is an ISBS. The backhaul data, which is required by any UE access to i, needs transmitting from i to the MBS. One way is direct backhaul and the wireless link between B and i is denoted as $L_{(B,i)}$). Another way is indirect backhaul where j acts as a relay base station through the wireless backhaul link; the wireless link between B and j is denoted as $L_{(B,j)}$ and the wireless link between j and i is denoted as $L_{(j,i)}$.

MBS is used here as a centralized control node for the backhaul routing design for deployment convenience. It plays the role of backhauling management for all the SBSs within its coverage, focusing on backhaul data rate requirement, wireless link interference and resource allocation. Moreover, it is assumed that the base station i in the system could only select one method from the direct and indirect backhaul modes for backhaul.

3 Problem Formulation

3.1 Problem Description

In the network, the BSBS may be far from the MBS in Euler distance with poor quality channel, or the interference of backhaul link may be serious, resulting in that the BSBS is not able to meet the backhaul rate demand of users. In such cases, we will choose a proper ISBS as relay for backhaul.

The direct mode for transmitting data back to the core network is simple and easy. However, the available resource is limited and its backhaul network throughput is lower than that of the indirect mode. Although the indirect mode is more complicated with higher cost than the direct mode, we can organize the backhaul network well. In the indirect mode, more resourc-

es are available so that the network throughput can be greatly improved. At the same time, the energy efficiency of the network can be improved by making full use of the resources reasonably.

The association matrix **X** is defined to characterize whether a wireless backhaul link is established between the base stations. In the association matrix **X**, $x_{(B,i)}=1$ indicates that a wireless backhaul link, $L_{(B,i)}$, is established between B and i, which is a direct backhaul mode. $x_{(B,j)}=1$ and $x_{(j,i)}=1$ mean that there are wireless backhaul links established between B and j, as well as between j and i. It is an indirect backhaul mode.

The association matrix **A** is defined to characterize whether particular sub-channel resource is occupied by a particular wireless backhaul link. In the association matrix **A**, $a^k_{(B,i)}=1$ indicates that B occupies the sub-channel k when transmitting the data to i through $L_{(B,i)}$, and vice versa $a^k_{(B,i)}=0$ means that the sub-channel k is not occupied by B when transmitting the data to i through $L_{(B,i)}$.

3.2 Problem Modeling

Based on the assumption, it can be deduced that on $L_{(B,i)}$, when B transmits data through the sub-channel k, i receives the signal to interference plus noise ratio (SINR) which is:

$$SINR^k_{(B,i)}=x_{(B,i)}\frac{a^k_{(B,i)}p^k_{(B,i)}h^k_{(B,i)}}{\sum_{j\in J}x_{(j,i)}a^k_{(j,i)}p^k_{(j,i)}h^k_{(j,i)}+\sigma^2}, \tag{1}$$

where $p^k_{(j,i)}$ is the transmit power of j on the sub-channel k when the wireless backhaul link is established by j, which is limited by the maximum transmit power of the SBS. $h^k_{(j,i)}$ represents the channel gain on the sub-channel k of the wireless backhaul link between j and i. The sum in the denominator is the interference from the surrounding SBSs when i receives the data transmitted from B by the sub-channel k. σ^2 is additive white Gaussian noise (AWGN), which can also be written as $\sigma^2=N_0B$.

Based on (1), the rate of i provided by B can be expressed as:

$$r_{(B,i)}=\sum_k b\log_2(1+SINR^k_{(B,i)}), \tag{2}$$

where b represents the bandwidth of the sub-channel.

Similarly, the data rate on $L_{(B,j)}$ can be expressed as:

$$r_{(B,j)}=\sum_k b\log_2(1+SINR^k_{(B,j)}). \tag{3}$$

When j receives the data transmitted by B through the sub-channel k, the SINR is:

$$SINR^k_{(B,j)}=x_{(B,j)}\frac{a^k_{(B,j)}p^k_{(B,j)}h^k_{(B,j)}}{\sum_{j\in J}x_{(j,i)}a^k_{(j,i)}p^k_{(j,i)}h^k_{(j,i)}+\sigma^2}. \tag{4}$$

Likewise, the data rate on $L_{(j,i)}$ is:

$$r_{(j,i)}=\sum_k b\log_2(1+SINR^k_{(j,i)}). \tag{5}$$

The SINR of i on sub-channel k is:

$$\begin{cases} SINR^k_{(j,i)}=x_{(B,j)}\dfrac{a^k_{(B,j)}p^k_{(B,j)}h^k_{(B,j)}}{I^k_{(j',i)}+I^k_{(B,i)}+I^k_{(B,j)}+\sigma^2} \\ I^k_{(j',i)}=\sum_{j'\neq j}x_{(j',i)}a^k_{(j',i)}p^k_{(j',i)}h^k_{(j',i)} \\ I^k_{(B,i)}=\sum_i x_{(B,i)}a^k_{(B,i)}p^k_{(B,i)}h^k_{(B,i)} \\ I^k_{(B,j)}=\sum_j x_{(B,j)}a^k_{(B,j)}p^k_{(B,j)}h^k_{(B,j)} \end{cases}. \tag{6}$$

The first sum in the denominator, $I^k_{(j',i)}$, in (6) indicates that i is disturbed by the other relay base stations on the same sub-channel. The second and third summations, $I^k_{(B,i)}$ and $I^k_{(B,j)}$, mean that i is disturbed by B. In all the backhaul links $L_{(B,j)}$ and $L_{(B,i)}$, up to one backhaul link occupies the sub-channel k, so either the second or third summation must be zero.

It can be analyzed that in the whole network, the total wireless backhaul throughput is provided by B and the relay base stations. By (1)–(6), the total throughput in the system can be calculated as:

$$R=\sum_i r_{(B,i)}+\sum_j r_{(B,j)}+\sum_j\sum_i r_{(j,i)}. \tag{7}$$

The power consumed by B can be expressed as:

$$P_B=\sum_k\left(\sum_i x^k_{(B,i)}a^k_{(B,i)}p^k_{(B,i)}+\sum_j x^k_{(B,j)}a^k_{(B,j)}p^k_{(B,j)}\right), \tag{8}$$

where the left half of the plus sign indicates the total power which is consumed by B to provide the data rate for i, and the right half is expressed as the power consumed on the wireless backhaul link of j. It must be ensured that B only provides service for only one SBS on a sub-channel when allocating sub-channels.

Similarly, the power consumed by j can be expressed as:

$$P_j=\sum_k\sum_i x^k_{(j,i)}a^k_{(j,i)}p^k_{(j,i)}. \tag{9}$$

It can be seen that the total energy consumption in the system is the sum of the power consumption of the wireless backhaul link on the MBS and the relay base stations:

$$P_{total}=P_B+\sum_j P_j. \tag{10}$$

The optimization goal is the overall energy efficiency of the system backhaul:

$$\arg\max\frac{R}{P_{total}}. \tag{11}$$

The system energy efficiency is constrained by the following:

$$C_1: r_{(B,i)} + \sum_j r_{(j,i)} \geqslant r_i^{req}, \forall i, \tag{12}$$

$$C_2: r_{(B,j)} \geqslant \sum_i r_{(j,i)}, \forall j, \tag{13}$$

$$C_3: P_j \leqslant P_j^{\max}, \forall j, \tag{14}$$

$$C_4: P_B \leqslant P_B^{\max}, \tag{15}$$

$$C_5: 1 - x_{(B,i)} = x_{(j,i)}, \forall i, \exists j, \tag{16}$$

$$C_6: x_{(j,i)} = x_{(B,j)}, \forall j, \exists i, \tag{17}$$

$$C_7: \sum_i x_{(B,i)} a_{(B,i)}^k + \sum_i x_{(B,j)} a_{(B,j)}^k \leqslant 1, \forall k, \tag{18}$$

$$C_8: x_{(j,i)}, x_{(B,j)}, x_{(B,i)} \in \{0,1\}, \forall j,i, \tag{19}$$

$$C_9: a_{(B,i)}^k, a_{(B,j)}^k, a_{(j,i)}^k \in \{0,1\}, \forall j,i,k. \tag{20}$$

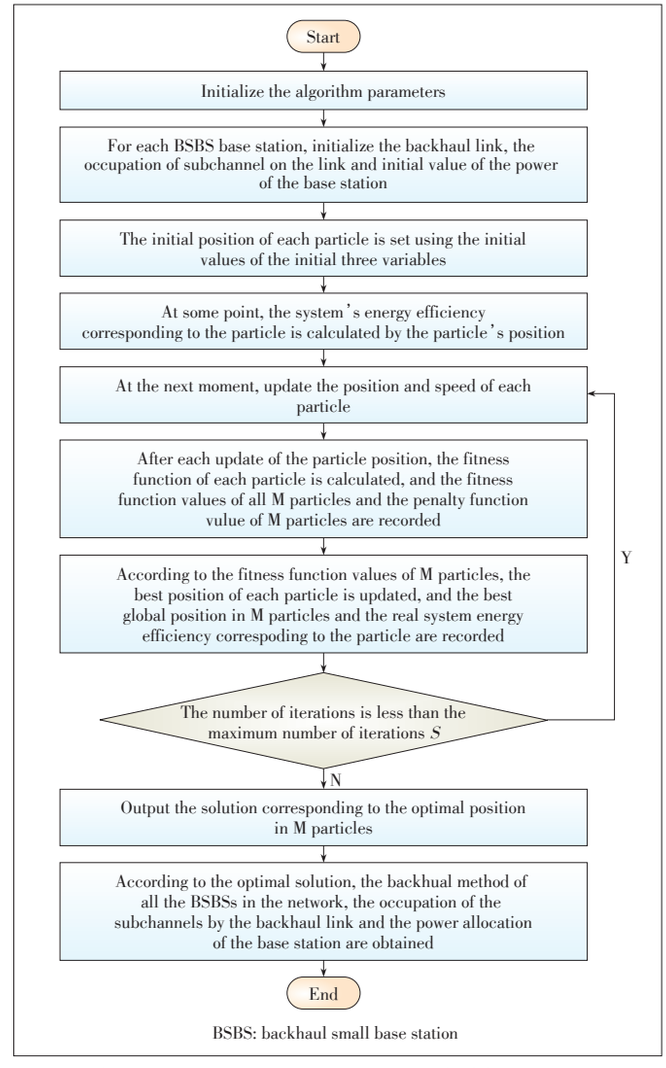
In these constraint conditions, the traffic load of all the users accessing i is expressed as r_i^{req}. C_1 indicates that it must be ensured that the data rate of the backhaul is higher than or equal to the user traffic load of i whether selecting direct backhaul or indirect backhaul. C_2 means that when j provides an indirect backhaul service for i as the relay, it is necessary to ensure that the data rate received by j is equal to or higher than the data rate sent out of j, which is reasonable and effective. When the rate on $L_{(B,j)}$ is higher than the data rate provided by j for i, j can still cache the data and then forward it to i gradually. C_3 indicates that the sum of the total power of j on each subchannel is smaller than the maximum power of the SBS. Similarly, C_4 indicates that the sum of the total power of B on each sub-channel is smaller than the maximum power of the MBS. C_5 and C_6 represent the choice of the backhaul mode of all BSBSs in UDN. C_5 indicates that i either selects direct backhaul or is provided with an indirect backhaul by at least one ISBS, j. And C_6 indicates that j must create a backhaul link with B as long as there is a BSBS receiving the backhaul data from j. C_7 indicates that any sub-channel can only be occupied at most once on the wireless backhaul link of B. C_8 and C_9 mean that the establishment of the radio backhaul link and the occupancy of backhaul link to the sub-channels are binary discrete variables. The value of these variables can only be 0 or 1.

4 Problem Resolving

The problem established by (11) to (20) is the joint optimization problem of backhaul mode selection, base station power allocation and sub-channel assignment. It is a non-deterministic polynomial-hard (NP-hard) problem [13]. Considering the computational cost, we introduce the quantum behavior particle swarm optimization (QPSO) algorithm [14] to solve the problem.

Fig. 3 shows the problem solving based on the QPSO algorithm. The concrete steps are described as follows:

1) The parameters for QPSO are initialized, including particle position, the total number of particles, and the maximum number of iterations. Each particle location includes three parts of information: the backhaul mode selection results of all base stations in the system, the occupancy results of all sub-channels on the wireless backhaul link, and the power allocation results of the base stations on each sub-channel. The total number of particles is M and the maximum number of iterations is S.
2) An initial value, $\mathbf{X}_m(0)(m = 1, \ldots, M)$, is assigned to all variables of each particle. The zero-variable for the backhaul link establishment and sub-channel occupancy is set by random initialization, then compared to 0.5. If the random number is larger than 0.5, it is set to 1 and if less than 0.5, it is set to 0. The transmit power of the base station is assigned by taking a random value between 0 and the maximum pow-



▲**Figure 3. The problem resolving based on the QPSO algorithm.**

er. We define the initial value of every parameter in every particle in this step.

3) All the particles are iterated. In the process of iteration, the energy efficiency of the system corresponding to the particle is calculated by the position of each particle at a certain time. We get the first indecisive result in this step.
4) The position and velocity of each particle are updated according to the characteristics of the QPSO algorithm. The next position of the particle is determined by the current position and velocity of the particle. We change the location of every particle to find the better solution.
5) After the particle is updated, the fitness function of each particle is calculated and the fitness function value and the corresponding penalty function value of all M particles are recorded. In this step, we get the result of each particle.
6) The optimal position of each particle is updated according to the fitness function of M particles. The global optimal position and the optimal position corresponding to the real energy efficiency maximum of M particles are recorded. According to the result of each particle, we find the best result in these particles.
7) It is determined whether the number of iterations s reaches the maximum number of iterations S. Reaching the maximum number of iterations makes the output $\mathbf{X}_m^*$ and the corresponding solution of the optimal position of M particles $\mathbf{G}(s)$, then we go to step (8); otherwise, return to step (5). We loop the iteration of these particles to get better results.
8) According to $\mathbf{X}_m^*$, the optimal solution of the optimal position $\mathbf{G}(s)$, the backhaul mode of all the BSBSs, the occupancy of all the sub-channels on the backhaul link, and the power allocation results of the base stations on each sub-channel are obtained. We finally get the best solution.

5 Simulation Results

The simulation scenario is shown in Fig. 1. The MBS can cover an entire 100 m × 100 m network, and the SBSs coverage radius is 40 meters.

The wireless channel model includes two parts, small-scale Rayleigh fading and large-scale path loss, which can be expressed as $h_{(j,i)}=h_0^2 d^{-\alpha}$. h_0 is the complex Gaussian channel coefficient, d is the distance between the BSBS and the MBS or relay base station, and the path loss factor $\alpha = 4$. The specific simulation parameters are shown in **Table 1**.

The backhaul method proposed in this paper is a combined one of the direct and indirect backhaul, while the most existing centralized backhaul in the real networks uses the direct mode. Therefore, we choose a direct backhaul algorithm to compare with the proposed wireless backhaul algorithm to verify the performance of the proposed algorithm.

It can be seen from **Fig. 4** that the proposed algorithm has converged. As the number of iterations increases, the maximum system energy efficiency is increasing. This is because with the operation of the quantum particle swarm optimization algorithm, the current optimal solution is constantly updated, and the energy efficiency of the system obtained by the current optimal solution also increases, indicating that the particle continuously approximates the suboptimal solution of the QPSO algorithm. The algorithm has reached the maximum system energy efficiency at about 700th iteration.

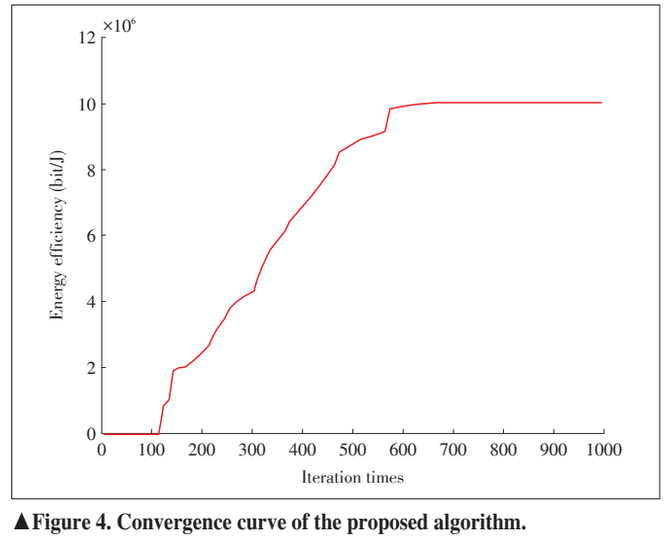
Fig. 5 shows the relationship between the number of BSBSs and the system energy efficiency. The number of BSBSs increases from 5 to 10 and the total number of SBSs in the system grows from 10 to 20, so that each BSBS has one or two ISBSs that can serve as its relay. The total energy efficiency of the system decreases gradually with the increase in the number of BSBSs. The system energy is consumed by the transmission of the MBS and the relay base stations on their respective wireless backhaul links.

When the number of BSBSs is small, the energy consumption of the system is almost from the MBS, which results in

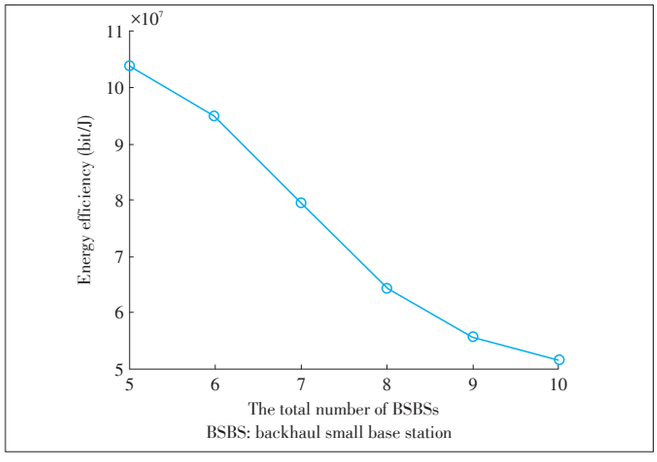
▼**Table 1. System simulation parameters**

Parameters	Value
MBS maximum transmit power	1 W
SBS maximum transmit power	0.3 W
MBS coverage	100 m
SBS coverage	40 m
The total number of SBSs	10–20
BSBS backhaul rate requirement	10 Mbit/s
System bandwidth	10 M
The number of sub-channels	20
Sub-channel bandwidth	500 kHz
Noise power spectral density	-174 dBm/Hz
Path loss factor	4

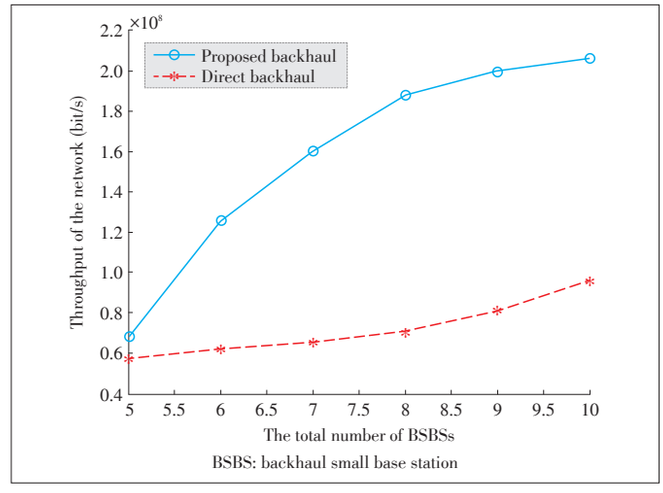
BSBS: backhaul small base station SBS: small base station
MBS: macro base station



▲**Figure 4. Convergence curve of the proposed algorithm.**



▲Figure 5. Relationship between the number of BSBSs and the system energy efficiency.

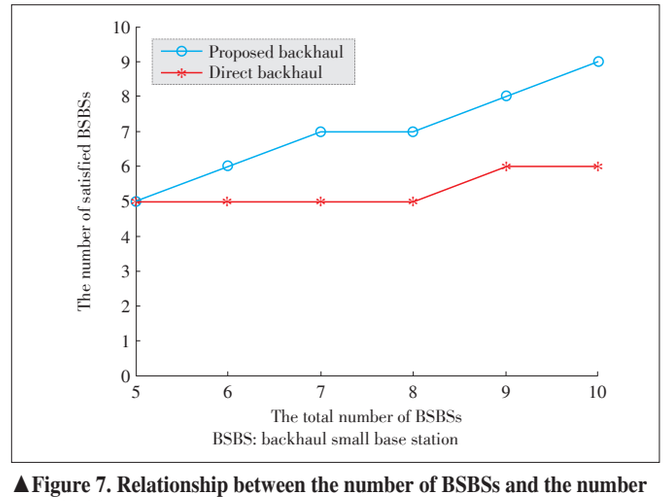


▲Figure 6. Relationship between the number of BSBSs and the system throughput.

high energy efficiency with higher throughput and lower energy consumption in the system. With the gradual increase in the number of BSBSs, the transmission power of the MBS could not meet the backhaul rate requirement of those BSBSs with poor channel quality, so that the energy consumption of the system also occurs in the relay base stations; the system energy consumption relatively increases with a large amount. Compared to the energy consumption, the increase of system throughput is relatively small, so that the system energy efficiency gradually reduces. In spite of this, the algorithm achieves the highest energy efficiency in the system by optimizing the backhaul mode and system resource allocation under the premise of ensuring the BSBS backhaul rate requirement at the specific time, the specific BSBS number and backhaul rate requirement.

Fig. 6 shows the relationship between the number of BSBSs and the system throughput. As can be seen in Fig. 6, using whether the direct backhaul or the proposed backhaul, the system throughput increases with the number of BSBSs increasing. Assuming that the backhaul rate requirement of each BSBS in the system is the same, with the increase in the number of BSBSs, the total backhaul rate requirement of the system will gradually increase, so that using whether the direct backhaul or the proposed backhaul, the system throughput will gradually increase. However, the rate of increase in system throughput in the proposed backhaul mode is greater than that in the direct backhaul mode, because there are some cases that the quality of the channels between these BSBSs and the MBS is poor in the network. The proposed backhaul can also provide the backhaul service for the BSBS by the relay base station, which makes the system throughput growth rate even greater. It can be seen that the proposed backhaul provides greater system throughput as the number of BSBSs increases compared to the direct backhaul.

Fig. 7 shows the relationship between the number of BSBSs and the number of BSBSs that meet the backhaul rate requirement. Likewise, the direct backhaul is compared to the proposed backhaul. With the increase in the total number of BSBSs, the number of BSBSs that meet the backhaul rate requirement is increasing whether using the direct backhaul or proposed backhaul. When the number of BSBSs is small, the total backhaul traffic load in the system is not very large. And the MBS is sufficient to meet the demand of the total backhaul rate of the whole system. Therefore, both backhaul modes can meet the requirement of all BSBS backhaul tasks, such as in the case that the number of BSBSs is 5. When the number of BSBSs increases, the total backhaul flow load increases in the system. The MBS may not be enough to support the gradually increasing backhaul traffic load. At this time, the ISBSs serve as relay base stations for BSBSs by proposed algorithm. Although the backhaul rate of some BSBSs can't be satisfied using whether the direct backhaul or proposed backhaul algorithm, the number of BSBSs satisfied by proposed backhaul is greater than by direct backhaul. It can be seen that the pro-



▲Figure 7. Relationship between the number of BSBSs and the number of BSBSs that meet the backhaul requirements.

posed backhaul performs better on supporting the backhaul traffic load of the entire system.

6 Conclusions

In this paper, a wireless backhaul algorithm for maximizing energy efficiency is proposed in UDN. We propose a model of combining direct backhaul and indirect backhaul, with consideration on the limited capability of SBSs, energy consumption during the transmission, channel quality and other factors. The corresponding sub-channels and transmit power are also allocated to maximize the system energy efficiency. The problem is a mixed integer nonlinear programming problem and solved by QPSO algorithm in this paper. The simulation results show that the proposed algorithm has desired performance to guarantee the high energy efficiency of the system and to effectively meet the BSBS backhaul rate requirement in the system. In the future, other factors for wireless backhaul routing may be considered such as load balance of the relay SBSs.

Reference

[1] D. Soldani and A. Manzalini, "Horizon 2020 and beyond: on the 5G operating system for a true digital society," in *IEEE Vehicular Technology Magazine*, vol. 10, no. 1, pp. 32–42, Mar. 2015. doi: 10.1109/MVT.2014.2380581.
[2] SK Telecom, "SK Telecom's view on 5G vision, architecture, technology, and spectrum," SK Telecom white paper, 2014.
[3] X. Ge, T. Han, G. Mao, et al., "Spectrum and energy efficiency evaluation of two-tier femtocell networks with partially open channels," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 3, pp. 1306–1319, Mar. 2014. doi: 10.1109/TVT.2013.2292084.
[4] H. Zhang, S. Huang, C. Jiang, et al., "Energy efficient user association and power allocation in millimeter-wave-based ultra dense networks with energy harvesting base stations," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 9, pp. 1936–1947, Sept. 2017. doi: 10.1109/JSAC.2017.2720898.
[5] W. Wang and G. Shen, "Energy efficiency of heterogeneous cellular network," in *IEEE 72nd Vehicular Technology Conference–Fall*, Ottawa, Canada, 2010, pp. 1–5. doi: 10.1109/VETECF.2010.5594361.
[6] M. Mirahsan, R. Schoenen, H. Yanikomeroglu, G. Senarath, and N. Dung-Dao, "User-in-the-loop for hethetnets with backhaul capacity constraints," *IEEE Wireless Communications*, vol. 22, no. 5, pp. 50–57, Oct. 2015. doi: 10.1109/MWC.2015.7306537.
[7] E. Pateromichelakis, M. Shariat, A. Ul Quddus, and R. Tafazolli, "Joint routing and scheduling in dense small cell networks using 60 GHz backhaul," in *IEEE International Conference on Communication Workshop (ICCW)*, London, UK, 2015, pp. 2732–2737. doi: 10.1109/ICCW.2015.7247592.
[8] V. Jungnickel, K. Manolakis, S. Jaeckel, et al., "Backhaul requirements for inter-site cooperation in heterogeneous LTE-advanced networks," in *IEEE International Conference on Communications Workshops (ICC)*, Budapest, Hungary, 2013, pp. 905–910. doi: 10.1109/ICCW.2013.6649363.
[9] H. Zhuang, J. Chen, and D. O. Wu, "Joint access and backhaul resource management for ultra-dense networks," in *IEEE International Conference on Communications (ICC)*, Paris, France, 2017, pp. 1–6. doi: 10.1109/ICC.2017.7996390.
[10] G. Nie, H. Tian, and J. Ren, "Energy efficient forward and backhaul link optimization in OFDMA small cell networks," *IEEE Communications Letters*, vol. 19, no. 11, pp. 1989–1992, Nov. 2015. doi: 10.1109/LCOMM.2015.2472535.
[11] S. Wang, R. Ruby, V. C. M. Leung, et al., "Sum-power minimization problem in multisource single-AF-relay networks: a new revisit to study the optimality," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 11, pp. 9958–9971, Nov. 2017. doi: 10.1109/TVT.2017.2739746.
[12] H. Zhang, Y. Dong, J. Cheng, M. J. Hossain, and V. C. M. Leung, "Fronthauling for 5G LTE-U ultra dense cloud small cell networks," *IEEE Wireless Communications*, vol. 23, no. 6, pp. 48–53, Dec. 2016. doi: 10.1109/MWC.2016.1600066WC.
[13] S. A. Cook, "The complexity of theorem-proving procedures," in *Proc. Third Annual ACM Symposium on Theory of Computing*, Shaker Heights, USA, 1971. doi: 10.1145/800157.805047.
[14] J. Sun, B. Feng, and W. Xu, "Particle swarm optimization with particles having quantum behavior," in *Proc. IEEE Congress on Evolutionary Computation*, Portland, USA, 2004, pp. 325–331. doi: 10.1109/CEC.2004.1330875.

Manuscript received: 2017-12-14

Biographies

FENG Hong (fenghong@bupt.edu.cn) received the B.E. degree in communication engineering from University of Science and Technology Beijing, China in 2015. He is currently working toward his master degree in information and communication engineering at Beijing University of Posts and Telecommunications, China. His research interests include wireless networking and communications and networking in 5G. He has published a paper at Chinacom 2017 and applied a patent in UDN fields.

LI Xi (lixi@bupt.edu.cn) received the B.E. and Ph.D. degrees in communication and information system from Beijing University of Posts and Telecommunications (BUPT), China in 2005 and 2010, respectively. She is currently an associate professor with the School of Information and Communication Engineering, BUPT. She has authored or co-authored over 100 papers in international journals and conferences. Her current research interests include resource management and networking in 5G, cloud computing, edge caching, and mobile Internet. She served as the chair of special track on cognitive testbed at the Chinacom 2011, the TPC member of many international conferences, including WCNC' 16/15/14/12, PIMRC' 18/17/12, Globecom' 18/17/15, ICC' 18/17/16/15, INFOCOM' 18, and CloudCom' 15/14/13, and peer-reviewer of several academic journals, such as *IEEE Communication Letter*, *Transactions on Vehicular Technology*, *Wireless Communication Magazine*, *Journal on Selected Areas in Communications*, and *IEEE Access*.

ZHANG Heli (zhangheli@bupt.edu.cn) received the B.S. degree in communication engineering from Central South University, China in 2009, and the Ph.D. degree in communication and information systems from Beijing University of Posts and Telecommunications, China in 2014. Since 2014, she has been a lecturer with the School of Information and Communication Engineering, BUPT. She has served as the reviewer of several academic journals, such as *IEEE Wireless Communications*, *IEEE Communication Magazine*, *IEEE Transactions on Vehicular Technology*, *IEEE Communication Letters*, and *IEEE Transactions on Networking*. She participated in many national projects funded by National Science and Technology Major Project, National "863" Hightech, and National Natural Science Foundation of China, and cooperated with many corporations in research. Her current research interests include heterogeneous networks, cognitive radio, resource management, cooperative communications, small cells, and mobile cloud computing.

CHEN Shuying (sychen@bupt.edu.cn) received the B.E. degree in communications engineering from Beijing University of Posts and Telecommunications (BUPT), China in 2017. She is currently working toward her M.S. degree in electronic and communication engineering at the Key Laboratory of Universal Wireless Communication, BUPT. Her research interests include the ultra-dense network (UDN) and resource management in 5G system.

JI Hong (jihong@bupt.edu.cn) received the B.S. degree in communications engineering and the M.S. and Ph.D. degrees in information and communications engineering from Beijing University of Posts and Telecommunications (BUPT), China in 1989, 1992, and 2002, respectively. In 2006, she was a visiting scholar with the University of British Columbia, Vancouver, Canada. She is currently a professor with BUPT. She has authored over 300 journal/conference papers. Several of her papers were selected for best papers. Her research interests are in the areas of wireless networks and mobile systems, including green communications, radio access, ICT applications, system architectures, cloud computing, software-defined networks, management algorithms, and performance evaluations. She has served as the co-chair at the Chinacom' 11, and a member of the Technical Program Committee of ISCIT' 17, GC' 17 Workshops, Globecom' 16/15/14/13/12/11, ICC' 13/12/11, IEEE VTC' 12S, and WCNC' 15/12. She is serving on the editorial boards of the *IEEE Transaction on Green Communications and Networking* and the *Wiley International Journal of Communication Systems*. She has guest-edited the *Wiley International Journal of Communication Systems*, with a special issue on Mobile Internet: Content, Security and Terminal.

DOI: 10.3969/j.issn.1673-5188.2018.02.005
http://kns.cnki.net/kcms/detail/34.1294.TN.20180606.1754.002.html, published online June 6, 2018

General Architecture of Centralized Unit and Distributed Unit for New Radio

GAO Yin, HAN Jiren, LIU Zhuang, LIU Yang, and HUANG He
(Algorithm Department, ZTE Corporation, Shanghai 201203, China)

Abstract

In new radio (NR) access technology, the radio access network (RAN) architecture is split into two kinds of entities, i.e., the centralized unit (CU) and the distributed unit (DU), to enhance the network flexibility. In this split architecture, one CU is able to control several DUs, which enables the function of base-band central control and remote service for users. In this paper, the general aspects of CU-DU split architecture are introduced, including the split method, interface functions (control plane functions and user plane functions), mobility scenarios and other CU-DU related issues. The simulations show the performance of Options 2 and 3 for CU-DU split.

Keywords

NR; CU; DU; F1 interface

1 Introduction

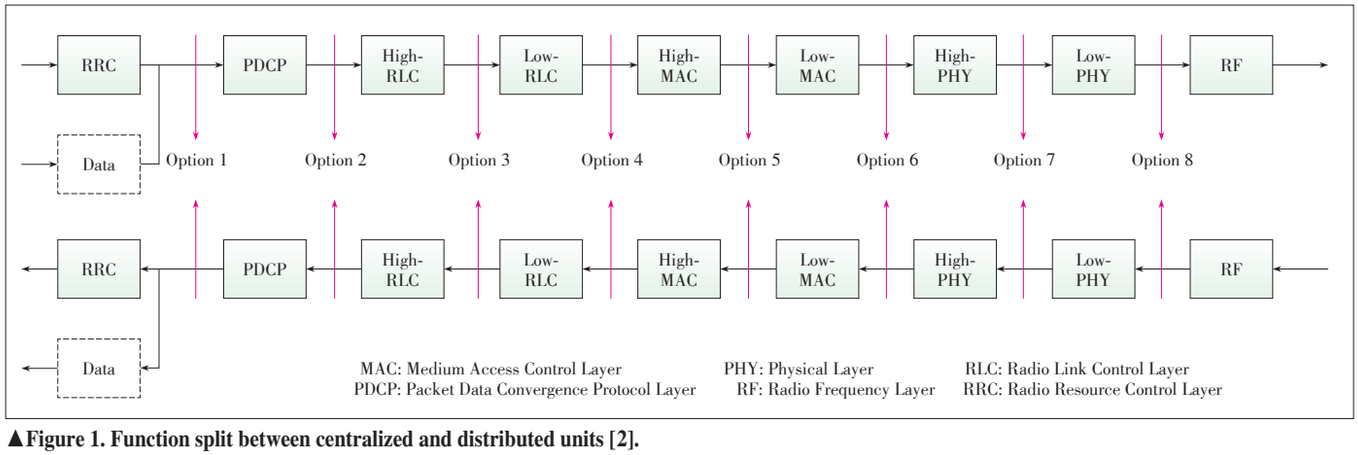
There are transport networks with performance that varies from high transport latency to low transport latency in real deployment. In order to cater for these various types of transport networks and realize multi-vendor CU-DU operation, the radio access network (RAN) architecture for new radio (NR) is split into two kinds of entities, i.e., the centralized unit (CU) and the distributed unit (DU). The latency-tolerant network function resides in the CU entity, and the latency-sensitive network function resides in the DU entity [1].

Fig. 1 shows the possible CU-DU split options [2]. Options 1, 2, 3, 4, and 5 are regarded as higher layer split variants, while Options 6, 7, and 8 are regarded as lower layer split variants in the case of CU-DU.

2 High Layer Split (HLS)

For a transport network with higher transport latency, higher layer splits may be applicable. On the other hand, for a transport network with lower transport latency, lower layer splits can also be applicable.

The choice of how to split functions in the 5G RAN architecture should offer good performance of services. The 3rd Generation Partnership Project (3GPP) agrees that there shall be nor-



▲Figure 1. Function split between centralized and distributed units [2].

mative work for a single HLS option (Option 2 or Option 3), and finally Option 2 for high layer RAN architecture split is selected because better performance with more high throughput and less latency restriction can be provided by Option 2 compared with Option 3. The detailed comparison of the two options based on simulation is shown in Section 7.

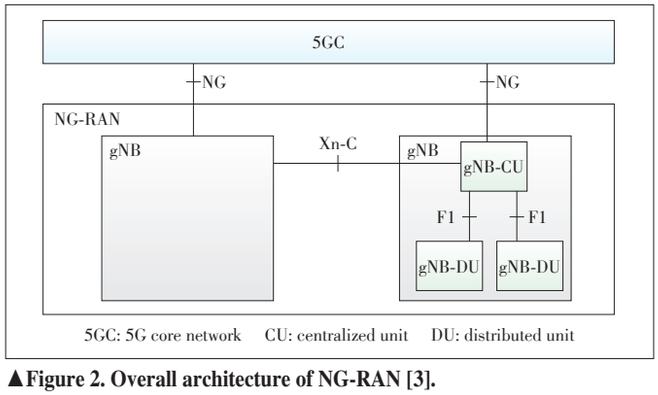
3 Overall Architecture

As shown in **Fig. 2**, in the next generation radio access network (NG-RAN), there are a set of next generation NodeBs (gNBs) connected to the 5G core network (5GC) through the NG interface, and the gNBs can be interconnected through Xn interface. For disaggregate cases, a gNB may consist of a gNB-CU and one or more gNB-DU(s), and the interface between gNB-CU and gNB-DU is called F1. The NG and Xn-C interfaces for a gNB terminate in the gNB-CU. One gNB-CU can connect to multiple gNB-DUs, and the maximum number of connected gNB-DUs is only limited by implementation.

In the 3GPP standard, one gNB-DU is supported to connect only one gNB-CU. However, a gNB-DU can be connected to multiple gNB-CUs by appropriate implementation for resiliency. Meanwhile, one gNB-DU can support one or more cells. The internal structure of the gNB is not visible to the core network and other RAN nodes, the gNB-CU and connected gNB-DUs are only visible to other gNBs and the 5GC as a gNB. With the analysis above, the following definitions of gNB-CU and gNB-DU can be obtained.

The gNB-CU is a logical node hosting Radio Resource Control Layer (RRC), Service Data Adaptation Protocol (SDAP) and PDCP protocols of the gNB or RRC and PDCP protocols of the evolved universal terrestrial radio access-new radio gNB (en-gNB) that controls the operation of one or more gNB-DUs. The gNB-CU terminates the F1 interface connected with the gNB-DU [3].

The gNB-DU is a logical node hosting Radio Link Control Layer (RLC), MAC and PHY layers of the gNB or en-gNB, and its operation is partly controlled by gNB-CU. One gNB-DU supports one or multiple cells. One cell is supported by only one gNB-DU. The gNB-DU terminates the F1 interface connected with the gNB-CU [3].



▲**Figure 2. Overall architecture of NG-RAN [3].**

4 F1 Interface Principle

The interface between gNB-CU and gNB-DU is called F1, and similar to NG or Xn interface in 5G RAN, it supports signalling exchange and data transmission between endpoints. Besides, F1 interface separates the radio network layer and the transport network layer, and it enables exchange of UE associated signalling and non-UE associated signalling. In addition, F1 interface supports control plane (CP) and user plane (UP) separation, therefore, the F1 interface functions are divided into F1-C function and F1-U function.

4.1 F1-C Function

Considering the control plane function of F1 interface, the F1 interface management, system information management, UE context management and RRC message transfer should be introduced.

The F1 interface management function mainly consists of F1 setup, gNB-CU configuration update, gNB-DU configuration update, error indication, and reset function.

The F1 setup function is responsible for the exchange of application level data between gNB-DU and gNB-CU, and it can activate the cells in gNB-DU. The F1 setup procedure is initiated by the gNB-DU. The gNB-CU configuration update and gNB-DU configuration functions are responsible for the update of application level data configuration between gNB-DU and gNB-CU. The gNB-DU configuration update can also activate or deactivate the cells in gNB-DU. Besides, the F1 setup and gNB-DU configuration update functions allow to inform the S-NSSAI (s) supported by the gNB-DU. In addition, the error indication function is responsible for indicating that an error has occurred and reset function is responsible for initializing the peer entity after node setup and after a failure event occurs.

As for system information management, the gNB-DU is responsible for system broadcast information scheduling and system information transmission. For the system information broadcast, the encoding of NR-Master Information Block (MIB) and System Information Block 1 (SIB1) is carried out by the gNB-DU, while the encoding of other system information (SI) messages is carried out by the gNB-CU.

For the sake of UE energy saving, the on-demand SI delivery is introduced over F1 interface as well. In this case, CU is in charge of processing the on-demand SI request from UE over MSG3 and sends System Information Delivery Command to tell gNB-DU broadcast the requested other SI(s), the UE is able to obtain the requested SI(s) from the gNB-DU when needed instead of monitoring the broadcast channel all the time.

The F1 UE context management function is responsible for the establishment and modification of the necessary overall UE context.

The establishment of F1 UE context is initiated by gNB-CU,

and gNB-DU can accept or reject the establishment based on admission control criteria (e.g., the resource is not available). Besides, the modification of F1 UE context can be initiated by either gNB-CU or gNB-DU. The receiving node can accept or reject the modification. Furthermore, the F1 UE context management function also supports the release of the context previously established in the gNB-DU. The release of the context is triggered by the gNB-CU either directly or following a request received from the gNB-DU. The gNB-CU requests the gNB-DU to release the UE context when the UE enters RRC_IDLE or RRC_INACTIVE.

The F1 UE context management function can also be used to manage data radio bearers (DRBs) and signaling radio bearers (SRBs), i.e., establishing, modifying and releasing DRB and SRB resources. The establishment and modification of DRB resources are triggered by the gNB-CU and accepted/rejected by the gNB-DU based on resource reservation information and QoS information to be provided to the gNB-DU. For each DRB to be setup or modified, the signal network slice selection assistance information (S-NSSAI) may be provided by gNB-CU to the gNB-DU in the UE context setup procedure and the UE context modification procedure.

The mapping between QoS flows and radio bearers is performed by gNB-CU and the granularity of bearer related management over F1 is radio bearer level. To support PDCP duplication for intra-DU carrier aggregation (CA), one data radio bearer should be configured with two GTP-U tunnels between gNB-CU and gNB-DU [4].

The RRC message transfer function is responsible for the transfer of RRC messages between gNB-CU and gNB-DU. RRC messages are transferred over F1-C, while the UE related RRC messages are transferred over the Uu interface.

4.2 F1-U Function

Considering the user plane function of F1 interface, the user data transfer and flow control should be introduced.

The user data transfer function allows the transferring of user data between gNB-CU and gNB-DU.

The flow control function allows controlling downlink user data transmission towards the gNB-DU. The function includes the transmitting procedure of DL USER DATA and DL DATAT DELIVEREY STATUS frames. There are several methods for the flow control enhancement of data transmission introduced in 3GPP standard [5].

The transfer procedure of downlink user data (DL USER DATA frame) aims to provide F1-U specific sequence number information when transferring user data carrying a DL PDCP PDU from gNB-CU to gNB-DU via the F1-U interface. For the DL USER DATA frame, in order to discard the redundant PDUs caused by the PDCP duplication, the discarded flag and the information on discarding the PDCP PDUs between a start and a stop range are added in the DL USER DATA frame, i.e., the DL discards the NR PDCP PDU SN start (first/last block) and the corresponding discarded block size (first/last block). For retransmitted data packets, a "retransmission flag" is introduced in the spare bit of DL_USER_DATA, which helps the gNB-DU to identify and handle the retransmitted packets with high priority. The gNB-CU can set the Report Polling Flag within the DL USER DATA frame to confirm DL DATA DELIVERY STATUS from the gNB-DU.

After receiving a DL USER DATA frame from the gNB-CU, the gNB-DU shall detect whether an F1-U packet is lost over the F1 interface and memorize the respective sequence number after it declares the respective F1-U packet as being "lost", and the gNB-DU shall transfer the remaining NR PDCP PDUs towards the UE and memorize the highest NR PDCP PDU sequence number of the NR PDCP PDU that has successfully been delivered in sequence towards the UE (in case RLC AM is used) and the highest NR PDCP PDU sequence number of the NR PDCP PDU that has been transmitted to the lower layers. The gNB-DU shall send the DL DATA DELIVERY STATUS if the Report Polling Flag is set.

The transfer procedure of Downlink Data Delivery Status (DL DATA DELIVEREY STATUS frame/DDDS frame) aims to provide feedback from gNB-DU to gNB-CU to allow the gNB-CU to control the downlink user data flow via the gNB-DU for the respective data radio bearer. For the DL DATA DELIVEREY STATUS frame, the highest successfully delivered/transmitted NR PDCP sequence number is added, which can help gNB-CU acquire more accurate data delivery status in the gNB-DU for RLC AM/UM mode data. For the fast data retransmission of lost PDCP PDUs caused by radio link outage, the DL DATA DELIVERY STATUS frame includes the indication of detected radio link outage/resume, together with the information on the highest NR PDCP PDU sequence number successfully delivered in sequence to the UE and the highest NR PDCP PDU sequence number transmitted to the lower layers. The gNB-DU shall indicated lost NR-U packets over the F1 interface within the DDDS frame and also set the desired buffer size for the concerned data bearer and the minimum desired buffer size for the UE within the DDDS frame.

After receiving the DL DATA DELIVERY STATUS frame from the gNB-DU, the gNB-CU shall regard the desired buffer size and the minimum desired buffer size as the amount of data desired from the gNB-DU, and remove the buffered PDCP PDUs according to the feedback of successfully delivered PDCP PDUs. The gNB-CU should also decide the actions necessary for undelivered/transmitted PDCP PDUs at the gNB-DU side, e.g., retransmitting corresponding PDUs to other available gNB-DUs when outage reported.

5 Mobility Scenarios

5.1 Intra-gNB-CU Mobility

In the Intra-gNB-CU mobility part, the standalone case and

dual connectivity case are considered.

5.1.1 Intra-NR Mobility

In this scenario, the source and target cells belong to different gNB-DUs in the same gNB-CU.

The gNB-CU makes a decision on the suitable target gNB-DU for handover, based on the UE measurement report. Then, the gNB-CU initiates the UE Context Setup procedure to assign resources on Uu and F1 for one or several RBs and to setup corresponding context for a given UE in the target gNB-DU. The target gNB-DU shall execute the requested RB configuration, and if available, stores the general UE context. At the next step, the gNB-CU sends the RRC reconfiguration message including Cell Group Config at least in the target gNB-DU to the UE. Finally, the UE sets up the RRC connection with the target gNB-DU and replies the RRC reconfiguration complete message. After the UE access to the target gNB-DU, the gNB-CU initiates the UE Context Release procedure to release the UE context in the source gNB-DU [6]. The signaling flow is shown in **Fig. 3** [3].

5.1.2 Inter-gNB-DU Mobility with Dual Connectivity

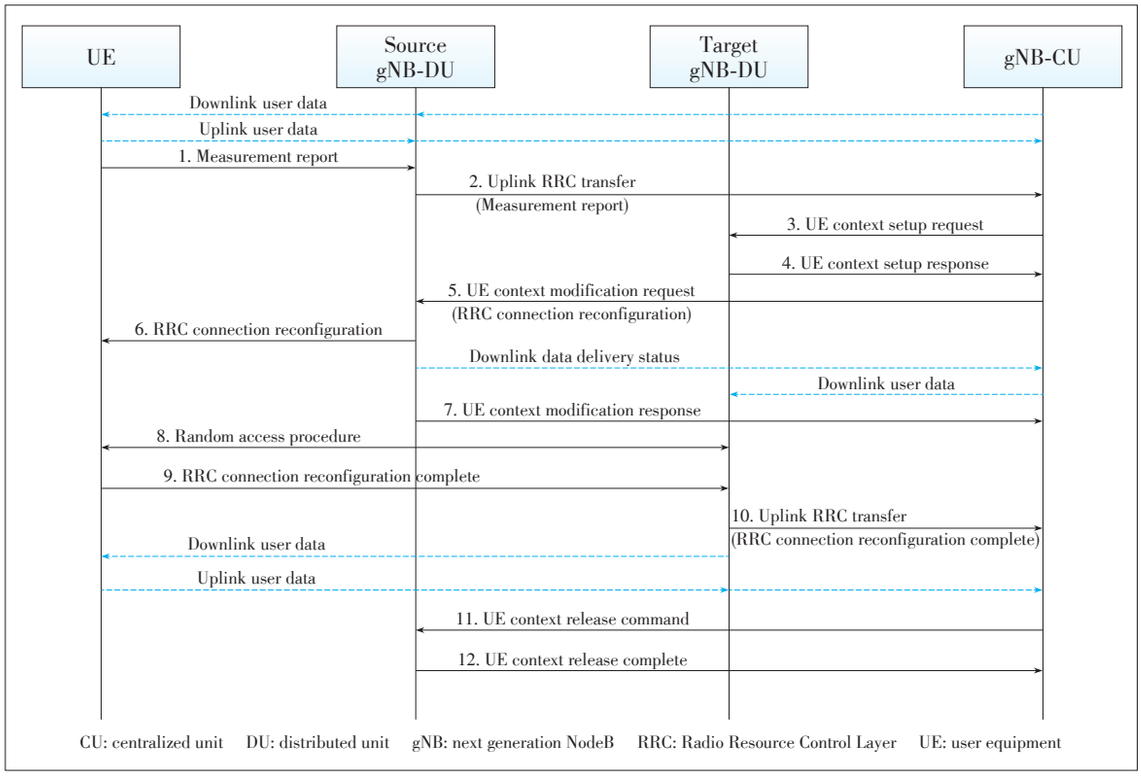
In this scenario, the source cell and the target cell belong to different gNB-DUs in the same gNB-CU, and a UE can connect with one or more gNB-DUs at the same time.

The UE moves between the cells belonging to different gNB-DUs. The gNB-CU makes a decision on suitable target gNB-DU addition based on the UE measurement report. Then, the gNB-CU initiates the UE Context Setup procedure to assign resources on Uu and F1 for one or several RBs and to setup corresponding context for a given UE in the secondary gNB-DU. The gNB-DU shall execute the requested RB configuration, and if available, stores the general UE context. At the next step, the gNB-CU sends the RRC reconfiguration message including Cell Group Config at least in the secondary gNB-DU to the UE. Finally, the UE sets up the RRC connection with the target gNB-DU (gNB-DU2) and replies the RRC reconfiguration complete message. After the UE context has established in the target gNB-DU (gNB-DU2), UE is connecting with both target gNB-DU (gNB-DU2) and source gNB-DU (gNB-DU1) at the same time. If one leg breaks during the dual connectivity, the fast centralized retransmission procedure of lost PDUs should be used [7]. This signaling flow is shown in **Fig. 4** [8].

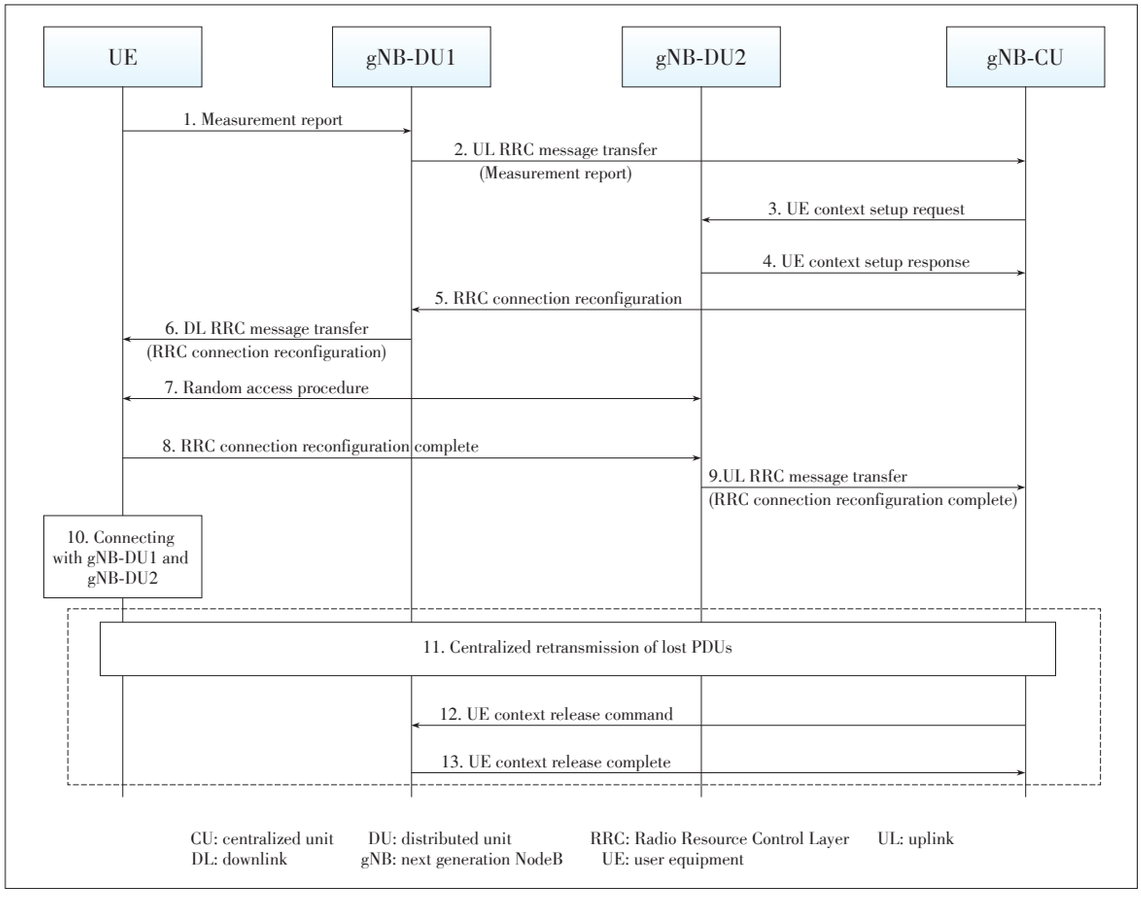
5.1.3 Evovled Universal Terrestrial Radio Access-New Radio Dual Connectivity (EN-DC) Mobility

In this scenario, the source cell and the target cell belong to different gNB-DUs in the secondary node.

The Master eNB (MeNB) makes a decision on the suitable target gNB-DU for handover, based on the UE measurement report. Then, after receiving the SgNB Modification Request message from the MeNB with SCG configuration, the gNB-CU initiates the UE Context Setup procedure to assign resources on Uu and F1 for one or several RBs and to setup corresponding context for a given UE in the target gNB-DU. The target gNB-DU shall execute the requested RB configuration, and if available,



◀Figure 3. Inter-gNB-DU mobility for intra-NR [3].



◀Figure 4. Inter-DU handover procedure with dual connectivity [8].

store the general UE Context. After that, gNB-CU sends the confirmed SCG configuration to the MeNB which needs to be transferred to UE. Finally, the UE sets up the RRC connection with the target gNB-DU. After the UE context has been established in the target gNB-DU, the gNB-CU initiates the UE Context Release procedure to release the UE context in the source gNB-DU [9]. **Fig. 5** shows the signaling flow [3].

6 Other CU-DU Related Issues

6.1 CU-DU Low Layer Split (LLS)

In addition to CU-DU HLS, lower layer split is also applicable and preferable to realize enhanced performance (e.g. centralized scheduling) for transport network with lower transport latency. In this case, the physical layer is split into LLS-CU and LLS-DU. The possible LLS options to be discussed are shown in **Fig. 6** [10].

The possible non-exhaustive functional split options (Fig. 6) for DL and UL are listed as below:

1) Option 6

All of the PHY functions reside in the DU.

2) Option 7-1

In the UL, FFT and CP removal functions reside in the LLS-DU, while the rest of PHY functions reside in the LLS-CU.

In the DL, iFFT and CP addition functions reside in the LLS-DU, while the rest of PHY functions also reside in the LLS-CU.

3) Option 7-2

In the UL, FFT and CP removal and resource de-mapping functions reside in the LLS-DU, while the rest of PHY functions reside in the LLS-CU.

In the DL, iFFT and CP addition, resource mapping and precoding functions reside in the LLS-DU, while the rest of PHY functions reside in the LLS-CU.

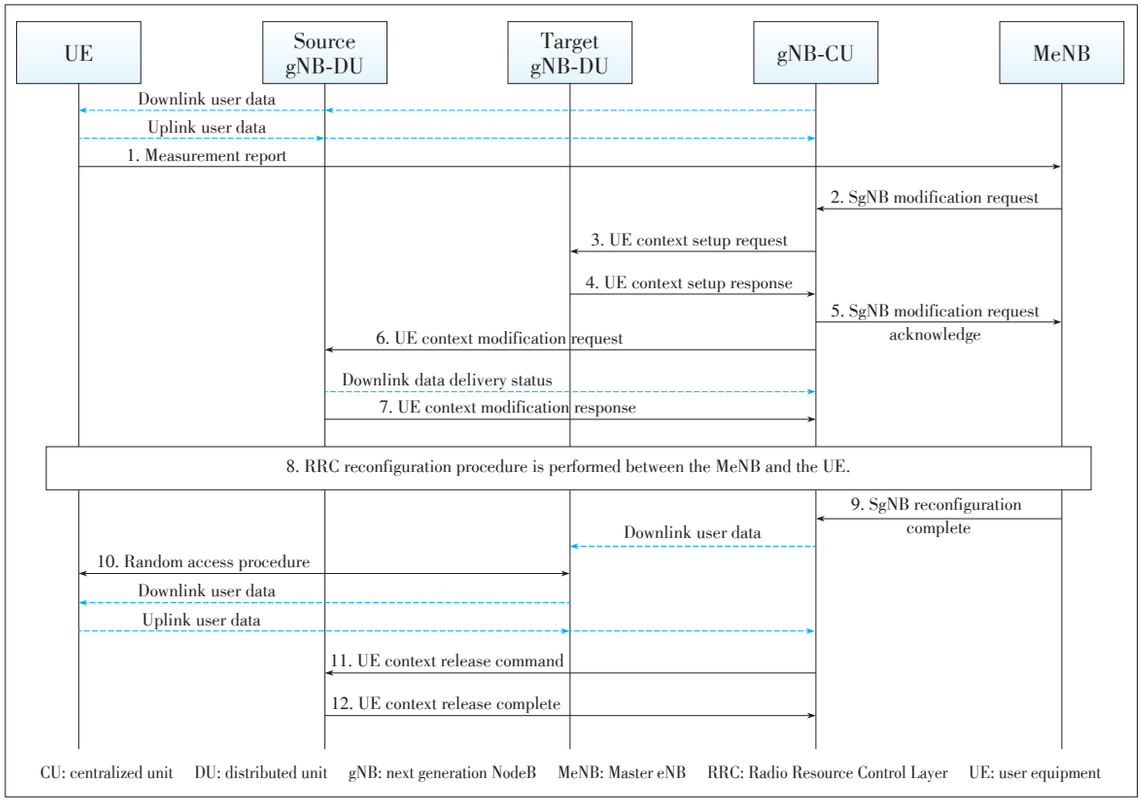
4) Option 7-3 (Only for DL)

Only the encoder resides in the LLS-CU, and the rest of PHY functions reside in the LLS-DU.

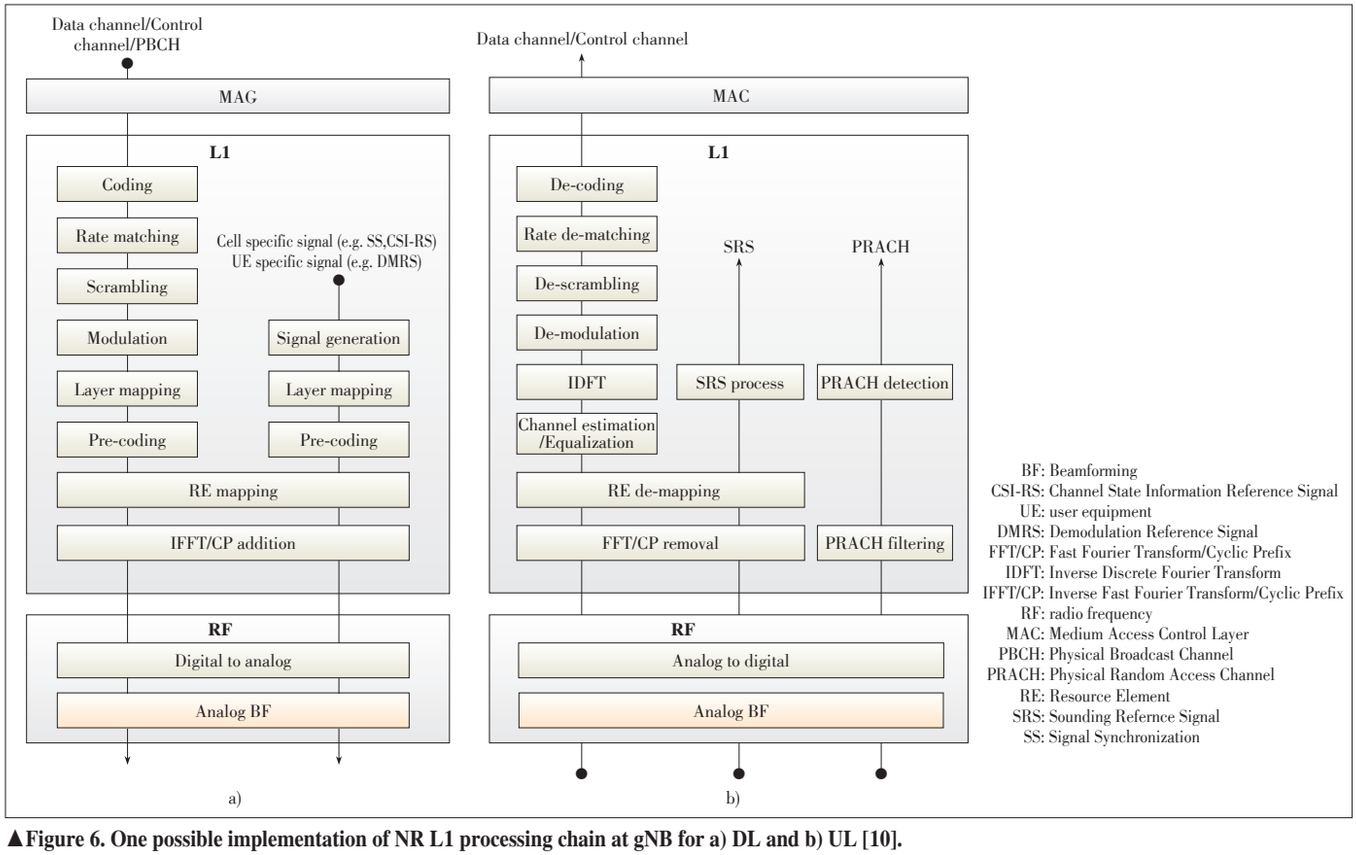
Additional potential functional split options were also considered. For the UL, there was a proposal to split between IDFT and Channel estimation/Equalization. Also, for both DL and UL, the possibility to split somewhere between Option 7-1 and Option 7-2 was proposed in light of digital beamforming [10].

6.2 Separation of Control Plane (CP) and User Plane (UP)

In order to provide the possibility of optimizing the location of different RAN functions based on the scenario and desired performance, the gNB-CU can be separated further into CU-CP



◀Figure 5. Inter-gNB-DU mobility using MCG SRB in EN-DC [3].



▲Figure 6. One possible implementation of NR L1 processing chain at gNB for a) DL and b) UL [10].

and CU-UP on the basis of HLS. The gNB-DU hosts the RLC/MAC/PHY protocols, the CU-CP hosts the control plane instance of PDCP and RRC protocols and the CU-UP hosts the user plane instance of PDCP (and SDAP) protocols [11]. The interface between CU-CP and CU-UP is named as E1. The overall RAN architecture with CU-CP and CU-UP separation is shown in **Fig. 7** [12].

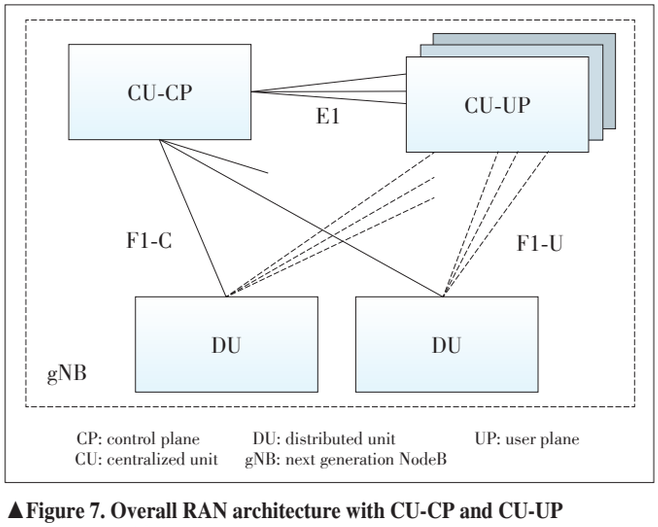
A gNB may consist of a CU-CP, multiple CU-UPs and multiple DUs. The CU-CP is connected to the DU through the F1-C interface, while the CU-UP is connected to the DU through the F1-U interface. The CU-UP is connected to the CU-CP through the E1 interface. Furthermore, one gNB-DU is connected to only one CU-CP and one CU-UP is connected to only one CU-CP. A gNB-DU or a CU-UP may be connected to multiple CU-CPs by appropriate implementation for resiliency. One gNB-DU can be connected to multiple CU-UPs under the control of the same CU-CP and one CU-UP can be connected to multiple gNB-DUs under the control of the same CU-CP.

The basic functions supported over the E1 interface include E1 interface management function and bearer management function, while such functions are still under investigation as E1 load management, E1 configuration update, inactivity detection, and new QFI notification.

6.3 CU-DU High Layer Split in E-UTRAN

In order to achieve better integration of LTE eNB with gNB, the converged architecture of LTE and NR is preferred, i.e. introducing central unit (LTE-CU) and distributed unit (LTE-DU) into Evolved UMTS Terrestrial Radio Access Network (E-UTRAN) with PDCP/RLC split (option 2). This architecture aims to utilize the transport network in an efficient way and minimize the impacts on legacy LTE transport network. It is easier for further network upgrading when LTE CU and DU are deployed in operators' networks [13].

The CU-DU high layer split in E-UTRAN follows that in NR, including the function split architecture and interface function. Similar as gNB in NR, the eNB is split into two entities, i.e., eNB-CU and eNB-DU, and the interface between eNB-CU and eNB-DU is named as V1. The V1 interface supports the same functions as the F1 interface except that some LTE features depend on operators' requirements when the eNB is connected to EPC, such as NB-IoT and eMTC.



▲**Figure 7. Overall RAN architecture with CU-CP and CU-UP separation [12].**

7 Simulations

Based on the TCP throughput efficiency of data transmission, the simulations were conducted to show the performance of Options 2 and 3 (Fig. 1) for CU-DU split.

For Option 2, RRC and PDCP are in the central unit; RLC, MAC, physical layer and RF are in the distributed unit. For Option 3, low RLC (partial function of RLC, which mainly includes the segmentation related function), MAC, physical layer and RF are in distributed unit; RRC, PDCP and high RLC (the other partial function of RLC, which mainly includes the ARQ related function) are in the central unit.

For Option 3, since the Automatic Repeat Request (ARQ) is located in CU, the RLC retransmission suffers a two-way fronthaul delay, including the delay for RLC status report and the delay for the following data retransmission. Considering the mechanism of TCP, the delay of RLC retransmission may lead to some negative impact on the throughput.

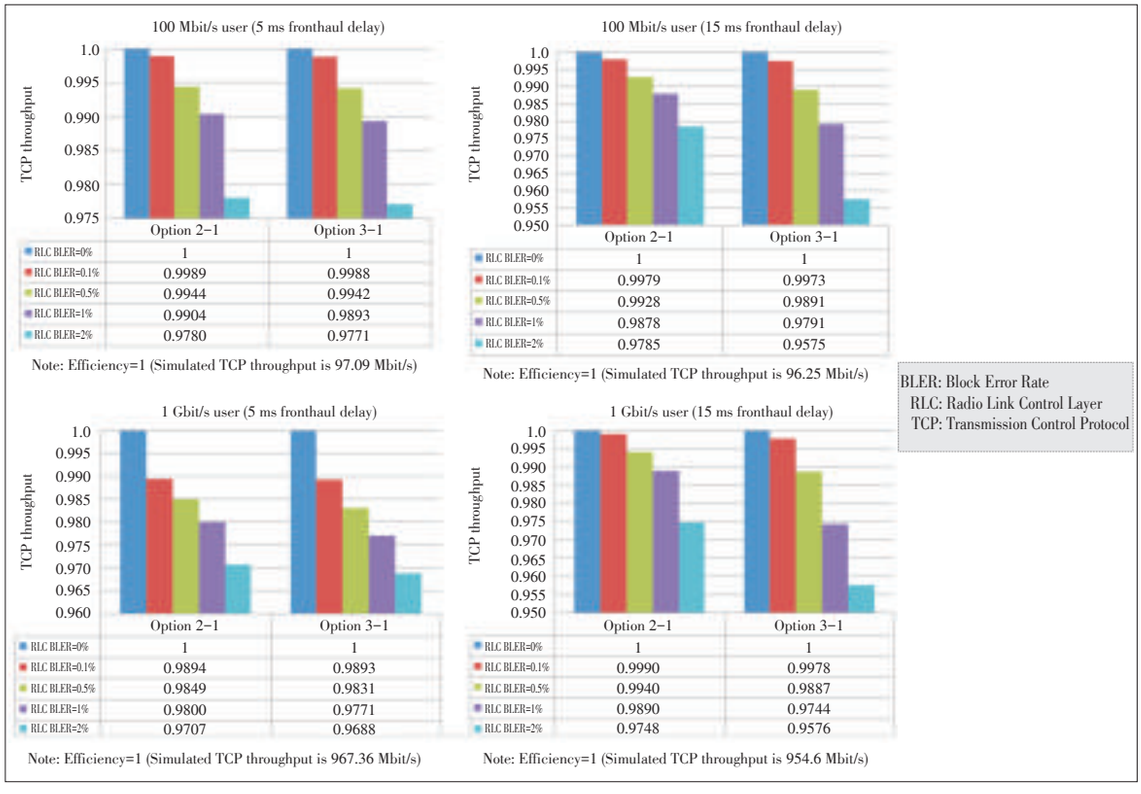
The simulation results of TCP throughput efficiency for Options 2 and 3 are shown in **Fig. 8** with different residual RLC BLER conditions. It can be observed that as the increase of the fronthaul delay, the TCP throughput will decrease. The TCP throughput of Option 3 decreases due to the additional retransmission delay from the fronthaul between CU and DU.

Slow-start is part of the congestion control strategy used by TCP. Once the slow-start threshold is reached, TCP changes from the slow-start algorithm to the linear growth (congestion avoidance) algorithm. Furthermore, if a ftp traffic model like 100 m file size and 1 G file size is used, the TCP slow start impacts performance more due to shorter simulation time. The simulation results in **Fig. 9** show that Option 2 performance is obviously better than Option 3 for short time TCP services considering TCP slow-start effects (initial TCP slow-start threshold set as 65,535).

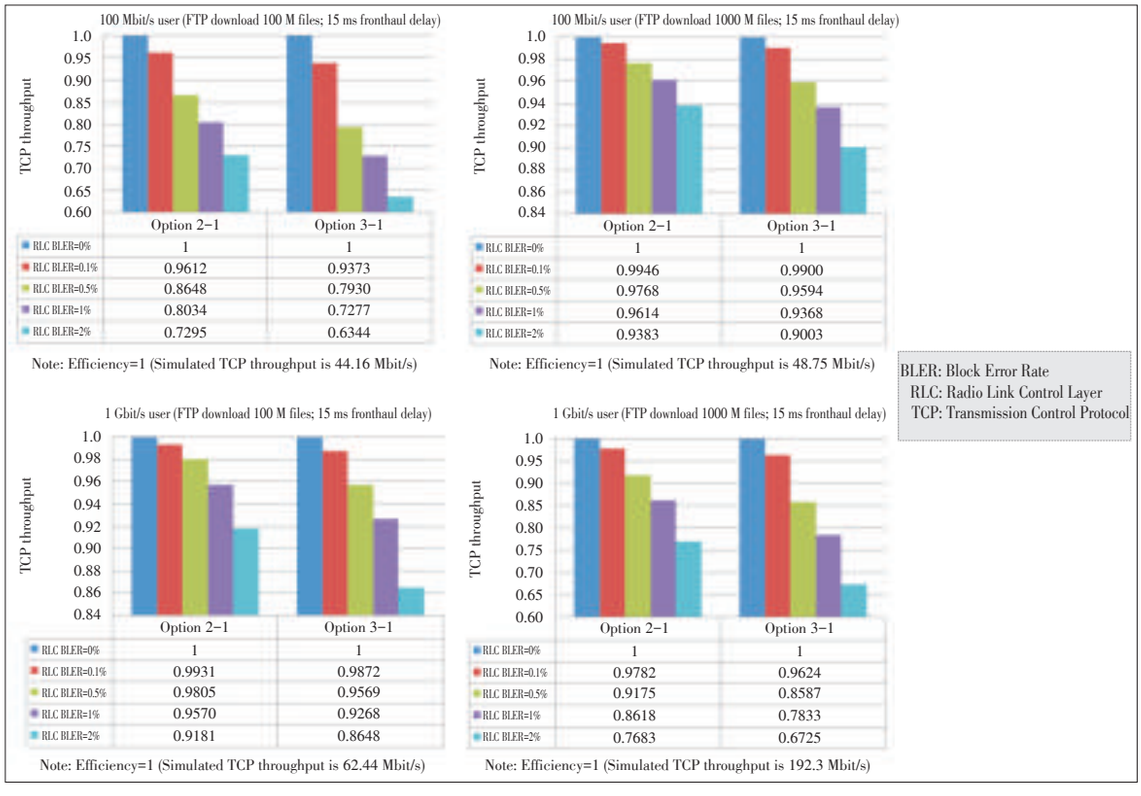
It can be seen that Option 3 introduces extra RLC retransmission delay, and the extra delay may lead to negative impact on the throughput, especially for short time TCP services considering TCP slow-start effects. Compared with Option 3, Option2 provides better performance.

8 Conclusions

In this paper, we introduce the progress of CU-DU architecture and present the architecture for CU-DU split in NG-RAN. The CU-DU interface functions and basic mobility scenarios



◀Figure 8. Simulation results on the data transmission in Options 2 and 3 (without TCP slow start impact).



◀Figure 9. Simulation results on the data transmission in options 2 and 3 (short time TCP services).

are discussed in this paper. The solutions to these challenges and potential optimization are also proposed. In addition, the other CU-DU related topics are also introduced, including CU-DU low layer split, separation of CP and UP, and the high lay-

er split in E-UTRAN.

References

[1] NTT DOCOMO, INC., "Revised WID on new radio access technology," RP-172109, 2017.
[2] *Study on new Radio Access Technology: Radio Access Architecture and Interfaces,* 3GPP TR38.801, April. 2017.
[3] *NG-RAN, Architecture Description; (Release 15)*, 3GPP TS 38.401, Jan. 2018.
[4] *NG-RAN, F1 General Aspects and Principles; (Release 15)*, 3GPP TS 38.470, Jan. 2018.
[5] *NG-RAM, NR User Plane Protocol; (Release 15),* 3GPP TS38.425, Dec. 2017.
[6] H. J. Zhang, N. Liu, X. L. Chu, et al., "Network slicing based 5g and future mobile networks: mobility, resource management, and challenges," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 138–145, Aug. 2017. doi: 10.1109/MCOM.2017.1600940.
[7] H. J. Zhang, Y. Qiu, X. L. Chu, K. Long, and V. C. M. Leung, "Fog radio access networks: mobility management, interference mitigation and resource optimization," *IEEE Wireless Communications*, vol. 24, no. 6, pp. 120–127, Dec. 2017. doi: 10.1109/MWC.2017.1700007.
[8] ZTE, "Discussion on inter-DU mobility with dual connectivity," R3-180134, 2018.
[9] H. J. Zhang, C. X. Jiang, J. L. Cheng, and V. C. M. Leung, "Cooperative interference mitigation and handover management for heterogeneous cloud small cell networks," *IEEE Wireless Communications*, vol. 22, no. 3, pp. 92–99, Jun. 2015. doi: 10.1109/MWC.2015.7143331.
[10] *Study of CU-DU Low Layer Split for NR (Release 15)*, 3GPP TS 38.816, Jan. 2018.
[11] Ericsson, New WID on separation of CP and UP for split option 2, RP-173831, 2017.
[12] *Study of Separation of NR Control Plane (CP) and User Plane (UP) for Split Option 2 (Release 15)*, 3GPP TS 38.806, Jan. 2018.
[13] China Unicom, Orange, China Telecom, Huawei, and HiSilicon, "Revised SID: study on eNB(s) architecture evolution for E-UTRAN and NG-RAN," RP-172707, Dec. 2017.

Manuscript received: 2018-02-05

Biographies

GAO Yin (gao.yin1@zte.com.cn) received the master's degree in circuit and system from Xidian University, China in 2005. Since 2005 she has been with the research center of ZTE Corporation and been engaged in the study of 3G/4G/5G technology. She has authored or co-authored about hundreds of proposals for 3GPP meetings and journal papers in wireless communications and has filed more than 100 patents. She was the rapporteurs of multiple 3GPP WIs. From August 2017, she has been elected as the vice chairman of 3GPP RAN3.

HAN Jiren (han.jiren@zte.com.cn) received the master's degree in wireless communication systems from University of Sheffield, UK in 2016. He is an advanced research engineer at the Algorithm Department, ZTE Corporation. His research interests include 5G wireless communications and signal processing.

LIU Zhuang (liu.zhuang2@zte.com.cn) received the master's degree in computer science from Xidian University, China in 2003. He is currently a senior 5G research engineer at ZTE R&D center, Shanghai. His research interests include 5G wireless communications and signal processing.

LIU Yang (liu.yang31@zte.com.cn) received the Ph.D. degree in communication and information systems from Beijing University of Posts and Telecommunications (BUPT), China in 2016. He was a visiting scholar at Department of Electrical and Computer Engineering of North Carolina State University, USA from 2013 to 2015. He is currently a 5G research engineer at ZTE R&D center, Shanghai. His research interests include 5G wireless communications and signal processing.

HUANG He (huang.he4@zte.com.cn) received the bachelor's degree in computer science and technology from Shanghai Jiao Tong University, China in 2004. He is currently the chief engineer of wireless innovation laboratory of ZTE Corporation and leads the research and standardization work on 5G RAN. He has filed more than 60 patents. He was the rapporteurs of multiple CCSA/3GPP SIs/WIs and the editors of the related protocols.

DOI: 10.3969/j.issn.1673-5188.2018.02.006
http://kns.cnki.net/kcms/detail/34.1294.TN.20180514.0918.002.html, published online May 14, 2018

Two-Codebook-Based Cooperative Precoding for TDD-CoMP in 5G Ultra-Dense Networks

GAO Tengshuang [1], CHEN Ying [2], HAO Peng [3], and ZHANG Hongtao [2]
(1. Tianjin University, Tianjin 300072, China;
2. Beijing University of Posts and Telecommunications, Beijing 100876, China;
3. Wireless Product R&D Institute, ZTE Corporation, Shenzhen 518057, China)

Abstract

In ultra-dense networks (UDN), the local precoding scheme for time-division duplex coordinated multiple point transmission (TDD-CoMP) can have a good performance with no feedback by using reciprocity between uplink and downlink. However, if channel is time-varying, the channel difference would cause codeword mismatch between transmitter and receiver, which leads to performance degradation. In this paper, a linear interpolation method is proposed for TDD-CoMP system to estimate the uplink channel at the receiver, which would reduce the channel difference caused by time delay and decrease the probability of codeword mismatch between both sides. Moreover, to mitigate severe inter-cell interference and increase the coverage and throughput of cell-edge users in UDN, a two-codebook scheme is used to strengthen cooperation between base stations (BSs), which can outperform the global precoding scheme with less overhead. Simulations show that the proposed scheme can significantly improve the link performance compared to the global precoding scheme.

Keywords

ultra-dense networks; coordinated multiple point transmission; time-division duplex; two codebooks; cooperative precoding

1 Introduction

In order to meet the significant traffic demands in 5G system, ultra-dense networks (UDN) have been proposed as a promising approach by getting access nodes as close as possible to user equipment (UE) [1]. In UDN, coordinated multi-point (CoMP) operation should be especially considered for it improves coverage and increases cell-edge throughput [2]. However, CoMP requires significant feedback overhead. In fact, overhead could be decreased in time-division duplex (TDD) system by making use of reciprocity between uplink and downlink. Therefore, channel reciprocity will play an important role in TDD-CoMP.

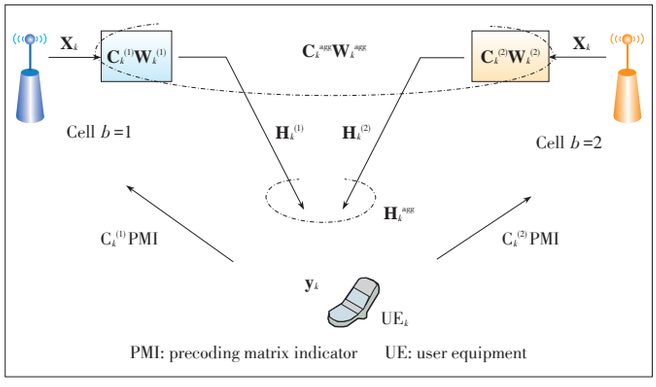
However, little difference between uplink and downlink could result in mismatch between precoding matrix and decoding matrix, which would seriously degrade the system performance [3]–[4]. So, in [5]–[6], Wiener filter was proposed to predict the channel state information (CSI) in next downlink transmission. In [7], a multiple-input multiple-output orthogonal frequency division multiplexing (MIMO-OFDM) downlink channel prediction technique based on Kalman filter was proposed for IEEE 802.16e systems. Although prediction could reduce performance loss caused by codeword mismatch, it leads to high computational complexity for transmitter and its benefit is limited. In this paper, a linear interpolation method is proposed for TDD-CoMP system to estimate the uplink channel at the receiver, which would reduce the channel difference caused by time delay and decrease the probability of codeword mismatch between both sides.

Therefore, by using reciprocity between uplink and downlink, local precoding scheme could have a good performance with no feedback [8]. However the performance of local precoding scheme was inferior to global precoding scheme owing to lack of cooperation between base stations (BSs). Hence, in this paper two codebooks known at both sides are constructed for strengthening cooperation between BSs.

Motivated by this, a precoding scheme based on two codebooks and linear interpolation is proposed for TDD-CoMP systems, where the transmitter and receiver can choose optimal codeword from codebooks according to the estimated channel by using linear interpolation.

2 System Model

As illustrated in **Fig. 1**, suppose the total number of CoMP cells is B. The baseband channel matrix between CoMP cell b with N_T antennas ($b=1,2,\cdots,B$) and UE_k (with N_R receive antennas) is denoted as $\mathbf{H}_k^{(b)}$ ($N_R \times N_T$). Let $\mathbf{W}_k^{(b)}$ be



▲Figure 1. The precoding scheme ($B=2$ CoMP cells).

the precoding matrix of cell b with size $N_T \times N_S$, where N_S is the number of transmission layers for UE_k. Let $\mathbf{C}_k^{(b)}$ be the synchronization codeword matrix with size $N_T \times N_T$. For simplicity, let $B=2$. The received symbols can be expressed as

$$y_k = \sqrt{P}\,\mathbf{H}_k^{agg}\mathbf{C}_k^{agg}\mathbf{W}_k^{agg}\mathbf{x}_k + \mathbf{n}_k, \quad (1)$$

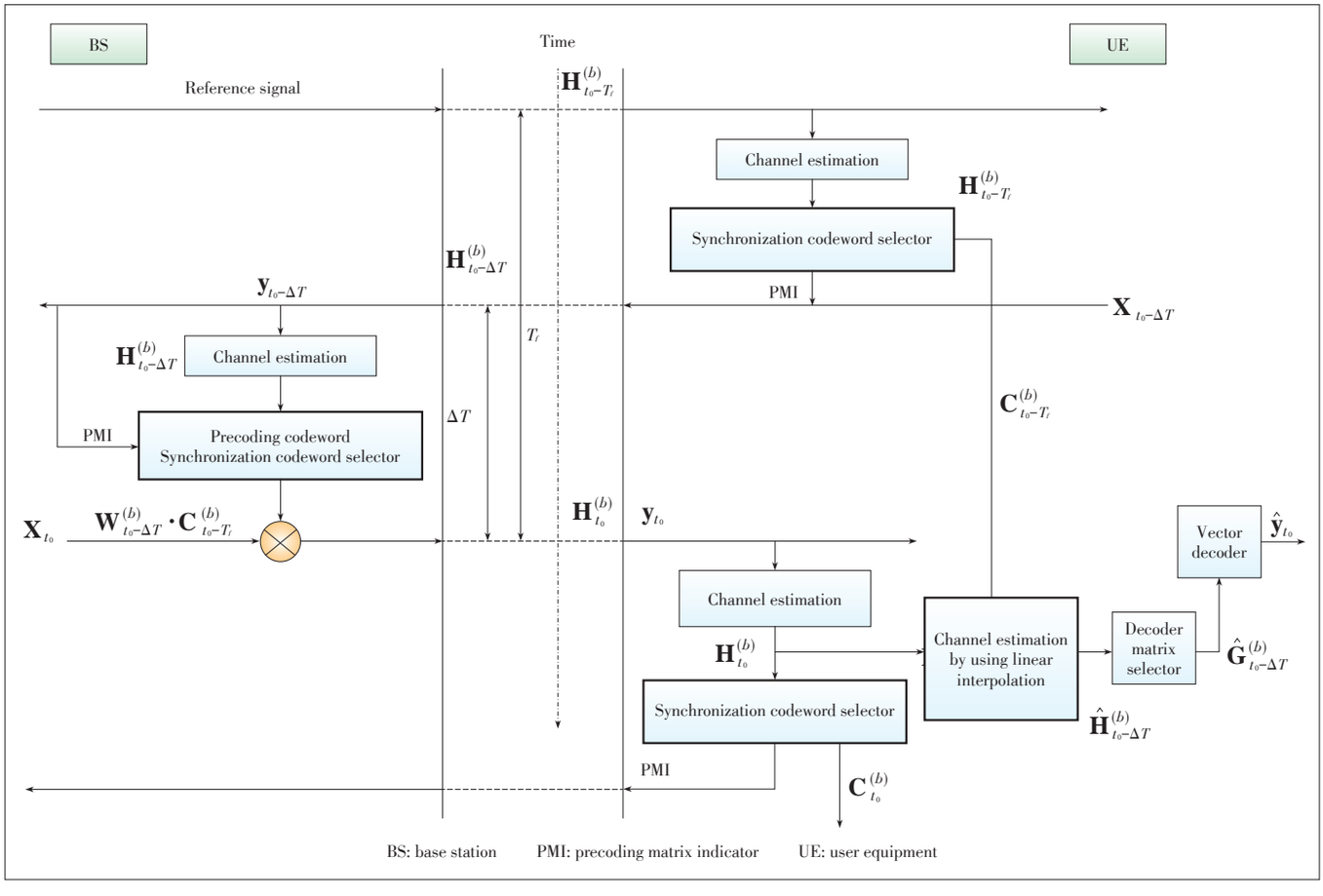
where $\mathbf{x}_k$ ($N_S \times 1$) is the transmission data with N_S layers, $\sqrt{P}$ is the total power on each layer from B CoMP cells, and $\mathbf{n}_k$ is the additive white Gaussian noise (AWGN) vector with covariance matrix $E\left[\mathbf{n}_k\mathbf{n}_k^H\right] = N_0\mathbf{I}_{N_R}$ with the operator $()^H$ representing a matrix conjugate transpose and $\mathbf{I}_{N_R}$ being the identity matrix of order N_R. For notational convenience, denote the aggregated channel matrix as $\mathbf{H}_k^{agg} = \left[\mathbf{H}_k^{(1)}\ \mathbf{H}_k^{(2)}\ \ldots\ \mathbf{H}_k^{(B)}\right]$ ($N_R \times BN_T$) and $\mathbf{W}_k^{agg} = \begin{bmatrix} \mathbf{W}_k^{(1)} \\ \mathbf{W}_k^{(2)} \\ \vdots \\ \mathbf{W}_k^{(B)} \end{bmatrix}$, $\mathbf{C}_k^{agg} = \begin{bmatrix} \mathbf{C}_k^{(1)} & 0 & 0 & 0 \\ 0 & \mathbf{C}_k^{(2)} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{C}_k^{(B)} \end{bmatrix}$.

3 Proposed Scheme

As illustrated in **Fig. 2**, the proposed scheme is composed of two key modules (that is, channel estimation by using linear interpolation and codeword selection based on codebook), where the design of codebooks is presented in Subsection 3.2.3.

3.1 Channel Estimation Based on Linear Interpolation

In traditional TDD precoding system, UE can obtain decod-



▲Figure 2. The proposed precoding scheme in TDD system.

ing matrix $\hat{\mathbf{G}}_{t_0}^{(b)}$ according to channel estimation $\mathbf{H}_{t_0}^{(b)}$, then the transmitted data vector is determined by using soft decision:

$$\mathbf{y}_{t_0}^{'} = \mathrm{deci}\left(\hat{\mathbf{G}}_{t_0}^{(b)}\mathbf{y}_{t_0}\right), \tag{2}$$

where

$$\mathbf{y}_{t_0} = \mathbf{H}_{t_0}^{(b)}\mathbf{C}_{t_0-T_f}^{(b)}\mathbf{W}_{t_0-\Delta T}^{(b)}\mathbf{x}_{t_0} + \mathbf{n}_{t_0}, \tag{3}$$

where t_0 denotes time index of system, ΔT represents the time delay between uplink and downlink transmission, and T_f is the frame duration.

The interpolation based channel estimation method aims to estimate the uplink channel in last frame. When the CSI at t_0 is available at UE, the estimated uplink channel $\hat{\mathbf{H}}_{t_0-\Delta T}^{(b)}$ could be given by

$$\hat{\mathbf{H}}_{t_0-\Delta T}^{(b)} = \mathbf{H}_{t_0}^{(b)} - \Delta T * \Delta\mathbf{H}^{(b)},\ \Delta\mathbf{H}^{(b)} = \frac{\mathbf{H}_{t_0}^{(b)} - \mathbf{H}_{t_0-T_f}^{(b)}}{T_f}. \tag{4}$$

Thus $\hat{\mathbf{G}}_{t_0-\Delta T}^{(b)}$ can be derived from $\hat{\mathbf{H}}_{t_0-\Delta T}^{(b)}$, (2) turns out to be:

$$\hat{\mathbf{y}}_{t_0} = \mathrm{deci}\left(\hat{\mathbf{G}}_{t_0-\Delta T}^{(b)}\mathbf{y}_{t_0}\right). \tag{5}$$

Considering the complexity of maximum likelihood (ML) receiver, the proposed scheme adopts a sub-optimal linear receiver.

In (1), receiver obtains estimated value of $\mathbf{y}_k$ by using $N_R \times N_T$ matrix $\mathbf{G}_k^{(b)}$:

$$\hat{\mathbf{y}}_k = \mathrm{deci}\left(\mathbf{G}_k^{(b)}\mathbf{y}_k\right). \tag{6}$$

When zero-forcing (ZF) receiver is used in system:

$$\mathbf{G}_k^{(b)} = \left(\mathbf{H}_k^{(b)}\mathbf{W}_k^{(b)}\right)^{-1}. \tag{7}$$

3.2 Codeword Selection Based on Codebook

The scheme of codeword selection based on codebook is composed of two parts.

3.2.1 Precoding Codeword Selection

UE_k can know the channel matrix $\mathbf{H}_k^{(b)}$ according to downlink reference signals, then the optimal precoding codeword which maximums the capacity of equivalent channel is chosen from the first codebook. The criterion can be expressed as

$$\mathbf{W}_k^{(b)} = \arg\max_{\mathbf{W}_{i,k}^{(b)} \in \mathbf{W}} \left(C\left(\mathbf{W}_{i,k}^{(b)}\right)\right), \tag{8}$$

$$C\left(\mathbf{W}_{i,k}^{(b)}\right) = \log_2\left(\det\left[\mathbf{I}_{N_s} + \frac{P}{N_S N_0}\mathbf{W}_{i,k}^{(b)H}\mathbf{H}_k^{(b)H}\mathbf{H}_k^{(b)}\mathbf{W}_{i,k}^{(b)}\right]\right), \tag{9}$$

where $b = 1,2$ denotes the coordinated BSs.

3.2.2 Synchronization Codeword Selection

With the aggregated matrix, the optimal synchronization codeword which maximums the capacity of equivalent channel can be chosen from the second codebook. The criterion can be expressed as

$$\mathbf{C}_k^{agg} = \arg\max_{\mathbf{C}_{i,k}^{agg} \in \mathbf{C}} \left(\mathbf{C}\left(\mathbf{C}_{i,k}^{agg}\right)\right), \tag{10}$$

$$\mathbf{C}\left(\mathbf{C}_{i,k}^{agg}\right) = \log_2\left(\det\left[\mathbf{I}_{N_s} + \frac{P}{N_S N_0}(\mathbf{C}_{i,k}^{agg}\cdot\mathbf{W}_k^{agg})^H\mathbf{H}_k^{aggH}\mathbf{H}_k^{agg}(\mathbf{C}_{i,k}^{agg}\cdot\mathbf{W}_k^{agg})\right]\right). \tag{11}$$

3.2.3 Codebook Design

Two codebooks are used in our scheme. The first codebook design can be found in TS36.211, and the second codebook is constructed as follows.

For transmission on 2-Tx antennas in TS36.211, the precoding matrix $\mathbf{W}$ can be denoted as

$$\mathbf{W} = \begin{bmatrix} a \\ b \end{bmatrix}. \tag{12}$$

With partial CSI at the transmitter (CSIT), the precoding matrix $\mathbf{W}$ tries to approximately match its eigen-beams to the channel eigen-directions (the eigenvectors of $\mathbf{H}^H\mathbf{H}$) and therefore reduces the interference among signals sent on these beams [3]. Now $\mathbf{W}$ is used to match the eigen-beams from B BSs in the aggregated channel eigen-directions and reduce the interference among signals sent from B BSs. For transmission on $B = 2$ BSs, the new codeword $\mathbf{C}_k^{(b)}$ based on $\mathbf{W}$ can be described by

$$\mathbf{C}_k^{(1)} = \begin{pmatrix} a & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & a \end{pmatrix},\quad \mathbf{C}_k^{(2)} = \begin{pmatrix} b & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & b \end{pmatrix}. \tag{13}$$

The synchronization codeword $C_k^{(1)}$, $C_k^{(2)}$ ($N_T \times N_T$) should be normalized as

$$\mathbf{C}_k^{(1)} = \mathbf{C}_k^{(1)}/\mathrm{norm}(\mathbf{C}_k^{(1)}),\ \mathbf{C}_k^{(2)} = \mathbf{C}_k^{(2)}/\mathrm{norm}(\mathbf{C}_k^{(2)}), \tag{14}$$

and the aggregated synchronization codeword matrix is defined by

$$\mathbf{C}_k^{agg} = \begin{bmatrix} \mathbf{C}_k^{(1)} & 0 \\ 0 & \mathbf{C}_k^{(2)} \end{bmatrix}. \tag{15}$$

Also, note that the second codebook size is equal to the 2-Tx

codebook size in TS36.211.

4 Simulation Results

To show the superiority of the proposed scheme, two sets of bit error ratio (BER) lower bounds are evaluated by the Monte Carlo method. An example of TDD frame structure in **Fig. 3** would give a clear illustration.

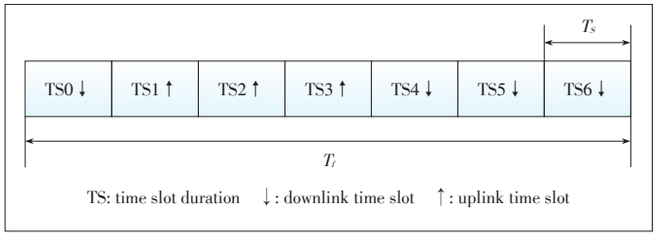
The CSI estimation delay is modeled by setting $\Delta T = 3.5T_s$. Assume that the mean square error (MSE) of time-varying channel is α^2, then the channel of different time can be modeled as

$$\mathbf{H}^{(l)}_{t_0-\Delta T} = \alpha \mathbf{H}^{(l)}_{t_0-T_f} + \sqrt{(1-\alpha^2)}\,\mathbf{I}, \tag{16}$$

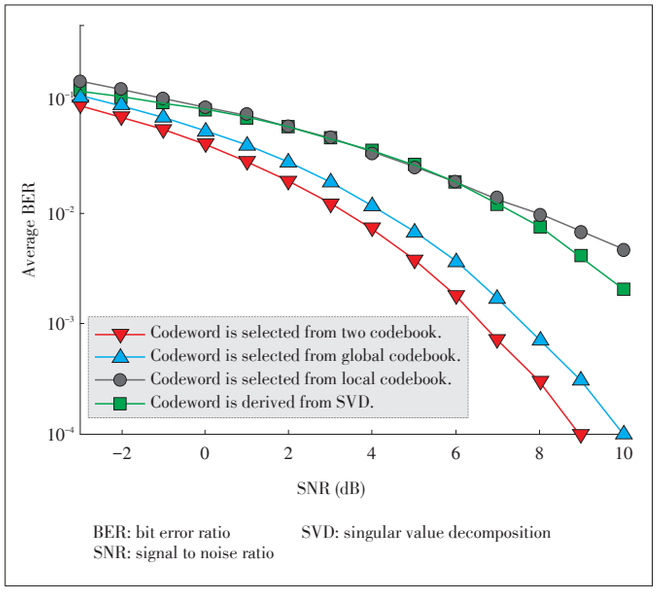
$$\mathbf{H}^{(l)}_{t_0} = \alpha \mathbf{H}^{(l)}_{t_0-\Delta T} + \sqrt{(1-\alpha^2)}\,\mathbf{I}. \tag{17}$$

Fig. 4 presents the BER performance versus transmitted signal to noise ratio (SNR) for ideal TDD system. The results show that our proposed scheme outperforms the global codebook scheme with less overhead because the proposed scheme makes full use of reciprocity between uplink and downlink in the ideal TDD system. It is also shown that the singular value decomposition (SVD) scheme outperforms the local codebook scheme without channel estimation by using linear interpolation.

Fig. 5 presents the BER performance versus transmitted SNR for the practical TDD system. As shown in the figure, the proposed technique is superior to the local codebook scheme owing to synchronization codeword overhead. Besides, the proposed technique is always superior to the two codebooks scheme, which is due to the fact that the channel estimation by using linear interpolation reduces the probability of mismatch between precoding matrix and decoding matrix. Fig. 5 also shows that the precoding scheme based on codebook is superior to the SVD scheme in practical system.



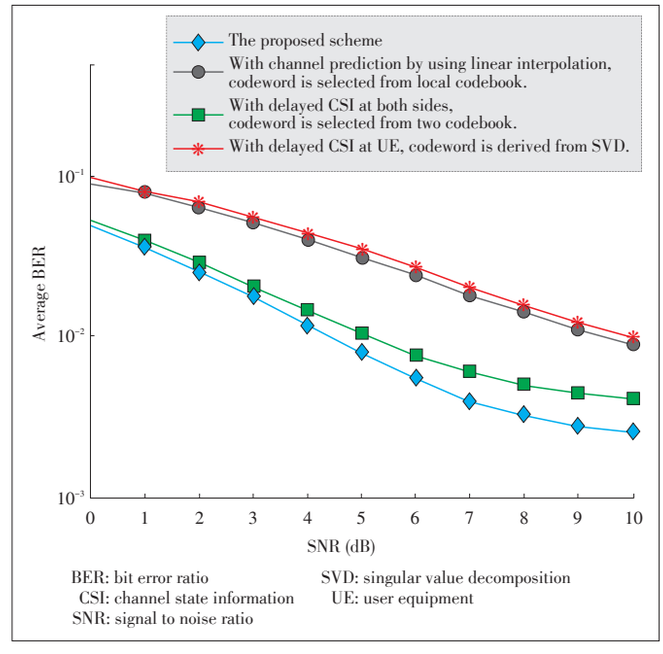
▲Figure 3. Example of TDD frame structure.



▲Figure 4. BER performance comparison in ideal TDD system.

5 Conclusions

In UDN, a novel precoding technique for TDD-CoMP system is proposed, which aims at reducing the probability of codeword mismatch and improving the cell-edge throughput. To reduce the CSI difference between both sides caused by time delay, a linear interpolation method is utilized at UE to estimate the CSI achieved at BS. Furthermore, the proposed scheme selects the codeword from two codebooks defined previously at both sides, in order to benefit from cooperation between BSs. As illustrated in simulation results, the proposed scheme could improve link performance of TDD-CoMP system and reduce feedback remarkably, compared to the global codebook scheme.



▲Figure 5. BER performance comparison in practical TDD system.

References

[1] M. Kamel, W. Hamouda, and A. Youssef, "Ultra-dense networks: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2522–2545, Fourthquarter 2016. doi: 10.1109/COMST.2016.2571730.

[2] L. Liu, V. Garcia, L. Tian, Z. Pan, and J. Shi, "Joint clustering and inter-cell resource allocation for CoMP in ultra dense cellular networks," in *IEEE International Conference on Communications (ICC)*, London, UK, Jun. 2015, pp. 2560–2564. doi: 10.1109/ICC.2015.7248710.

[3] M. Vu and A. Paulraj, "MIMO wireless linear precoding," *IEEE Signal Processing Magazine*, vol. 24, no. 5, pp. 86–105, Sept. 2007. doi: 10.1109/MSP.2007.904811.

[4] C. Gao, M. Enescu, and X. G. Che, "On the feasibility of SVD-based downlink precoding in future TDD systems," in *IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Athens, Creece, September 2007, pp. 1-5. doi: 10.1109/PIMRC.2007.4394635.

[5] G. Lebrun, S. Spiteri, and M. Faulkner, "Channel estimation for an SVD-MIMO system," in *IEEE International Conference on Communications (ICC)*, Paris, France, Jun. 2014, pp. 3025–3029. doi: 10.1109/ICC.2004.1313087.

[6] Y. Tan, G. Lebrun, and M. Faulkner, "An adaptive channel SVD tracking strategy in time-varying TDD system," in *IEEE Semiannual Vehicular Technology Conference (VTC)*, Jeju, South Korea, Apr. 2003, pp. 769–773. doi: 10.1109/VETECS.2003.1207648.

[7] C. Min, N. Chang, J. Cha, and J. Kang, "MIMO-OFDM downlink channel prediction for IEEE802.16e systems using kalman filter," in *IEEE Wireless Communications and Networking Conference (WCNC)*, Kowloon, China, Mar. 2007, pp. 942–946. doi: 10.1109/WCNC.2007.179.

[8] Alcatel-Lucent, "Downlink non-coherent SU-CoMP schemes comparison for TDD systems," Alcatel-Lucent Shanghai Bell, Alcatel-Lucent, R1-092159, May 2009.

Manuscript received: 2018-03-05

Biographies

GAO Tengshuang (gaotengshuang@szhrss.gov.cn) received his M.S. degree in software engineering from Tianjin University, China in 2016. He is currently pursuing for Ph.D. degree in software engineering at Tianjin University. He has almost 12 years of experience in software engineering field, especially in interdisciplinary fields such as computing science applying in wireless communications.

CHEN Ying (destinyshore@bupt.edu.cn) received the bachelor's degree in communication engineering from Beijing University of Posts and Telecommunications (BUPT), China in 2016. She is currently working toward the M. Tech. degree in communication and information engineering at the School of Information and Communication Engineering, BUPT. Her research interests include the emerging technologies of 5G wireless communication network.

HAO Peng (hao.peng@zte.com.cn) received his M.S. degree in communication engineering from Beijing University of Posts and Telecommunications, China. He joined ZTE Corporation in 2006 and worked on system and link level simulation of 4G system. He has also been involved in the research of physical layer key technologies of LTE and LTE-A system. He authored 70 3GPP proposals and more than 100 Chinese and international patents. His research interests include MIMO technologies and the design of interference mitigation algorithms. He is currently serving as a senior engineer of ZTE and in charge of the study of 5G ultra dense network.

ZHANG Hongtao (htzhang@bupt.edu.cn) received the Ph.D. degree in communication and information systems from Beijing University of Posts and Telecommunications (BUPT), China in 2008. He is currently an associate professor with the School of Information and Communications Engineering, BUPT. He has published more than 60 articles on international journals and conferences, and has filed more than 30 patents. He is the author of seven technical books. His research interests include 5G wireless communication and signal processing.

DOI: 10.3969/j.issn.1673-5188.2018.02.007
http://kns.cnki.net/kcms/detail/34.1294.TN.20180427.1001.004.html, published online April 27, 2018

Markov Based Rate Adaption Approach for Live Streaming over HTTP/2

XIE Lan[1], ZHANG Xinggong[1], HUANG Cheng[2], and DONG Zhenjiang[2]
(1. Institute of Computer Science and Technology, Peking University, Beijing 100080, China;
2. ZTE Corporation, Nanjing 210012, China)

Abstract

Dynamic adaptive streaming over HTTP (DASH) has been widely deployed. However, large latency in HTTP/1.1 cannot meet the requirements of live streaming. Data-pushing in HTTP/2 is emerging as a promising technology. For video live over HTTP/2, new challenges arise due to both low-delay and small buffer constraints. In this paper, we study the rate adaption problem over HTTP/2 with the aim to improve the quality of experience (QoE) of live streaming. To track the dynamic characteristics of the streaming system, a Markov-theoretical approach is employed. System variables are taken into account to describe the system state, by which the system transition probability is derived. Moreover, we design a dynamic reward function considering both the quality of user experience and dynamic system variables. Therefore, the rate adaption problem is formulated into a Markov decision based optimization problem and the best streaming policy is obtained. At last, the effectiveness of our proposed rate adaption scheme is demonstrated by numerous experiment results.

Keywords

DASH; live; rate adaption; Markov decision

1 Introduction

In recent years, dynamic adaptive streaming over HTTP (DASH) has been widely adopted for providing uninterrupted video streaming service to users with dynamic network conditions and heterogeneous devices [1], [2]. Contrary to the past Real-Time Transport Protocol/User Datagram Protocol (RTP/UDP), the use of HTTP over Transmission Control Protocol (TCP) greatly simplifies the traversal of firewalls and network address translators (NAT) which can be easily deployed within content delivery networks (CDN). Moreover, the rate adaption scheme is one of the most essential components to improve the streaming quality. By far, many rate adaptation schemes have been designed for DASH, including bandwidth-based rate adaption schemes and buffer-based rate adaption schemes [3], [4]. Akhshabi et al. [1] compared rate adaption for three popular DASH clients: Netflix client, Microsoft Smooth Streaming [5], and Adobe Open Source Meda Framework (OSMF). The conclusion in [1] indicates that none of the rate adaptation is good enough.

On the other hand, to track the dynamic characteristics of streaming system, Markov theory has been shown to be effective [6]–[8]. Regarding the work by García's et al. [6], they use Stochastic Dynamic Programming (SDP) optimization to solve the rate adaption problem for DASH. In which, the cost function is designed to stable the buffer occupancy at a certain level, leading to frequent video bitrate fluctuations. In [9], rate selection is performed offline by a Markov Decision Process (MDP) assuming that the available bandwidth can be estimated using a transition matrix. Applying the model online, however, may result in inaccurate estimation due to unpredictable characteristics of network conditions. This work is further extended in [7] and [10].

However, most of existing works focus on Video on Demand (VoD) service, and they are not suitable for live streaming where low latency is required. Moreover, by adopting HTTP/2, a low end-to-end latency can be ensured because multiple video fragments can be pushed to clients by a single request [11], [12]. This has been demonstrated by Wei et al. [13] based on their multiple experiments on video streaming over HTTP/2. As for the rate adaptation in live streaming, some new challenges arise; for example, the low start-up delay is required, and the bandwidth variation cannot be smoothed by setting a large buffer.

In this paper, we study the rate adaptation problem for live streaming over HTTP/2. Similar to [6], the Markov theory is applied to analyze the dynamic characteristics of the system and the rate adaptation problem is formulated into an optimization problem. To track the dynamic characteristics of the streaming system, several system variables are used to describe the system state, including video rate, buffer occupancy, available bandwidth, playback deadline and download time for each segment, and then the system transition probability is derived. Moreover, a dynamic reward function is designed under three scenarios of buffer occupancy to meet the requirement of user experience. The experiments done by bandwidth trace have shown that our proposed algorithm can provide a smooth and high video rate while guaranteeing a continuous video playback.

The rest of the paper is organized as follows. Section 2 presents the overview of Markov Decision based rate adaptation. The system state is introduced and the state transition probabil-

This work was supported in part by China "973" Program under Grant No. 2014CB340303" and ZTE Industry-Academia-Research Cooperation Funds.

ity is derived in Section 3. In Section 4, the dynamic reward function is described. At last, we show experiment results in Section 5, and conclude the paper in Section 6.

2 Markov Based Rate Adaptation

In this paper, we propose a Markov based rate adaption approach. First, we define system state at stage k as u_k, which has taken into account several system variables. An action a_k is defined at stage k, denoting a specific bitrate that is assigned for segment $k+1$. After taking action a_k, the system state transfers from u_k to u_{k+1}, i.e.,

$$u_k = f(u_k, a_k). \tag{1}$$

Due to the stochastic nature of the system, it can be characterized in terms of conditional probability distribution among states, that is, $P(u_{k+1}|u_k, a_k)$ which is the transition probability from u_k to u_{k+1} under action a_k.

On the other hand, in order to evaluate how good an action is, a reward function is also designed and the reward of action a_k or state u_k is defined as $R(u_k, a_k)$. Then, the long-term reward can be written as:

$$V(u_k, a_k) = \sum_{\{u_{k+1}\}} P(u_{k+1}|u_k, a_k) \cdot (R(u_k, a_k) + \gamma V(u_{k+1}, a_{k+1})), \tag{2}$$

where $V(u_k, a_k)$ calculates the sum of the rewards of all the possible next state u_{k+1} and $0 < \lambda < 1$ is a future discount rate that controls how much effect future rewards have on the decision at the current stage.

The streaming policy π is a mapping between system state u_k and action a_k. Obviously, finding the optimal strategy policy $\pi^*(u_k)$ which can maximize (2) is the goal of MDP. Therefore, our rate adaptation task can be finally formulated as an optimization problem:

$$\pi^*(u_k) = \arg\max_{a_k} V(u_k, a_k). \tag{3}$$

The detailed definition and analysis of the MDP is formulated mathematically in the following section.

3 State Transition Probability

In this section, we introduce the system state in detail and its transition probability. At stage k, the k-th segment is pushed to the client. Different from VoD service, in a live streaming scenario, media segments are available only after they have been generated. To reveal this feature, we have considered the playback deadline and the arrival time of each segment, and the system state is defined as:

$$u_k = \{q_k, v_k, \hat{t}_k, t_k, bw_k\}, \tag{4}$$

with each parameter stands for buffer occupancy, video bitrate, playback deadline, actual arrival time, and available bandwidth respectively. Given state u_k, an a_k action is taken to select the video rate for the next segment, i.e., $v_{k+1} = a_k(u_k)$. Note, if the segment is not available at the server side, a wait action will be taken.

The buffer occupancy q_k denotes the buffer level when segment k is just completely downloaded, and it is measured in second. It increases when a segment is pushed from the server and descends when segments are consumed by playing. Therefore, the buffer occupancy evolution can be written as:

$$q_{k+1} = \max\left\{q_k + T_s - \frac{v_{k+1} \cdot T_s}{bw_{k+1}}, 0\right\}, \tag{5}$$

where T_s is the duration of one segment.

For parameters $\hat{t}_k$ and t_k, they are used to characterize the time attributes of each segment. Moreover, we define T_d as the start-up delay, the playback deadline of segment $k+1$ can be written as:

$$\hat{t}_{k+1} = (k+1) \cdot T_s + T_d. \tag{6}$$

On the other hand, in live streaming, a segment is pushed to the client as soon as the previous segment has been totally sent out. Therefore, the actual arrival time of segment k is determined by both the previous segment arrival time t_k and the transmission duration of segment k:

$$t_{k+1} = t_k + \frac{v_{k+1} \cdot T_s}{bw_{k+1}}. \tag{7}$$

Generally, it is difficult to estimate the statistic of the bandwidth accurately. However, it has been widely known that the Markov channel models are useful tools to describe the variations of bandwidth. Thus, we also apply the Markov model to describe the available bandwidth. According to the Markov property, the state at any time instance only depends on its previous state. Considering for taking an action under a certain state, all the factors in the next system state can be calculated using (5)–(7) except bandwidth itself. Therefore, given a previous state u_k and an action a_k, the transition probability of the MDP is related to the transition probability of bandwidth which can be given as:

$$P(u_{k+1}|u_k, a_k) = P(bw_{k+1}|bw_k). \tag{8}$$

4 Dynamic Reward Function

In this section, we propose a reward function to measure how good an action is. Previous works [6]–[8] pointed out several factors that have impacts on user experience. In this paper, four system factors have been considered, including video rate and its smoothness, buffer occupancy, and playback deadline constraint. The impact of these factors on user experience is denoted by R_1, R_2, R_3, R_4 respectively. Then, the reward for an action can be evaluated by

$$R(u_k, a_k) = aR_1 + bR_2 + cR_3 + dR_4, \tag{9}$$

where a, b, c, d are weights for each factors.

In the following, we will derive the four reward functions included in $R(u_k,a_k)$ under three scenarios, since the factors a user concerns are generally different under different streaming scenarios.

4.1 Scenario 1: $q_k < T_s$

When the buffer occupancy is lower than T_s, i.e., the segment duration, the risk of buffer underflow is high. In this case, a low bitrate is preferred so as to avoid playback freeze. From the video quality perspective, the reward value gained from the video rate obeys the logarithmic relationship and R_1 is defined as:

$$R_1 = \ln(v_k - B_{\min} + \varepsilon), \tag{10}$$

where $B_{\min}$ is the lowest available bitrate and ε is an arbitrary small positive number. On the other hand, rate switch among continuous segments will bring negative effect on user visual perception. Therefore, the smoothness of video rate on user experience R_2 can be presented as:

$$R_2 = \ln(|v_k - v_{k-1}| + \varepsilon). \tag{11}$$

At last, when $q_k < T_s$, more attention should be paid to avoid buffer underflow. One of the most effective method is to select the lowest bitrate. Therefore, we can simply define R_3 and R_4 as:

$$R_3 = -\infty, \tag{12}$$

$$R_4 = -\infty. \tag{13}$$

4.2 Scenario 2: $q_k > q_{high}$

We predefined a threshold q_{high} used to avoid buffer overflow. When the buffer occupancy is larger than q_{high}, the risk of buffer overflow is high and the risk of buffer underflow is low. In this case, a high rate is preferred. For the video rate and its smoothness on user experience, they are defined the same as Scenario 1. For buffer occupancy, we prefer to select a video rate that can drag the buffer occupancy to be no higher than q_{high}, and R_3 is written as:

$$R_3 = q_{high} - q_k. \tag{14}$$

At last, R_4 measures the difference between the download time and playback time. When $\hat{t}_k > t_k$, some video segments has been buffered. On the other hand, if $\hat{t}_k < t_k$ the buffer is empty and playback freeze happens. Therefore, we can define R_4 as:

$$R_4 = \begin{cases} \hat{t}_k - t_k, \hat{t}_k - t_k \geqslant T_s \\ -\infty, \hat{t}_k - t_k < T_s \end{cases}. \tag{15}$$

4.3 Scenario 3: $T_s \leqslant q_k \leqslant q_{high}$

When buffer occupancy satisfies $T_s \leqslant q_k \leqslant q_{high}$, i.e., the buffered video time is between one segment duration and the overflow threshold, the risk of both buffer overflow and underflow is low. The video rate and its smoothness on user experience are defined the same as in Scenarios 1 and 2. For buffer occupancy, we prefer to select a video rate that can keep the buffer occupancy stable in range of $[T_s, q_{high}]$ and R_3 is written as:

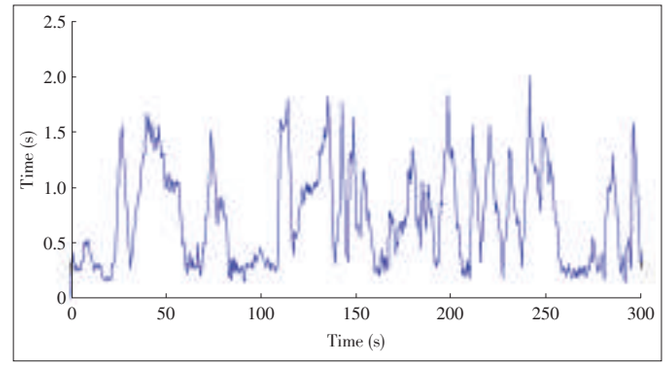
$$R_3 = \frac{T_s + q_{high}}{2}. \tag{16}$$

R_4 here is simply computed as the same as that in Scenario 2.

5 Experiments

In our experiment, same with Netflix, the server provides five different versions of video bitrates {300 kbit/s, 700 kbit/s, 1.5 Mbit/s, 2.5 Mbit/s, 3.5 Mbit/s}. Each version is an equal-length segment, with the length of 0.3 s. For the start-up delay, we set $T_d = 2$ s which is the length of seven segments. This setting can not only ensure a low startup delay, but also provide a continuous video playback as will be shown in the experiment results. For the buffer overflow threshold, we set $q_{high} = 1.8$ s, i.e., the length of six segments, which can be used to avoid that the buffered video time is higher than initial buffered video data so as to switch to a higher video rate. For performance comparison, we extend the previous SDP-based rate adaption scheme proposed in [6] to deliver live content using a regular request-response mode for requesting the video segment one by one. We also implement the bandwidth based K-Push scheme [13] where the client issues a request every K segments with the same rate. We evaluate the three rate adaptation schemes on real-world bandwidth trace. For our proposed method, the state is updated according to environment change and the best action has been chosen for deciding the video bitrate.

We first evaluate the difference between the playback deadline and download time of each segment in our scheme. The result is shown in **Fig. 1**. From the result we can find that when $T_d = 2$ s, the time difference for all segment is higher than zero, i.e. no playback freeze happens. On the other hand, whenever the time difference is high (approach to 1.8 s), it will quickly



▲**Figure 1. Time difference between the playback deadline and download time.**

decrease. This is because high time difference means the segment needs to wait for a long time before being played. Therefore, a high video rate can be selected.

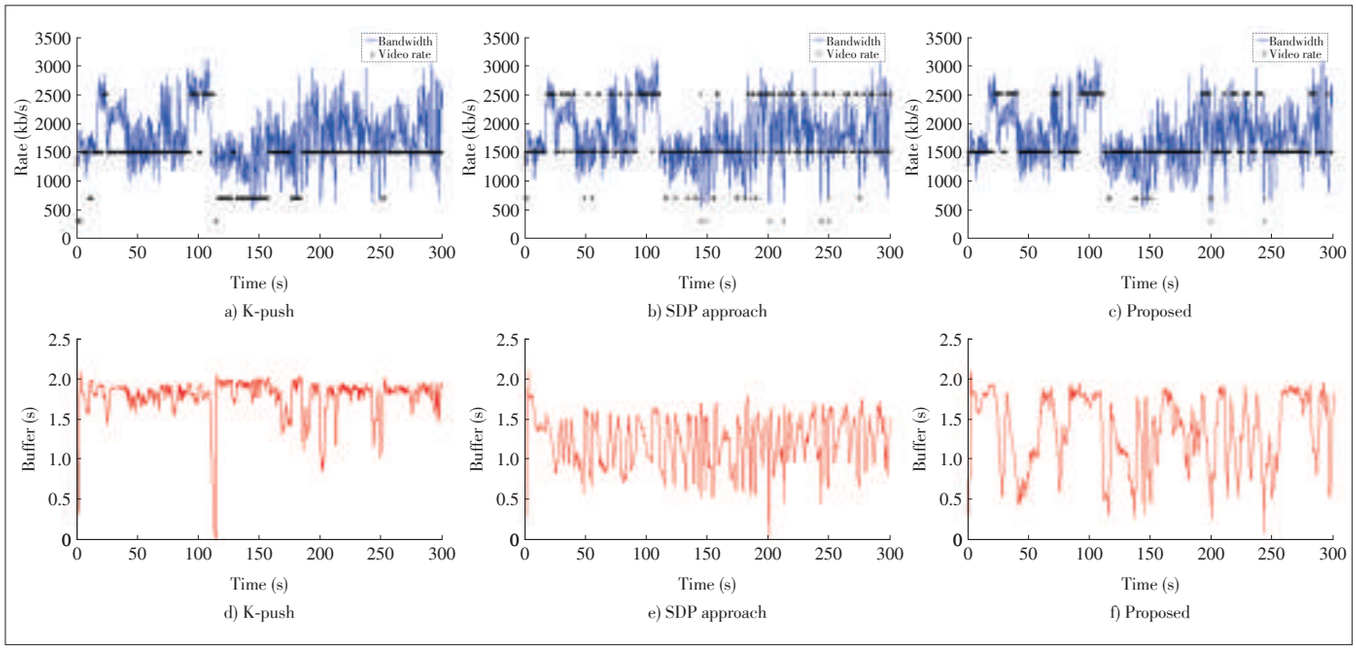
At last, we compared the performance of all the three schemes. The results (**Fig. 2**) demonstrate that the video rate follows the principle of designing reward function well and that the rate decreases when buffer occupancy is low thus preventing playback freeze. Moreover, smoothness is focused if buffered video is adequate to keep continuous video playback. Compared with the SDP approach, our proposed scheme obtains a much smoother video rate. This is because in the SDP approach, the reward function is designed to stabilize the buffer occupancy without considering the smoothness of video rate well. From Fig. 2b we can find that the video rate is mainly switched between 1.5 Mbit/s and 2.5 Mbit/s in the SDP approach, and there are also a few segments whose rate are assigned to be 700 kbit/s or 300 kbit/s due to the low buffer occupancy. At last, when compared with the bandwidth based K-Push scheme, we can find that both have close performance on the smoothness of video rate. This is mainly because in the bandwidth based K-Push scheme, consecutive K segments are assigned the same video rate and bandwidth variations can be smoothed well. However, since video rate is switched every K segments, the bandwidth based K-Push scheme is insensitive to bandwidth variation and playback freeze happens as Fig. 2d shows.

5 Conclusions

In this paper, we have studied the rate adaption problem for live streaming over HTTP/2 by the Markov theory. To track the dynamic characteristics of the streaming system, we have defined a system state and several system variables are taken into account, including video rate, buffer occupancy, available bandwidth, playback deadline, and download time for each segment, and then the system transition probability is derived. We also have designed a dynamic reward function considering both the quality of user experience and dynamic system variables. Therefore, the rate adaption problem is formulated into a Markov decision based optimization problem and the best streaming policy is obtained. At last, the experiments by bandwidth trace have demonstrated the high effectiveness of our proposed rate adaption scheme.

References

[1] S. Akhshabi, A. C. Begen, and C. Dovrolis, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over http," in *Proc. 2nd Annual ACM Conference on Multimedia Systems*, San Jose, USA, 2011, pp. 157–168. doi: 10.1145/1943552.1943574.
[2] T. Stockhammer, "Dynamic adaptive streaming over http-: standards and design principles," in *Proc. 2nd Annual ACM Conference on Multimedia Systems*, San Jose, USA, 2011, pp. 133–144. doi: 10.1145/1943552.1943572.
[3] C. Zhou, X. Zhang, L. Huo, and Z. Guo, "A control-theoretic approach to rate adaptation for dynamic http streaming," in *2012 IEEE Visual Communications and Image Processing*, San Diego, USA, 2012, pp. 1–6. doi: 10.1109/VCIP.2012.6410740.
[4] C. Zhou, C.-W. Lin, X. Zhang, and Z. Guo, "A control-theoretic approach to rate adaption for dash over multiple content distribution servers," *IEEE Transactions on Circuts and Systems for Video Technology*, vol. 24, no. 4, pp. 681–694, Apr. 2014. doi: 10.1109/TCSVT.2013.2290580.
[5] A. Zambelli, "IIS smooth streaming technical overview," Microsoft Corporation, USA, vol. 3, 2009.
[6] S. Garcia, J. Cabrera, and N. Garcia, "Quality-control algorithm for adaptive streaming services over wireless channels," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 1, pp. 50–59, Feb. 2015. doi: 10.1109/JST-



▲**Figure 2. Performance comparison. a)–c) are video bitrates of three rate adaption approaches; d)–f) are buffer occupancy of three rate adaption approaches.**

SP.2014.2331912.
[7] M. Xing, S. Xiang, and L. Cai, "A real-time adaptive algorithm for video streaming over multiple wireless access networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 4, pp. 795– 805, Apr. 2014. doi: 10.1109/JSAC.2014.140411.
[8] T. Andelin, V, Chetty, D. Harbaugh, S. Warnick, and D. Zappala, "Quality selection for dynamic adaptive streaming over http with scalable video coding," in *Proc. 3rd Multimedia Systems Conference*, Chapel Hill, USA, 2012, pp. 149–154. doi: 10.1145/2155555.2155580.
[9] S. Xiang, L. Cai, and J. Pan, "Adaptive scalable video streaming in wireless networks," in *Proc. 3rd Multimedia Systems Conference*, Chapel Hill, USA, 2012, pp. 167–172. doi: 10.1145/2155555.2155583.
[10] M. Xing, S. Xiang, and L. Cai, "Rate adaptation strategy for video streaming over multiple wireless access networks," in *Proc. IEEE Global Communications Conference*, Atlanta, USA, 2013, pp. 5745–5750.
[11] C. Mueller, S. Lederer, C. Timmerer, and H. Hellwagner, "Dynamic adaptive streaming over http/2.0," in *IEEE international Conference on Multimedia and Expo (ICME)*, San Jose, USA, 2013, pp. 1– 6. doi: 10.1109/ICME.2013.6607498.
[12] T. C. Thang, H. T. Le, A. T. Pham, and Y. M. Ro, "An evaluation of bitrate adaptation methods for http live streaming," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 4, pp. 693– 705, Apr. 2014. doi: 10.1109/JSAC.2014.140403.
[13] S. Wei and V. Swaminathan, "Low latency live streaming over http2.0," in *Proc. Network and Operating System Support on Digital Audio and Video Workshop*, Singapore, Singapore, 2014, pp. 37–37.

Manuscript received: 2017-01-01

Biographies

XIE Lan (xielan@pku.edu.cn) received the B.S. degree from Beijing University of Technology, China in 2015. She is currently a master degree student at the Institute of Computer Science and Technology, Peking University. Her research interests include dynamic HTTP streaming and VR video streaming. She received Beijing Outstanding Graduate Award in 2015.

ZHANG Xinggong (zhangxg@pku.edu.cn) received the Ph.D. degree from Peking University, China in 2011. He is currently a professor at the Institute of Computer Science and Technology, Peking University. His research interests include video conferencing, dynamic HTTP streaming, information - centric networking and VR video streaming. He received the First Prize of the Ministry of Education Science and Technology Progress Award in 2006 and the Second Prize of the National Science and Technology Award in 2007.

HUANG Cheng (huang.cheng5@zte.com.cn) received the M.S. degree from Southeast University, China in 2006. He is currently a senior system engineer and multimedia standard manager at ZTE Corporation. His research interest include video coding, storage, transport, and multimedia systems.

DONG Zhenjiang (dong.zhenjiang@zte.com.cn) received his M.S. degree in telecommunication from Harbin Institute of Technology, China in 1996. He is the deputy head of the Service Institute of ZTE Corporation and standing director of Chinese Association for Artificial Intelligence. His research interests include cloud computing, mobile internet, natural language processing, and multimedia analysis.

DOI: 10.3969/j.issn.1673-5188.2018.02.008
http://kns.cnki.net/kcms/detail/34.1294.TN.20180427.1001.002.html, published online April 27, 2018

SOPA: Source Routing Based Packet-Level Multi-Path Routing in Data Center Networks

LI Dan[1], LIN Du[1], JIANG Changlin[1], and Wang Lingqiang[2]

(1. Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China;
2. Pre-Research & Standard Department, Wireline R&D Institute, ZTE Corporation, Beijing 100084, China)

Abstract

Many "rich-connected" topologies with multiple parallel paths between servers have been proposed for data center networks recently to provide high bisection bandwidth, but it remains challenging to fully utilize the high network capacity by appropriate multi-path routing algorithms. As flow-level path splitting may lead to traffic imbalance between paths due to flow size difference, packet-level path splitting attracts more attention lately, which spreads packets from flows into multiple available paths and significantly improves link utilizations. However, it may cause packet reordering, confusing the TCP congestion control algorithm and lowering the throughput of flows. In this paper, we design a novel packet-level multi-path routing scheme called SOPA, which leverages OpenFlow to perform packet-level path splitting in a round-robin fashion, and hence significantly mitigates the packet reordering problem and improves the network throughput. Moreover, SOPA leverages the topological feature of data center networks to encode a very small number of switches along the path into the packet header, resulting in very light overhead. Compared with random packet spraying (RPS), Hedera and equal-cost multi-path routing (ECMP), our simulations demonstrate that SOPA achieves 29.87%, 50.41% and 77.74% higher network throughput respectively under permutation workload, and reduces average data transfer completion time by 53.65%, 343.31% and 348.25% respectively under production workload.

Keywords

data center networks; multi-path routing; path splitting

The work was supported by the National Basic Research Program of China (973 program) under Grant No. 2014CB347800 and No. 2012CB315803, the National High-Tech R&D Program of China (863 program) under Grant No. 2013AA013303, the Natural Science Foundation of China under Grant No.61170291, No.61133006, and No.61161140454, and ZTE Industry-Academia-Research Cooperation Funds.

1 Introduction

Data center networks connect hundred of thousands of servers to support cloud computing, including both front-end online services (e.g., web search and gaming) and back-end distributed computations (e.g., distributed file system [1] and distributed data processing engine [2], [3]). Recognizing that the traditional tree-based topology cannot well embrace the bandwidth-hungry cloud services, in recent years many "rich-connected" data center network topologies have been proposed, such as Fat-Tree [4], VL2 [5], BCube [6] and FiConn [7]. These new topologies provide multiple paths between any pair of servers, and greatly increase the network bisection bandwidth. For instance, in a Fat-Tree network, there are x equal paths between two servers from different pods, where x is the number of core switches in the network; while in a $BCube(n,k)$ network, $k+1$ non-disjoint paths exist between any two servers, not to mention the paths with overlapping links.

Although the advanced data center networks enjoy high network capacity, it remains challenging how to fully utilize the capacity and provide high network throughput to upper-layer applications. Multi-path routing is necessary to exploit the abundant paths between servers. The existing multi-path routing schemes can be divided into two categories, namely, flow-level path splitting and packet-level path splitting. In flow-level path splitting solutions, traffic between two servers is split into different paths at the flow granularity. All the packets belonging to a 5-tuple flow traverse the same path, so as to avoid out-of-order delivery. For examples, equal-cost multi-path routing (ECMP) uses 5-tuple hashing to choose the path for a flow from the multiple candidates, with the possibility of hash collision and unequal utilization of the paths; in order to avoid the hash collision between large flows, Hedera [8] explores a centralized way to schedule the flows by spreading large flows into different paths. However, the flow sizes and packet sizes of different flows are usually diversified, which can also lead to traffic imbalance among different paths.

Packet-level path splitting, on the other hand, splits traffic in the packet granularity, i.e., packets from a flow can be put to different paths. Since packets of the same flow are usually of similar sizes, packet-level path splitting achieves desirable traffic balance among multiple candidate paths. However, a major concern of packet-level path splitting is that it may cause packet reordering for TCP flows. Although recent studies showed that the path equivalence in modern data center networks can help mitigate the packet reordering problem, random next-hop selection in random packet spraying (RPS) [9] still results in considerable packet reordering and unsatisfactory flow throughputs, which is worsen when link fails and network symmetry is broken [9]. DRB [10] employs IP-in-IP encapsulation/decapsulation [11] to select the core level switch and uses re-sequencing buffer at the receiver to absorb reor-

dered packets, which not only introduces much traffic overhead, but also causes considerable re-sequencing delay [10].

In this paper we design SOPA, a new packet-level path splitting scheme to carry on multi-path routing in data center networks. SOPA advances the state of art by two technical innovations.

First, rather than introducing an additional buffer at the receiver, SOPA increases the fast retransmit (FR) threshold (i.e., the number of duplicate ACKs received at the sender that acknowledge the same sequence number) used in TCP to trigger FR, so as to mitigate the impact of packet reordering on reducing flow's throughput. In the current TCP congestion control algorithm, packet reordering is regarded as an indicator of packet loss, hence three duplicate ACKs will cause packet retransmit at the sender without waiting for the timeouts. Although it works well in single-path routing, in multi-path routing paradigm it misleads the congestion control algorithm, since in most cases packet reordering does not come from packet loss. By increasing the FR threshold, say, to 10, SOPA significantly reduces the number of unnecessary packet retransmits, which accordingly improves the effective throughput for a flow.

Second, instead of randomly selecting the next-hop switch or using IP-in-IP encapsulation to pick a core level switch, SOPA employs source routing to explicitly identify the path for each packet. Increasing the FR threshold only cannot help improve the flow's throughput any more when the FR threshold exceeds a certain value, because more timeouts and packet retransmissions will occur if there are not enough ACKs. SOPA lets the source server adopt a round-robin approach to select the path for a packet and uses source routing to encode the path into the packet header. In a Fat-Tree network, SOPA leverages the topological characteristic and only needs at most four additional bytes to identify the path. As a result, packet reordering is significantly mitigated by exactly balanced traffic loads among the equivalent paths with negligible traffic overhead. Source routing is also very easy to implement on commodity switching chips [12] or the emerging SDN/OpenFlow paradigm [13], without updating the switch hardware.

The NS-3 based simulation results show that SOPA can achieve high network throughput under different workloads, no matter what the network size is. Under the synthesized permutation workload, the average throughput achieved by SOPA is 29.87%, 50.41% and 77.74% higher than that of RPS, Hedera and ECMP, respectively, in a Fat-Tree network built with 24-port switches. Under the workload from a production data center, compared with RPS, Hedera, and ECMP, SOPA improves the average throughput by 53.65%, 343.33% and 348.34%, respectively, in the same network. SOPA can also gracefully encompass link failures without significant performance degradation.

The rest of this paper is organized as follows. Section 2 introduces the background knowledge and related works. Section 3 describes the design of SOPA. Section 4 presents the evaluation results. Finally Section 5 concludes the paper.

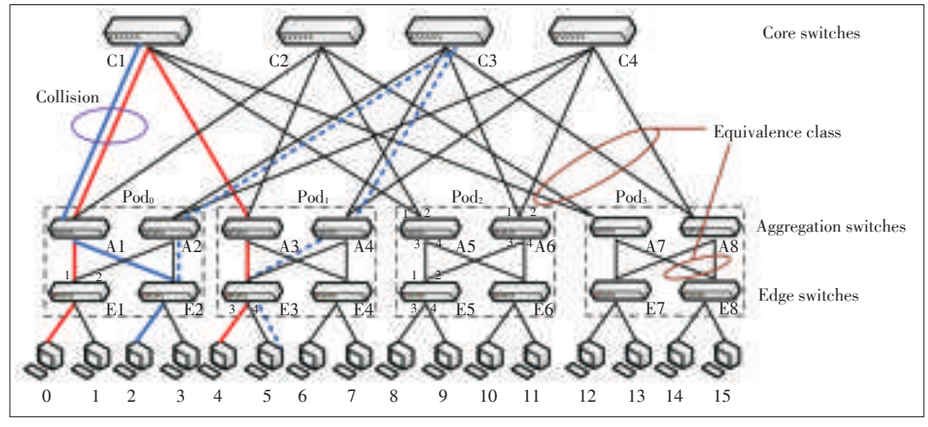
2 Background and Related Work

2.1 Data Center Network and Fat-Tree

A data center network interconnects tens of thousands of, or even hundreds of thousands of servers, and provides routing service to upper-layer applications. Large-scale distributed computations run on these servers and high volumes of traffic are exchanged among the servers. In order to accelerate traffic transfer in bandwidth-hungry applications, "rich-connected" topologies are proposed to increase the network capacity. A typical characteristic of such networks is that usually more than one path exists between any pair of servers.

Fat-Tree [4] is one of the representative "rich-connected" topologies, as shown in **Fig. 1**. The switches are organized into three levels. For a K-array Fat-Tree network (i.e., built with K-port switch), there are K pods (K = 4 in the example), each containing two levels of $K/2$ switches, i.e., the edge level and the aggregation level. Each K-port switch at the edge level uses $K/2$ ports to connect the $K/2$ servers, and uses the remaining $K/2$ ports to connect the $K/2$ aggregation-level switches in the same pod. At the core level, there are $(K/2)^2$ K-port switches and each switch has one and only one port connecting to one pod. The total number of servers supported in Fat-Tree is $K^3/4$. For two servers in different pods in Fat-Tree network, there are $(K/2)^2$ paths between them.

In Fat-Tree network, when the packets are forwarded from lower level switches to higher level switches, there are multiple next hops to choose. While there is only one next hop if the



▲**Figure 1. A Fat-Tree network with K = 4.**

packets are forwarded from higher level switches to lower level switches. Let us take the Fig. 1 as an example, and assume a packet needs to be transferred from server 0 to server 4. When the packet arrives at switch E1, it has two next hops, i.e., A1 and A2. Let us suppose that it chooses A1. After arriving at A1, the packet still has two choice, i.e., C1 and C2. After arriving at the core level switch, the packet should be forwarded downwards, and there is only one choice. For instance, if the packet chooses C1 as the next hop at A1, there is only one path to reach server 4 from C1, i.e., C1→A3→E3→4. Similarly, if C2 is chosen as the next hop at A1, the sole path to server 4 is C2→A3→E3→4.

In order to fully utilize the high network capacity of the advanced data center network topologies and provide upper-layer applications with high network throughput, many multipath routing schemes are proposed. Based on the splitting granularity, proposed multi-path routing schemes can be divided into two categories, namely, flow-level path splitting and packet-level path splitting. The former guarantees that packets from the same flow traverse the same path while the latter does not. Besides, multi-path TCP is also designed to utilize the multiple paths in transport level. In what follows we describe the related works respectively.

2.2 Multi-Path Routing with Flow-Level Path Splitting

As a traditional multi-path routing scheme based on flow-level path splitting, ECMP hashes the 5-tuple of every packet to determine the next hop from multiple candidates. VL2 [5] also depends on ECMP to utilize the multiple links in a Fat-Tree like network. However, ECMP fails in balanced utilization of the multiple candidate paths due to the following reasons. First, the random feature of hashing may cause unequal number of flows put in the candidate paths. Second, flows contain different numbers of packets. Hashing collision may forward flows with more packets to the same path, resulting in imbalanced traffic volume. Third, even flows equal in the numbers of packets, the packet size may be different, which also leads to traffic imbalance. Fig. 1 shows an example of hashing collision in ECMP. There are two flows, one from server 0 to server 4, while the other from server 2 to server 5. When switch A1 adopts ECMP to hash the two flows, a collision occurs and both flows choose the link A1→ C1. As a result each flow only grabs half of the link bandwidth. But if we can schedule the flow from server 2 to server 5 to use the path of E2 → A2 → C3 → A4 → E3, no collision exists and each flow can send data with full speed.

In order to overcome the hash collision problem of ECMP, Hedera [8] and Mahout [14] adopt a centralized way to schedule big flows, while using ECMP only for small flows. Hedera depends on edge switches to identify the big flows. Once the bandwidth consumed by the flows exceeds a pre-set threshold (i.e., 10% of the link bandwidth), these flows are identified as big flows. The centralized controller periodically collects information of big flows from edge switches, calculates routing path for each big flow, and installs the routing entries on corresponding switches. Mahout [14] identifies big flows at hosts by detecting the socket buffer taken by the flows, and uses Type of Service (TOS) field in IP header to tag big flows. Each edge switch only needs to install a single routing entry to redirect packets to the centralized controller before routing entries are installed, which greatly reduces the number of routing entries installed on edge switches. Although Hedera and Mahout improve the flow-level path splitting algorithm by spreading big flows into different paths, traffic imbalance among paths still exists when flows are with unequal numbers of packets or unequal packet sizes.

2.3 Multi-Path Routing with Packet-Level Path Splitting

By packet-level path splitting, packets from a flow are distributed to all the candidate paths. Since packets of the same flow are usually of similar sizes, packet-level path splitting can achieve much more balanced utilization of the multiple links. Though it is widely concerned that packet-level splitting may cause packet reordering and confuse TCP congestion control algorithm, the recent work of RPS [9] shows promising results by exploiting the topological feature of data center networks. RPS defines a group of links as an equivalence class, which includes all the outgoing links from the switches at the same hop along all the equal-cost paths [9]. As Fig. 1 shows, links E8→A7 and E8→A8 belong to an equivalence class. RPS tries to keep equal-cost paths between any source-destination pair with similar load by randomly spreading traffic into the links of equivalence class. Considering that the equal-cost paths have the same lengths, if they have similar load as well, the end-to-end latencies along the paths will also be similar. It benefits ordered delivery of packets from different paths, reducing unnecessary FRs at the receiver. But the random packet splitting used in RPS may not result in exactly balanced link utilizations, which will lead to problems as we will show later in this paper.

DRB [10] employs the structure feature of Fat-Tree network and adopts IP-in-IP encapsulation/decapsulation to achieve balanced traffic splitting. In Fat-Tree network, there are many candidate routing paths between each source-destination pair, and each routing path exclusively corresponds to a core switch or an aggregation switch, which is called bouncing switch. For each packet, once the bouncing switch which the packet traverses is picked, the routing path is determined as well. DRB employs the sender to pick a bouncing switch for a packet, and uses IP-in-IP encapsulation to force the packet to take the expected routing path. To mitigate packet reordering, DRB adds a re-sequencing buffer in the receiver below TCP, which stores the reordered packets and postpones delivering the reordered packets to TCP. However, this solution also has the following shortcomings. First, IP-in-IP encapsulation introduces an additional packet overhead of 20 bytes. Second, each connection is

equipped with a re-sequencing buffer, and a timer is set up for each reordered packet, which occupies considerable storage and computation resources at the receiver. Third, the re-sequencing buffer causes additional delays to deliver a packet to upper layers, which may affect applications with real-time requirements.

2.4 Multi-Path TCP (MPTCP)

MPTCP [15]–[17] is a transport-layer solution to efficiently utilize the available band-widths in multiple paths. MPTCP splits a flow into many sub-flows, and each sub-flow independently picks a routing path from the available candidates. To achieve fairness and improve throughput, MPTCP couples all the sub-flows together to execute congestion control [17], so as to shift traffic from more congested paths to less loaded ones. In order to eliminate the negative effect of packet reordering, apart from global sequence space, each sub-flow also has its own subsequence space [15]. Each sub-flow uses its own subsequence number to conduct congestion control just as the standard TCP does. A subsequence space to global sequence space mapping scheme is proposed to assemble data at receivers. Compared with multipath routing schemes, MPTCP focuses more on congestion control and fairness issues, paying the overhead for establishing and maintaining the states of sub-flows.

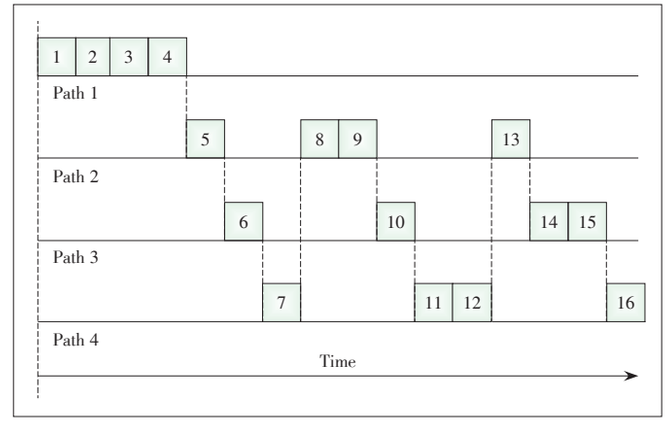
3 SOPA Design

SOPA is a multi-path routing scheme based on packet-level path splitting. In this section we firstly analyze the problems with random packet splitting, then we present the design details of SOPA including two technical components, namely, increasing the FR threshold (to mitigate the impact of packet reordering on lowering a flow's throughput) and source-routing based packet splitting (to mitigate packet reordering).

3.1 Problems with Random Packet Splitting

We start with discussing the problems of random packet splitting, in which every switch randomly splits packets from a flow into equal-cost next hops. From statistical perspective, random packet splitting may lead to similar traffic loads among all the candidate paths during the whole transmission period. However, given a specific short time interval, splitting packets in a random manner cannot guarantee allocating the same amount of traffic to each candidate path. If the load difference among the paths is enough to cause packet reordering and confuse TCP congestion control algorithm to trigger FR, the throughput of the flow will degrade significantly.

Fig. 2 illustrates an example of random packet splitting. We assume the sender's congestion window is big enough to send out 16 packets without waiting for ACKs from the receiver, and there are 4 candidate paths between the two ends. Each box in the figure represents a packet. We can see that each path gets the same share of packets (4 packets on each path) during the transmission period. But at the beginning of the transmission, the first 4 packets are all allocated to path 1. If unfortunately other flows also allocate packets as shown in Fig. 2, path 1 may have larger queuing delay than other paths in the initial transmission period. The difference in queuing delay can lead to packet reordering at the receiver.



▲ **Figure 2. An example to illustrate that random packet splitting may cause packet reordering and FR. (Each box denotes a packet, and the number represents the sequence number of the packet. Although random packet splitting allocates 4 packets to each path during the whole period, the instant loads of the paths are different, leading to difference in the queuing delays of the paths. The arrival order of the first 7 packets can be: 1, 5, 6, 7, 8, 2, and 3, which will result in a FR and degrade the throughput of the flow.)**

We assume the arriving order of the first 7 packets is: 1, 5, 6, 7, 8, 2, 3. In this case, each reordered packet (packet 5, 6, 7 and 8) will prompt the receiver to send an ACK for the expected packet to the sender, i.e., packet 2. According to the default setting, the three duplicate ACKs will lead the sender into FR phase, cutting down the congestion window into half. So the network throughput drops even though the network is not congested at all. Although the example is just an illustrative one, the simulation below based on NS-3 indeed demonstrates this problem.

In this simulation, we use a Fat-Tree network built by 4-port switches, as shown in Fig. 1. There are 16 servers in the network. A single flow is established between server 0 and server 5, and server 0 sends 100 MB of data to server 5. We set different capacities to links in different levels to intentionally set different oversubscription ratios [4]. All the links connecting servers and edge-level switches as well as the links connecting edge-level and aggregation-level switches have 1 Gbit/s bandwidth, while the capacity of links connecting aggregation-level and core-level switches varies from 1 Gbit/s, 750 Mbit/s, 500 Mbit/s, to 250 Mbit/s. In other words, the oversubscription ratio of the Fat-Tree network varies from 1:1, 4:3, 2:1, to 4:1, respectively.

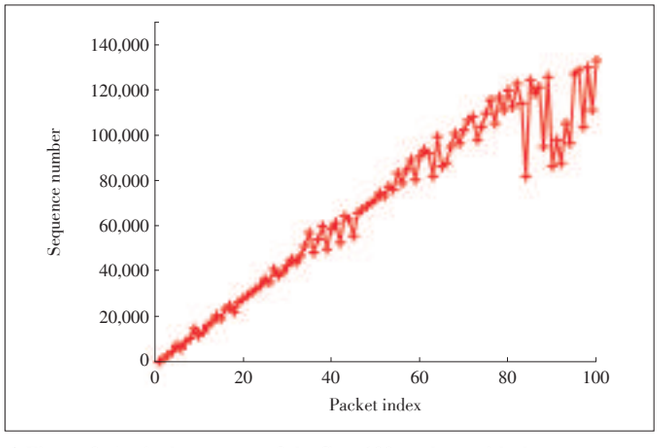
There are 4 candidate paths between server 0 and server 5 (each corresponding to a core-level switch). As a result, the ideal network throughput for the flow in all the scenarios is 1 Gbit/s.

However, when oversubscription ratio varies from 1:1, 4:3, 2:1, to 4:1, the actual throughput of the flow is 986.06 Mbit/s, 966.31 Mbit/s, 578.42 Mbit/s and 296.03 Mbit/s respectively. We can see that the throughput under random packet splitting degrades significantly when the oversubscription ratio grows. When the oversubscription ratio is 4:1, the throughput is only 296.03 Mbit/s, much lower than the ideal value (1 Gbit/s). The reason is that as the oversubscription ratio increases, the bandwidth of links between aggregation-level and core-level switches become smaller. When packets are forwarded upwards, a bottleneck will be built between these two levels of switches, resulting in longer queuing delay. Furthermore, the imbalanced traffic splitting illustrated in Fig. 2 allocates different traffic loads to the candidate paths, and the packets on light-loaded paths will experience shorter delay than that allocated to the more heavily-loaded paths. When the oversubscription ratio is higher, the queuing delay in the bottleneck paths will be longer and the impact of traffic imbalance on the packet reordering will be more significant. Therefore, more FRs are triggered when the oversubscription ratio is higher, resulting in poorer performance.

To validate our analysis, we record the arrival sequence of the first 100 packets under the oversubscription ratio of 4:1, as shown in **Fig. 3**. The x-axis of Fig. 3 denotes the arrival order of packets on server 5, and the y-axis shows the sequence number of each received packet. We observe many reordered packets. These reordered packets send enough duplicate ACKs back to the sender to trigger FR. Trace data shows that, during the whole transmission period (100 MB of data transmission), 347 times of FRs occur at the sender, causing it to resend 2406 packets in total (i.e., 3.35% of all the packets). The FRs reduce the congestion window at the sender and thus degrade the flow's throughput.

3.2 Increasing FR Threshold

Since FR due to packet reordering is the root cause of the undesirable performance of random packet splitting, we want to understand why random packet splitting brings so many FRs. To answer this question, we briefly review the congestion control algorithm of TCP, particularly, the FR algorithm.



▲**Figure 3. Arrival sequence of the first 100 packets with the oversubscription ratio of 4:1. (The random packet splitting causes many reordered packets.)**

In order to ensure reliable data delivery, TCP adopts the acknowledgement scheme, i.e., once receiving a data packet, the receiver sends an ACK message back to the sender. To improve efficiency, modern TCP does not send an ACK for each received packet. Instead, the receiver uses an ACK to acknowledge a batch of sequential packets. However, a reordered packet will prompt an ACK to be sent out immediately. The sender sets a retransmission timer for each unacknowledged packet. If a sent packet or its ACK is dropped, the sender does not know whether the packet has been correctly received or not, and it will resend the packet when the retransmission timer timeouts. This scheme might result in low throughput, because once a packet is lost, TCP has to wait for the expiration of the retransmission timer to resend the packet. During this timeout period, no more new packets can be sent since the congestion window does not slide forward. To tackle this issue, FR is proposed, which is triggered by three duplicate ACKs.

3.2.1 Algorithm of FR

We use one example to illustrate the working process of FR. In this example, the sender sends out 5 packets in a batch. Unfortunately the first packet is lost, while the 4 subsequent ones successfully arrive at the receiver, each triggering an ACK since it is a reordered packet. Based on today's TCP configuration, three duplicate ACKs trigger the sender to immediately resend the first packet, rather than waiting for its timeout. The FR algorithm is widely used for TCP congestion control in the single-path routing, in which packets belonging to the same flow always traverse the same path if there is no topology change and with high probability a reordered packet indicates there is packet loss due to congestion. However, in the setting of packet-level multi-path routing, packets from the same flow would go through different paths, which may have different loads and queue lengths. As a result, the receiver will get more reordered packets, even if there is no congestion and packet loss in the network.

3.2.2 Benefit and Problem with Increasing FR Threshold

Since packet reordering is most unlikely the sign of congestion in packet-level multi-path routing and unnecessary FRs are the key reasons to lower the throughput, an intuitive idea is to increase the FR threshold to avoid FRs as much as possible. However, does increasing FR threshold really help improve the flow's throughput? And if it does, how large should the FR threshold be? To answer these questions, we need to differentiate two cases, namely, whether there is packet loss or not.

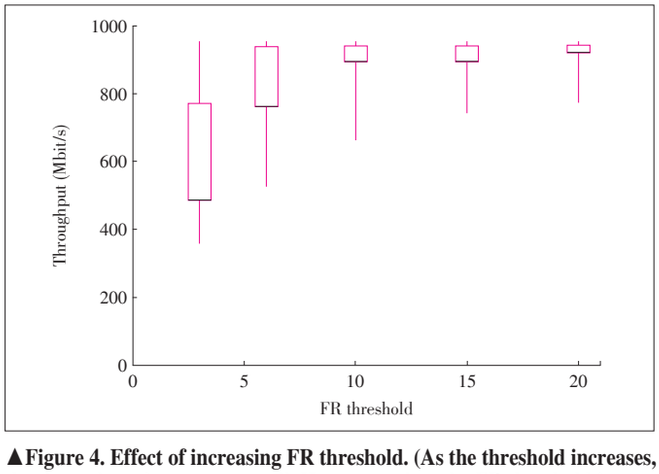
If there is no packet loss, all the reordered packets will finally arrive the receiver, and increasing the FR threshold can efficiently avoid unnecessary packet retransmissions caused by

packet reordering and accordingly greatly improve the flow's throughput.

However, if packet loss indeed occurs, the situation is a little bit tricky. If there are enough subsequent packets after the lost packet arriving at the receiver, enough duplicate ACKs can be received by the sender to trigger FR, even if we increase the FR threshold. The only difference is that the time to resend the lost packet(s) is postponed. Considering the Round-Trip Time (RTT) in data center network is very small (e.g., several hundred microseconds), we believe that the performance enhancement by increasing the FR threshold outweighs the introduced delay. However, if there are not enough subsequent packets after the lost packet to trigger FR (e.g., when the congestion is window is small), the lost packet will be transmitted after the retransmission timer timeouts. In this case, increasing the FR threshold will result in more packet retransmissions by timeouts, which inversely degrades the flow's throughput since the TCP's retransmission timeout value (RTO) is much larger than the RTT in data center network.

We use two examples simulated in NS-3 to study the effect of increasing the FR threshold in random packet splitting.

Example 1: We use a 24-array Fat-Tree network and a permutation workload [9], in which each server is a sender or receiver for just one flow. In this workload, there are at most 144 parallel paths for every flow. There is no packet loss in this case, since Fat-Tree network is non-blocking. The FR threshold is set as 3 (default value in TCP), 6, 10, 15, and 20, respectively. **Fig. 4** shows the flows' throughputs against the FR threshold. For each candlestick in the figure, the top and bottom of the straight line represent the maximum and minimum value of the flows' throughputs, respectively. The top and bottom of the rectangle denote the 5th and 99th percentile of average throughput, respectively. The short black line is the average throughput of all the flows. We can see that the throughput indeed improves as the FR threshold increases. By checking the traces, we find that when the FR threshold is 3, 6, 10, 15 and 20, FR takes place for 28708, 6753, 516, 62 and 3 times, respectively.



▲**Figure 4. Effect of increasing FR threshold. (As the threshold increases, the throughput improves as well. However, when the FR threshold is larger than 10, the improvement of performance is quite marginal.)**

However, when the FR threshold is larger than 10, the improvement of average throughput is quite marginal: the average throughput is 896.24 Mbit/s, 919.82 Mbit/s, and 921.16 Mbit/s when the FR threshold is 10, 15, and 20, respectively. Besides, the minimum flow throughput is also less than expected, because FRs are still triggered even the threshold is larger than 10. When the FR threshold is 10, 15 and 20, there are 516, 62 and 3 flows experiencing FR, and the average number of FRs for each flow is 2.02, 1.26 and 1 time(s), respectively. If we continue increasing the FR threshold, indeed we can eliminate FR, but the following example will show that we cannot increase the FR threshold to an unlimited value.

Example 2: We use the same network topology as Example 1 and assume a client fetches data from 3 servers to mimic an operation in distributed file system and the sum of data size is 64 MB. During the file transmission period, a short flow starts to send small data (64 KB) to the same client. In our simulation, when the FR thresholds are 3, 6, 10, 15, and 20, the throughputs of the short flow are 53.16 Mbit/s, 61.14 Mbit/s, 57.84 Mbit/s, 2.41 Mbit/s and 2.42 Mbit/s, respectively.

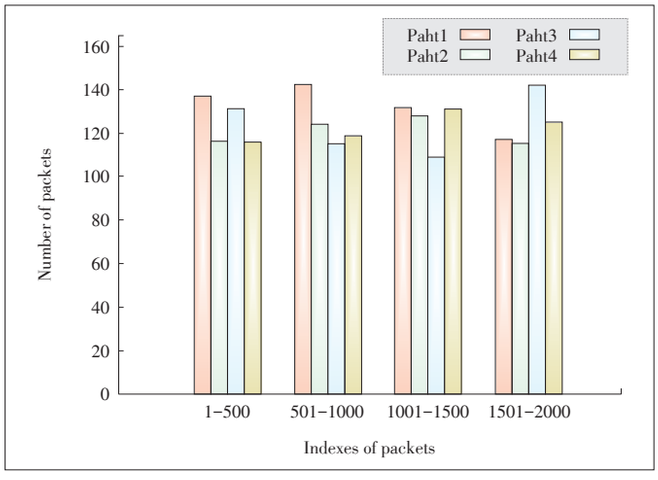
The two examples above demonstrate that increasing the FR threshold has both benefits and problems for the throughput improvement. On one hand, higher FR threshold helps avoid many unnecessary FRs caused by packet reordering. On the other hand, when packet loss indeed occurs, higher FR threshold also causes more timeouts of retransmission timers and thus smaller congestion window. We thus argue that simply increasing the FR threshold alone cannot solve the problem thoroughly. In SOPA we set the FR threshold as 10 (motivated by Examples 1 and 2), and seek for more balanced packet splitting solutions among the equal-cost paths.

3.3 Source-Routing Based Packet Splitting

3.3.1 Design Rationale

We first run a simple simulation to demonstrate that random packet splitting can lead to aggravated packet reordering. We setup a single flow in a 4-array Fat-Tree network, which sends 100 MB data. There are 4 parallel paths between the sender and the receiver. During the entire transmission period, the percentiles of packets allocated to the four paths are 24.89%, 25.10%, 25.01% and 25.00% respectively. However, within every 500 consecutive packets, the number of packets allocated to each path deviates significantly from each other. For the first 2000 packets, **Fig. 5** shows the number of packets allocated to each path for every 500 consecutive packets.

We learn from the figure that the maximum deviation from the average value can be up to 13.6% (this observation well matches the example shown in Fig. 2), even though the overall ratios are roughly close to each other. Due to the imbalanced traffic allocation, some paths may experience congestion and



▲Figure 5. Packets allocation in random packet splitting. (This figure shows how the first 2000 packets are allocated to 4 equal-cost paths. Each group of square columns represents the allocation of 500 packets. Even though almost the same traffic is allocated to each path during the whole transmission period, the instant allocations to the paths are different. The maximum deviation from the average allocation is 13.6%.)

even packet loss, while others do not. Consequently, we need a solution which can avoid packet reordering instead of tolerating packet reordering only.

Rather than randomly selecting the next hop for a packet, SOPA explicitly identifies the routing paths for packets of a flow in a round-robin fashion, which results in exactly balanced utilization of the equal-cost paths regardless of the workload. There are two possible solutions to explicit path identification, namely, switch based and server based.

In switch-based solution, we can let each switch explicitly forward the packets from a flow to the interfaces in a round-robin way. However, it requires the switch to maintain the state for each flow, i.e., records the forwarding interface of the last packet from the flow. Since the number of concurrent flows in data center network is huge [5] and the technical trend for today's data center network is to use low-end commodity switches [4]–[6], it is quite challenging, if not impossible, to implement this idea in a practical data center switch.

In server-based solution, the senders are responsible for explicitly identifying the routing path of each packet and inserting the routing information into the packet, i.e., using source routing. Switches only need to support source routing, without the requirement to maintain per-flow state. It has been shown feasible to realize source routing by programming today's commodity switching chips [12]. Moreover, source routing is even easier to be implemented in the emerging SDN/OpenFlow paradigm, by appropriately configuring the flow tables in switches [13]. In one word, source-routing based packet splitting puts the complexity into servers which have sufficient calculation power and memory [18], while does not need to update the switches' chips.

SOPA further reduces the packet overhead caused by source routing in a Fat-Tree network by exploiting the topological feature. In Fat-Tree network, we define upward forwarding as forwarding packets from a lower-level switch to a higher level switch (i.e., from edge switch to aggregation switch, or from aggregation switch to core switch); and define downward forwarding as forwarding packets from a higher-level switch to a lower-level switch (i.e., from core switch to aggregation switch, or from aggregation switch to edge switch). Although there are 6 hops at most for a source-destination pair (if source and destination are located in different pods), the routing path is determined by the two upward forwarding hops. Therefore, at most two intermediate switches are inserted into the IP headers to support source-routing in SOPA. For each hop in upward forwarding, we use 1 byte to store the forwarding interface. It can theoretically support a 512-array Fat-Tree network (noting that only half of a switch's interfaces are upward ones), which includes 33,554,432 servers in total. Since we only need to store up to two hops' information, the overhead introduced by source routing in SOPA is at most 4 bytes (one byte is option-type octet, two bytes are for two hops' information, and the fourth byte is a padding byte to ensure that the IP header ends on a 32 bit boundary). For a packet size of 1500 bytes, the traffic overhead is only 0.26%.

3.3.2 Technical Components

SOPA includes three major technical components to realize the source-routing based packet splitting, which are described as follows.

Calculating Candidate Paths: This component runs at servers. Flows in a Fat-Tree network can be divided into three categories, namely, inter-pod flows (i.e., the source and destination are in different pods), inter-rack flows (i.e., the source and destination are in the same pod but different racks), and intra-rack flows (i.e., the source and destination are under the same edge switch). The path lengths (in terms of number of hops) for the three categories of flows are 6, 4 and 2, respectively. For the intra-rack flows, there is only one path between the senders and receivers, and we do not need to calculate the candidate paths at all. When calculating the routing paths for inter-pod flows and inter-rack flows, we only need to focus on upward forwarding hops. It is worth noting that the candidate paths should exclude the ones with faulty links. We assume there is a failure notification scheme that the source can be aware of the updated topology, which is easy to realize if there is a central controller in the data center.

Selecting Flow Path: This component runs at servers too. Before sending out a packet, the sender needs to select a routing path from the candidate ones for the packet. The sender chooses the paths for outgoing packets in a round-robin fashion to make sure that each path grabs the same share of traffic from the flow. Once the path is chosen, the sender inserts the path information into the optional field of IP header. Given the powerful processing capability and large memory in today's data

center servers, the processing overhead is affordable.

Packet Forwarding: This component runs at switches. The switches' function in SOPA is simply forwarding packets according to the source routing information (upward forwarding) or the destination address (downward forwarding). In an SDN/OpenFlow network [13], forwarding keys are extracted from the packet header and matched against the flow table. In a regular network implementing source routing, in upward forwarding the next hop is popped from the source routing field and used as a key to lookup the forwarding table; while in downward forwarding the table lookup is operated on the destination address.

3.3.3 Benefit and Problem with Source Routing

Fig. 6 shows the performance between source-routing based packet splitting and random packet splitting. We run the same simulation as in Section 3.2 and also set the FR threshold as 3, 6, 10, 15 and 20, respectively. From the simulation results we find that source routing outperforms random splitting under all settings. Both the two solutions improve the flows' throughputs as the FR threshold increases. However, when the FR threshold reaches 10, source routing achieves close-to-optimal throughput for all the flows, by the balanced traffic splitting among the equal-cost paths.

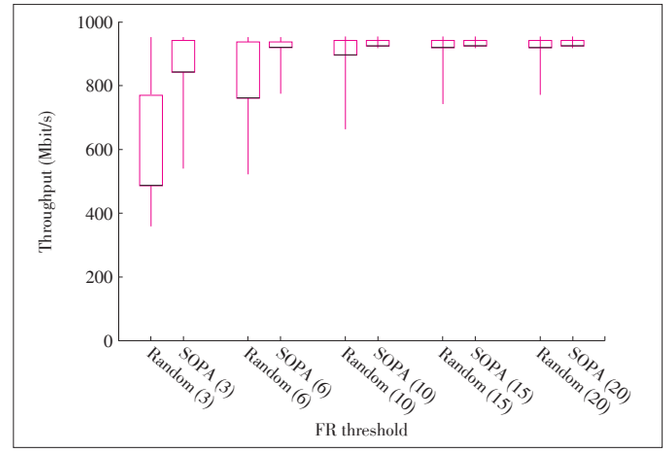
We can also observe from Fig. 6 that using source routing alone cannot completely avoid unnecessary FR. When source routing is employed and the FR threshold is set to 3, the receivers see 2.83% reordered packets, and 2728 times of FRs are triggered (in contrast there are 10.96% reordered packets and 28,708 times of FRs in random packet splitting). The reason is that even we use source routing to ensure balanced traffic splitting, the end-to-end delays of the candidate paths are not the same (the simulations in Section 4.2 show this phenomenon). So the arrival order of packets is not strictly consistent with the sending order. Furthermore, in production data center networks there may be other reasons causing packet reordering, such as the buggy driver of network interface cards (NICs).

As a result, in SOPA we employ both increasing the FR threshold (to 10) and source-routing based packet splitting to improve the flows' throughputs in packet-level multi-path routing. Note that SOPA does not need to update the switch hardware. The modifications on server side is also light-weighted. We only need to rewrite the FR threshold number in TCP congestion control algorithm and maintain the routing path of the last packet for every outgoing flow.

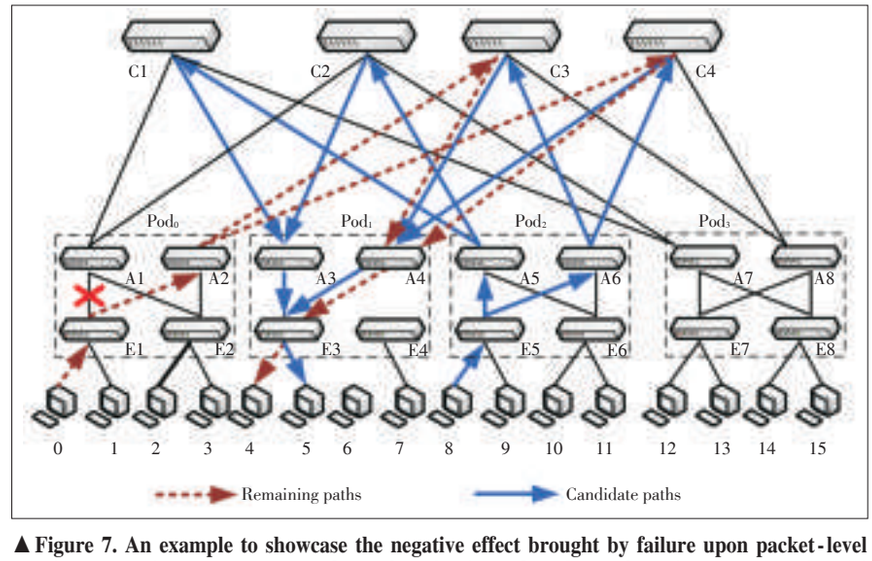
3.4 Failure Handling

The discussion above assumes there is no failure and the network topology is symmetric. However, failures are common in data center networks instead of exceptions [4], [19]. The failures will break the symmetry of the network, and imbalanced traffic load will be allocated to candidate paths, resulting in more aggravated packet reordering. We use an example shown in **Fig. 7** to explain the negative effect brought by failure upon packet-level multi-path routing.

There are two flows, one from server 0 to 4 while the other from server 8 to 5. If there is no failure, both flows have four candidate paths. Now we assume the link between E1 and A1 is broken. Because some paths of flow 0→4 contain the faulty link, the flow can only take two paths, i.e., E1→A2→C3→



▲**Figure 6. Performance comparison between random packet splitting and SOPA. (The number in the parentheses denotes the FR threshold. Both random packet splitting and SOPA improve the performance as the threshold increases, and SOPA outperforms random packet splitting in all settings.)**

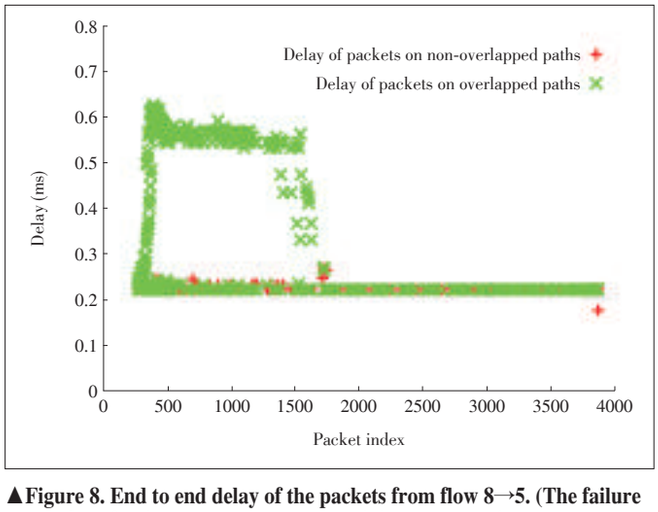


▲**Figure 7. An example to showcase the negative effect brought by failure upon packet-level multi-path routing. (There are two flows, flow 0→4 and flow 8→5. If the link between E1 to A1 fails, the flow (0→4) can only take the two remaining paths, while the other flow (8→5) can still use the four candidate paths, which may cause load imbalance across multiple paths of flow 8→5, degrading its performance.)**

A4→E3 and E1→A2→C4→A4→E3 (displayed as dotted arrow lines in the Fig. 7 and called remaining paths). On the contrary, flow 8→5 can still use the four candidate paths (displayed as solid arrow lines in Fig. 7). Both flows evenly allocate traffic to its available paths, and load imbalance occurs across the four candidate paths of flow 8→5. Different queue lengths will be built among the four paths, leading to aggravated packet reordering at the receiver (server 5).

Fig. 8 validates our analysis, which shows the end to end delay of all packets from flow 8→5. The figure displays the delays for two groups of packets. One group of packets takes the paths which overlap with the remaining paths of flow 0→4, and these paths are called overlapped paths. The other group of packets takes the paths that do not overlap with the remaining paths of flow 0→4, and these paths are called non-overlapped paths. For the first 1700 packets, the packets allocated to the overlapped paths experience much longer delay than those allocated to the non-overlapped paths. Subsequent packets experience almost the same delay. It is because during the initial transmission period, two flows run simultaneously, and the traffic load of the overlapped paths is higher than that of the non-overlapped paths. The different traffic loads across the candidate paths of flow 8→5 result in much more reordered packets at the receiver (i.e., server 5), leading to degraded performance. So flow 0→4 finishes earlier than flow 8→5. In the late stages of data transmission, there is only flow 8→5, so the packets from all paths get similar delays.

From the example, we see that link failure can break the balanced traffic allocation, resulting in more aggravated packet reordering. One possible solution is to employ random early detection (RED) for flows that experience link failure, as in [9]. The goal is to reduce the transmission rate of the flow, decrease the difference in queue lengths of the candidate paths,



▲**Figure 8. End to end delay of the packets from flow 8→5. (The failure causes flow 0→4 only can take two remaining paths, which are overlapped with two candidate paths of flow 8→5. The figure shows the packets on the overlapped paths experience much longer delay than the packets allocated to the non-overlapped paths.)**

and mitigate the packet reordering introduced by failure. We argue that this solution does solve the problem efficiently. RED only starts to tag packets when the queue length exceeds a threshold. Before reducing the transmission rate (which is caused by tagged ACK), the flows still send data as if no failure occurs. As a result, different queue lengths can still be built among the candidate paths of the flow whose paths are not affected by the failure, just as Fig. 8 shows.

Another possible solution is to introduce a re-sequencing buffer at the receiver to absorb the reordered packets, as in [10]. Reordered packets are restored in the re-sequencing buffer and postponed to deliver to TCP. A timer is set for each reordered packet. If an expected packet arrives before timeout, the timer is canceled and this packet along with in-sequence packets are delivered to TCP. Otherwise, the timer expires and the reordered packet is handed to TCP to send back ACKs. Given no packet loss, the solution works fine. However, if some packets are dropped, the buffer has to wait for the expiration of the timer to deliver the packets to TCP, and an additional delay is introduced.

Packet reordering is the main negative effect brought by failure. As long as unnecessary FRs are avoided, the performance can still be guaranteed. SOPA increases the FR threshold, which can effectively avoid unnecessary FRs due to reordered packets and greatly mitigate the negative effect brought by failure. Even in case of packet loss, reordered packets can also produce duplicate ACKs in time to trigger FR. We will evaluate our design under scenarios with failure in Section 4.5.

4 Evaluation

4.1 Simulation Setup

In this section, we evaluate the performance of SOPA using NS-3 [20], which is a packet-level simulator. We use Fat-Tree as the data center network topology. Given the switches composing the Fat-Tree network have K ports, the total number of servers in the network is $K^3/4$. We set K=24 by default unless specified otherwise. The bandwidth of each link is set to 1 Gbit/s, and the latency of each link is 25 ns. The packet size is set as 1500 bytes (including IP header and TCP header). For SOPA, the FR threshold is 10, and for the other schemes, we use TCP's default configuration, i.e., FR threshold of 3. From previous analysis, we know that the throughput of random packet splitting degrades when the oversubscription ratio increases, so we set 1:1 oversubscription ratio for all the simulations, which actually favors random traffic splitting.

In the following simulations, we use RPS as the implementation of random traffic splitting. We firstly use a small-scale simulation to demonstrate that the random packet splitting creates imbalanced load distribution and degraded performance. Then we compare SOPA with ECMP, Hedera and RPS by using large-scale simulations. We implement Hedera following

[8], and use Global First Fit algorithm to calculate the routing paths for big flows. We use two workloads, i.e., permutation workload and production workload, to study various traffic scenarios (details about these two workloads will be introduced later). At last, we also evaluate the performance of SOPA with link failures.

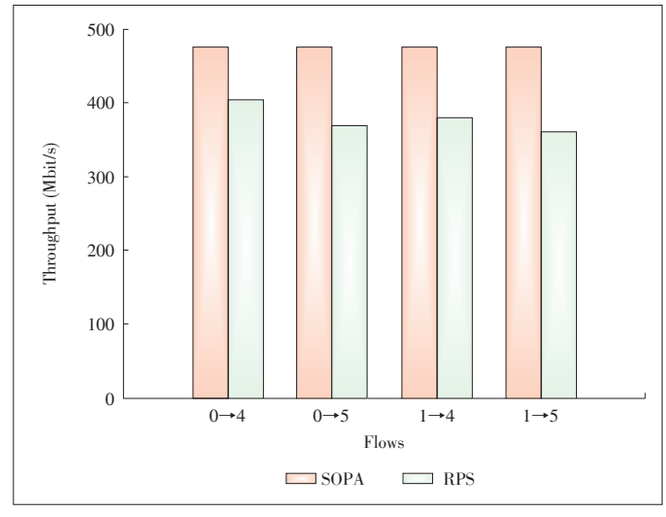
4.2 Negative Effect of Imbalanced Load Distribution

In this section, we use a small-scale simulation to demonstrate the negative effect of imbalance load distribution. A 4-array Fat-Tree network, just as Fig. 1 shows, is used in this simulation. We set up four flows in the simulation, i.e., flow 0→4, 0→5, 1→4, and 1→5, respectively. Each flow sends 10 MB data to its destination. We run both SOPA and RPS in the simulation, respectively.

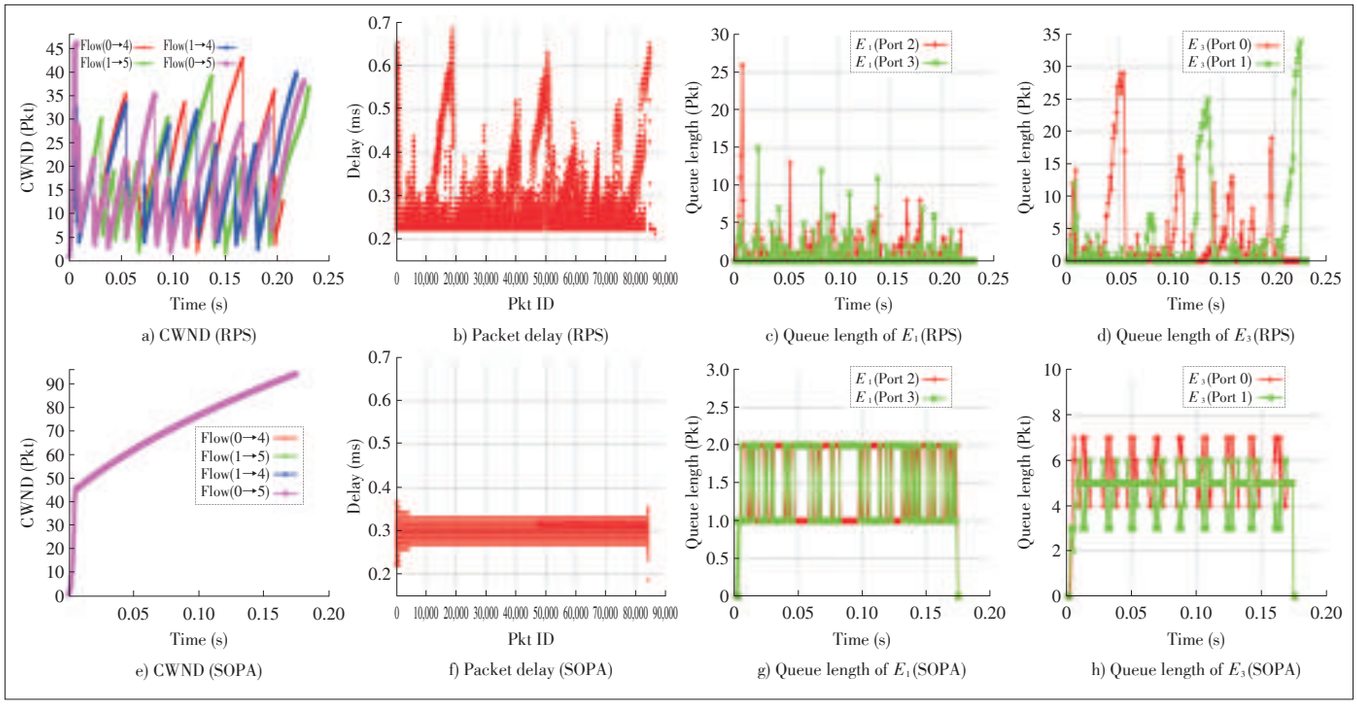
Fig. 9 shows the flows' throughput. When SOPA is used, each flow grabs the fair share of bandwidth, and the throughput of each flow is about 475 Mbit/s. However, when using RPS, the throughput of each flow varies much. The maximum throughput is 404.27 Mbit/s (flow 0→4), while the minimum throughput is 360.55 Mbit/s (flow 1→5). The average throughput of these four flows is only 378.30 Mbit/s. We thoroughly analyzes the reason for the difference in the performance of SOPA and RPS as follows.

4.2.1 Evolvement of CWND

First, we check the evolvement of the congestion window (CWND), as shown in **Figs. 10a** and **10e**. Fig. 10a demonstrates that the size of congestion window of each flow varies much when RPS is used, because each flow experiences many times of FR. For the 4 flows, the FR takes place for 23, 24, 23 and 31 times, respectively. Each FR halves the size of congestion window, and the throughput drops down accordingly. On the contrary, Fig. 10e showcases that when SOPA is employed, the congestion window of each flow increases monotonically during the whole transmission period (from slow start phase to congestion avoidance phase). Therefore, SOPA achieves good



▲**Figure 9. Throughput of 4 flows. (SOPA allocates traffic evenly, and each flow grabs fair share of bandwidth, and the throughput of each flow is about 475 Mbit/s. However, RPS fails to achieve balanced traffic allocation, the average throughput of these four flows is only 378.30 Mbit/s.)**



▲**Figure 10. Comparisons between SOPA and random packet spraying.**

performance, while RPS does not.

4.2.2 End-to-end Packet Delay

To find out why RPS causes so many reordered packets and then many unnecessary FRs, we measure the end-to-end packet delay. **Figs. 10b** and **10f** show the packet delay when RPS and SOPA are employed, respectively. Fig. 13b shows that the packets' delays vary heavily in RPS. The maximum packet delay is 687.4 us, while the minimum packet delay is 211 us. However, the end to end delays of all packets fluctuate within a small range when SOPA is adopted, which is shown in Fig. 10f. The maximum packet delay is 366.27 us, while the minimum packet delay is 187 us. And SOPA only introduces 0.4% reordered packets during the simulation, while 14.92% packets experience reordering when RPS is adopted. This aggravated packet reordering can be attributed to imbalanced traffic splitting of RPS, which builds up different queue lengths on switches and packets from different paths experience different delays. We measure the queue lengths to validate our analysis.

4.2.3 Queue Length

In the simulation all the traffic of the four flows goes through the edge switch E1 and E3 in Fig. 1. The two switches are the highest-loaded switches, so we focus on the queue lengths of the two switches. Since all traffic is forwarded upwards at switch E1 and switch E3 forwards all traffic downwards, we measure the queue lengths of all the upward forwarding ports on E1 (i.e., port 2 and port 3) and the queue lengths of all downward forwarding ports on E3 (i.e., port 0 and port 1).

Figs. 10c and **10d** plot the queue lengths on switch E1 and E3 for RPS, respectively. The figures reveal that RPS builds very different queue lengths on different forwarding ports due to imbalanced traffic splitting. We take switch E1 as an example, at 0.008 s, the queue length of the port 2 is 16 packets, while the queue of the port 3 is empty. Switch E3 also has more diverse queue lengths on port 0 and port 1. For example, at 0.2250 s, the queue of port 0 is empty, while the queue length of port 1 is 34 packets. Note that the average queue length of E3 is bigger than that of E1. Since RPS produces imbalanced traffic splitting at each hop, and E1 is the first hop and E3 is the last hop of these flows. As the last hop, the larger queue length of E3 embodies the accumulated imbalanced traffic splitting at previous hops.

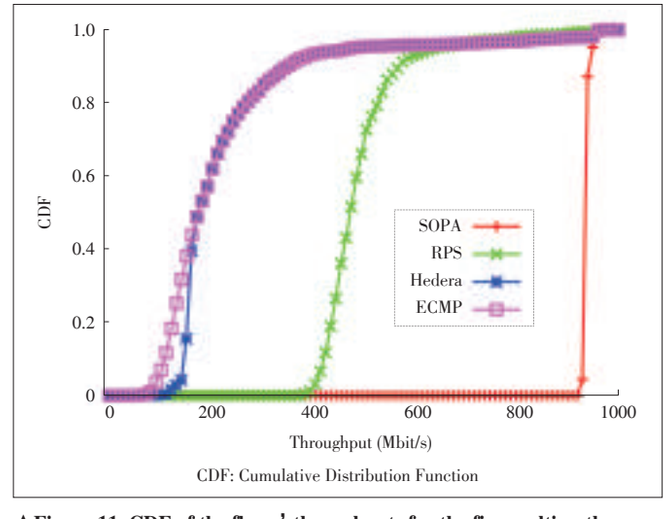
For SOPA, the queue lengths on switch E1 and E3 are shown in Figs. 10g and 10h, respectively. Compared with Figs. 10c and 10d, it is obvious that the queue lengths of different ports on each switch are almost the same, due to the more balanced traffic splitting achieved by SOPA. As a consequence, SOPA does not introduce aggravated packet reordering, and no FR is triggered. However, RPS creates different queue lengths on switches' different forwarding ports, packets on different routing paths may experience different delays (as Fig. 10b shows). Therefore, the receivers see larger number of reordered packets, and many unnecessary FRs are triggered.

For SOPA, the queue lengths on switch E1 and E3 are shown in Fig. 10g and Fig. 10h, respectively. Compared with Fig. 10c and Fig. 10d, it is obvious that the queue lengths of different ports on each switch are almost the same, due to the more balanced traffic splitting achieved by SOPA. As a consequence, SOPA does not introduce aggravated packet reordering, and no FR is triggered. However, RPS creates different queue lengths on switches' different forwarding ports, packets on different routing paths may experience different delays (as Fig. 10b shows). Therefore, the receivers see larger number of reordered packets, and many unnecessary FRs are triggered.

4.3 Permutation Workload

We then study the performance of various multi-path routing schemes, namely, SOPA, RPS, Hedra and ECMP, under synthesized permutation workload. The senders and receivers are picked randomly. We use a Fat-Tree network with K =24, in which there are 3456 servers in total. Each sender sends 10 MB data to its receiver. All flows start simultaneously.

Fig. 11 shows the Cumulative Distribution Function (CDF) of all the flows' throughput. We can see that SOPA significantly outperforms the other three multi-path routing schemes. The average throughput of SOPA is 925.13 Mbit/s, and all the flows' throughputs are above 910 Mbit/s. Compared with SOPA, the average throughput of the flows drops by 47.47% in RPS. The fundamental reason is as explained above: RPS cannot evenly split the traffic across candidate paths, and unequal queue lengths will be built. So the receivers will receive more reordered packets, and unnecessary FRs will be triggered at the senders. As flow-based path splitting solutions, Hedera and ECMP expose even lower performance than RPS. Compared with SOPA, the average throughput drops by 75.21% and



▲**Figure 11. CDF of the flows' throughputs for the five multi-path routing schemes under permutation workload. (Both SOPA and DRB outperform the other three routing schemes, and SOPA also achieves more balanced traffic splitting than DRB.)**

76.48%, respectively. The basic reason is that the flow-based multi-path routing cannot fully utilize the rich link resource in the Fat-Tree network. ECMP achieves the lowest throughput because the hashing results of some flows may collide and be scheduled to the same path, which can be avoided by centralized negotiation in Hedera.

4.4 Production Workload

We next study the performance under a more realistic workloads from a production data center [5]: 95% flows are less than 1 MB, and only less than 2% flows exceeds 10 MB. Different from the permutation workload, in this simulation the data is transmitted by multiple flows instead of a single one. The flows are issued sequentially, and each flow randomly picks a destination server. All the servers start data transmission at the same time, and we measure the average throughput of the data transmission under this workload.

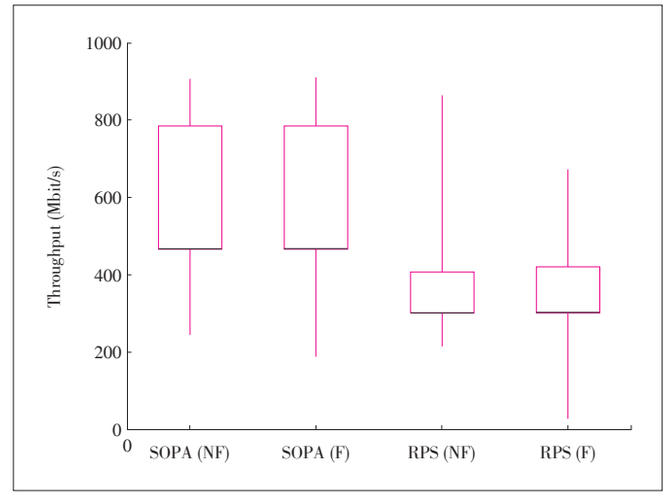
In our simulation, the average throughput for SOPA, RPS, Hedera, and ECMP are 465.5 Mbit/s, 302.96 Mbit/s, 105.005 Mbit/s, and 103.849 Mbit/s, respectively. ECMP gets the lowest throughput, since it neither evenly spreads the flows into multiple paths nor considers traffic sizes in splitting. Hedera performs a little better than ECMP, but the gap is marginal (less than 2 Mbit/s). It is because Hedera targets at scheduling large flows but in this workload 95% flows are small ones (with data size less than 1 MB). Both RPS and SOPA can achieve much higher throughput, due to the fine-grained link utilization by packet-level path splitting. Compared with RPS, SOPA can even improve the throughput by 53.65%, since it explicitly spreads the traffic into the multiple paths in a more balanced way. The result is consistent with that in previous simulations.

4.5 Link Failures

We then evaluate the performance of SOPA when failure occurs. Since failure brings more negative effect for packet-level traffic splitting (introducing more aggravated packet reordering), we only compare SOPA with RPS in this group of simulations. We use production workload to conduct simulation in the same topology as that in the previous simulation. We let the leftmost aggregation switch in the first Pod break down. **Fig. 12** shows the result. In order to showcase the effect of failure, the performance without failure is also plotted.

The x-axis of Fig. 12 denotes both multi-path routing schemes under different settings, wherein "NF" in parenthesis means no failure, and "F" in parenthesis denotes failures. The y-axis shows the throughput of flows. Similarly, for each candlestick in the figure, the top and bottom of the straight line represent the maximum and minimum values of the flow's throughput, respectively. The top and bottom of the rectangle denote the 5th and 99th percentile of average throughput, respectively. The short black line is the average throughput of all flows.

The performance of SOPA is almost not affected by the link failure at all, and only the minimum throughput decreases from 244 Mbit/s to 188.17 Mbit/s. This mild performance degradation is attributed to the high FR threshold of SOPA, which can absorb the more reordered packets introduced by the failure. However, failure brings more negative effects to RPS, and both the maximum and minimum throughput of RPS are dropped. When there is no failure, the maximum and minimum throughput are 865.41 Mbit/s and 214.54 Mbit/s, respectively. But in the failure case, their values drop to 672.25 Mbit/s and 26.17 Mbit/s, respectively. This performance degradation is primarily caused by timeouts of the retransmission timers. Trace data shows that when failure occurs, RPS experiences 67 times of timeout and 12000 packets have been dropped (as a contract, there is no packet loss when no failure). The packet losses are caused by traffic congestion, since RPS cannot achieve balanced traffic splitting and the failure aggravates this situation. However, benefiting from balanced traffic splitting, SOPA does not cause packet loss, and there is not a single timeout in the simulation, with or without failures.



▲**Figure 12. The performance comparison between SOPA and RPS under production workload when failures occur. (In order to show the effect of failure, the performance without failure is also plotted. "NF" means no failure, while "F" denotes that failure has occurred.)**

5 Conclusion

Many "rich-connected" topologies have been proposed for data centers in recently years, such as Fat-Tree, to provide full bisection bandwidth. To achieve high aggregate bandwidth, the flows need to dynamically choose a path or simultaneously transmit data on multiple paths. Existing flow-level multipath routing solutions do not consider the data size, and may lead to traffic imbalance. While the packet-level multipath routing scheme may create large queue length differential between candidate paths, aggravating packet reordering at receivers and thus triggering FR at the senders. In this paper we design SOPA to efficiently utilize the high network capacity. SOPA

adopts source routing to explicitly split data to candidate routing paths in a round robin fashion, which can significantly mitigate packet reordering and thus improve the network throughput. By leveraging the topological feature of data center networks, SOPA encodes a very small number of switches into the packet header, introducing a very light overhead. SOPA also immediately throttles the transmission rate of the affected flow as soon as the failures are detected to promptly mitigate the negative affect of failures. NS-3 based simulations show SOPA can efficiently increase the network throughput, and outperform other schemes under different settings, irrespective of the network size.

Reference

[1] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The google file system," in *Proc. 19th ACM Symposium on Operating Systems Principles*, New York, USA, 2003, pp. 29–43.
[2] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," in *Proc. 6th Symposium on Operating Systems Design and Implementation*, Berkeley, USA, 2004, pp. 137–149.
[3] M. Isard, M. Budiu, Y. Yu, A. Birrell, and D. Fetterly, "Dryad: distributed data-parallel programs from sequential building blocks," *in Proc. 2nd ACM SIGOPS/EuroSys European Conference on Computer Systems*, New York, USA, 2007, pp. 59–72. doi: 10.1145/1272998.1273005.
[4] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," in *Proc. ACM SIGCOMM 2008 Conference on Data Communication*, Seattle, USA, 2008, pp. 63–74. doi: 10.1145/1402958. 1402967.
[5] A. Greenberg, J. R. Hamilton, N. Jain, et al., "VL2: a scalable and flexible data center network," in *Proc. ACM SIGCOMM 2009 Conference on Data Communication*, Barcelona, Spain, 2009, pp. 51– 62. doi: 10.1145/1592568. 1592576.
[6] C. Guo, G. Lu, D. Li, et al., "BCube: a high performance, server-centric network architecture for modular data centers," in *Proc. ACM SIGCOMM*, Barcelona, Spain, 2009, pp. 63–74.
[7] D. Li, C. Guo, H. Wu, et al., "Scalable and cost-effective interconnection of data-center servers using dual server ports," *IEEE/ACM Transactions on Networking*, vol. 19, no. 1, pp. 102–114, Feb. 2011. doi: 10.1109/TNET.2010.2053718.
[8] M. Al - Fares, S. Radhakrishnan, B. Raghavan, N. Huang, and A. Vahdat, "Hedera: Dynamic flow scheduling for data center networks," in *Proc. 7th USENIX Symposium on Networked Systems Design and Implementation*, San Jose, USA, 2010, pp. 1–15.
[9] A. Dixit, P. Prakash, Y. Hu, and R. Kompella, "On the impact of packet spraying in data center networks," in *Proc. IEEE INFOCOM*, Turin, Italy, 2013, pp. 2130–2138. doi: 10.1109/INFCOM.2013.6567015.
[10] J. Cao, R. Xia, P. Yang, et al., "Per-packet load-balanced, low-latency routing for clos - based data center networks," in *Proc. Ninth ACM Conference on Emerging Networking Experiments and Technologies*, Santa Barbara, USA, 2013, pp. 49–60. doi: 10.1145/2535372.2535375.
[11] IETF. (2013, Mar. 2). *IP encapsulation within IP* [Online]. Available: https://datatracker.ietf.org/doc/rfc2003
[12] C. Guo, G. Lu, H. J. Wang, et al., "Secondnet: a data center network virtualization architecture with bandwidth guarantees," in *Proc. 6th International Conference on Emerging Networking Experiments and Technologies*, Philadelphia, USA, 2010. doi: 10.1145/1921168.1921188.
[13] ONF. (2017, Apr. 1). *Open networking foundation* [Online]. Available: https://www.opennetworking.org
[14] A. Curtis, W. Kim, and P. Yalagandula, "Mahout: Low - overhead datacenter traffic management using end - host - based elephant detection," in *Proc. IEEE INFOCOM*, Shanghai, China, 2011, pp. 1629– 1637. doi: 10.1109/INFCOM. 2011.5934956.
[15] C. Raiciu, S. Barre, C. Pluntke, et al., "Improving datacenter performance and robustness with multipath TCP," in *Proc. ACM SIGCOMM*, Toronto, Canada, 2011, pp. 266–277. doi: 10.1145/2018436.2018467.
[16] C. Raiciu, C. Paasch, S. Barr, et al., "How hard can it be? Designing and implementing a deployable multipath TCP," in *USENIX Symposium of Networked Systems Design and Implementation*, San Jose, USA, 2012, pp. 29–29.
[17] D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley, "Design, implementation and evaluation of congestion control for multipath TCP," in *Proc. 8th USENIX Conference on Networked Systems Design and Implementation*, Boston, USA, 2011, pp. 99–112.
[18] M. Alizadeh, A. Greenberg, D. A. Maltz, et al., "Data center TCP (DCTCP)," in *Proc. ACM SIGCOMM*, New York, USA, 2010, pp. 63– 74. doi: 10.1145/1851182.1851192.
[19] R. Niranjan Mysore, A. Pamboris, N. Farrington, et al., "PortLand: a scalable fault - tolerant layer 2 data center network fabric," in *Proc. ACM SIGCOMM*, Barcelona, Spain, 2009, pp. 39–50. doi: 10.1145/1594977.1592575.
[20] *NS-3* [Online]. Available: http://www.nsnam.org

Manuscript received: 2017-03-22

Biographies

LI Dan (tolidan@tsinghua.edu.cn) received the M.E. degree and Ph.D. from Tsinghua University, China in 2005 and 2007 respectively, both in computer science. Before that, he spent four undergraduate years in Beijing Normal University, China and got a B.S. degree in 2003, also in computer science. He joined Microsoft Research Asia in Jan. 2008, where he worked as an associate researcher in Wireless and Networking Group until Feb. 2010. He joined the faculty of Tsinghua University in Mar. 2010, where he is now an associate professor at Computer Science Department. His research interests include Internet architecture and protocol design, data center network, and software defined networking.

LIN Du (lindu1992@foxmail.com) received the B.S. degree from Tsinghua University, China in 2015. Now, he is a master candidate at the Department of Computer Science and Technology, Tsinghua University. His research interests include Internet architecture, data center network, and high-performance network system.

JIANG Changlin (jiangchanglin@csnet1.cs.tsinghua.edu.cn) received the B.S. and M.S. degrees from the Institute of Communication Engineering, PLA University of Science and Technology, China in 2001 and 2004 respectively. Now, he is a Ph.D. candidate at the Department of Computer Science and Technology, Tsinghua University, China. His research interests include Internet architecture, data center network, and network routing.

WANG Lingqiang (wang.lingqiang@zte.com.cn) received the B.S. degree from Department of Industrial Automation, Zhengzhou University, China in 1999. He is a system architect of ZTE Corporation. He focuses on technical planning and pre-research work in IP direction. His research interests include smart pipes, next generation broadband technology, and programmable networks.

DOI: 10.3969/j.issn.1673-5188.2018.02.009
http://kns.cnki.net/kcms/detail/34.1294.TN.20180608.1702.002.html, published online June 8, 2018

Open Source Initiatives for Big Data Governance and Security: A Survey

HU Baiqing, WANG Wenjie, and Chi Harold Liu
(Beijing Institute of Technology, Beijing 100081, China)

Abstract

With the rapid development of Internet technology, the volume of data has increased exponentially. As the large amounts of data are no longer easy to be managed and secured by the owners, big data security and privacy has become a hot issue. One of the most popular research fields for solving the data security and data privacy is within the scope of big data governance and security. In this paper, we introduce the basic concepts of data governance and security. Then, all the state-of-the-art open source frameworks for data governance and security, including Apache Falcon, Apache Atlas, Apache Ranger, Apache Sentry and Kerberos, are detailed and discussed with descriptions of their implementation principles and possible applications.

Keywords

big data; security; governance; open source initiatives

1 Introduction

In recent years, the value of big data has been much recognized by both research community and governmental agencies. However, the rapid development of big data technology has brought more unsolved problems [1]. Data have different values in different spatial-temporal domains as well as in different businesses. In order to maximize its value, using the Internet to share data is inevitable. However, as various enterprises are independent from each other, their data systems and data storage structures are also different. It is thus quite challenging to share the data between them, resulting in a common phenomenon of information islands. Meanwhile, it is challenging to guarantee the data security and privacy when sharing data between different data systems back and forth.

As enterprises start to collect, store, process and exchange large volume of data in the course of addressing these opportunities, they face increasing challenges in the areas of data security, maintaining data privacy, and meeting related compliance obligations. Traditional IT security approaches mainly focus on protecting the organizations' IT infrastructure, by securing the network edge and end points and protecting the data that are stored and moved through the infrastructure. However, the focus of this paper is on how to provide efficient services for managing the entire data lifecycle while protecting its security, which relatively speaking has sparse research exposure from the software development perspective.

In order to solve these problems, big data governance and security has become one of the hottest research areas. Big data governance aims to establish a unified standardized platform, which obtains data from different data sources and can satisfy various data operational requirements as well as conducting lifecycle data management (such as data audit, selection, and migration), to maximize the data value. Moreover, this unified standardized platform can enforce permission settings for different metadata, securing the data for different users on the basis of time points and IP addresses.

Towards this end, this paper aims to introduce how to achieve the governance and security of big data from open source component design and implementation perspectives. Specifically, our contribution is threefold.

1) We extensively review the related studies for big data governance and security, and compare their advantages and disadvantages.
2) We describe five open source initiatives, namely, Apache Atlas, Falcon, Ranger, Sentry, and Kerboros, from both conceptual and architectural perspectives, and discuss their usages in different scenarios.
3) We introduce four future research directions for big data governance and security.

2 Concepts of Big Data Governance and Security

The term "big data" refers to very large or complex data sets that cannot be managed by traditional data processing software. Big data can be converted into very useful knowledge for enterprises to make efficient decisions. It is generally accepted that big data has three "v"s: velocity, variety, and volume [2], and IDC believes big data should also have value besides the three "v"s. Moreover, IBM thinks big data is certain to be of veracity [3].

With the continuous development and popularization of the Internet technology, the data quantity is growing exponentially. Thus the concept of big data is introduced. With the deep inte-

gration of big data and cloud computing technology, the data is no longer easy to be managed by data owners using the traditional technologies. Therefore, big data security and privacy has attracted much attention [4]. At the same time, how to govern the data is also a conundrum.

This paper summarizes all the state-of-the-art technologies for governing big data, from three aspects: principle, scope, and implementation and assessment. Big data governance principle refers to the primary and basic instructive principle that big data follows, which is useful for big data management. In order to efficiently collect, effectively integrate and sufficiently utilize data, big data management principle can be subdivided into the principle of effectiveness, the principle of value, the principle of unity, the principle of openness, and the principle of security. The governance of big data mainly involves five key fields: the lifecycle of big data, the frame of big data, the safety and privacy of big data, the quality of big data and the service innovation of big data. The implementation and evaluation of big data provide an instructive project for enterprises from three aspects: the implementation environment, implementation procedure and assessment of implementation results.

Big data provides a new opportunity for all application areas as well as a challenge for information security. It is of great research value and importance to governments, enterprises, and individuals, thus data security, the precondition of big data development, has become a hot research issue in academic and industrial circles [5]. Big data not only refers to the massive amount of information but also its complexity and sensitivity, which will attract potential aggressors. Furthermore, collected big data includes lots of enterprise operational data, customer data, and individual privacy and detailed records of all kinds of behaviors. The concentrated storage of these data increases the risk of privacy leakage. Meanwhile, the lack of certain definition of data ownership and usage right also increases the risk [6]. Despite the multifaceted advantages of cloud computing, concerns about data leakage or abuse impede its application for security-sensitive tasks. Recent investigations have revealed that the risk of unauthorized data access is one of the biggest concerns for users of big data [7].

Big data imposes challenges and opportunities for auditing. Big data audit is conducted by third-party auditors who are independent of auditing targets; the auditors make comprehensive examination and evaluation of the procedure of big data governance and conduct a series of activities such as putting forward questions and suggestions to the supreme leader of the auditing targets. Big data audit aims to understand the overall situation of an organization's big data activities, to review and evaluate the organization's goal of achieving big data governance, to fully identify and assess the risks associated with the evaluation, to make comments and suggestions for improvement, and to achieve the goal of big data governance. The process of big data audit generally includes setting audit objectives, determining the risk areas of big data audit, setting an audit plan, building the environment of big data audit, carrying out the plan and issuing audit results and governance recommendations [8].

In order to ensure the quality of data governance, its audit mainly focuses on the data supervision and evaluation, being of four aspects: content, architecture, security, and lifecycle. The audit follows certain standards. At present, big data audit methods are mainly divided into traditional audit methods, IT internal audit methods, and big data audit methods. Furthermore, the audit of big data also needs a certain technical means to avoid any blind review and evaluation. The current audit models of provable data possession (PDP) and proof of retrievability (POR) for cloud storage can only be applied to static data audits but fail to support auditing of dynamic data. In order to solve this problem, the third party auditor (TPA) model is proposed, which can efficiently audit the data and also completes the pubic audit for protecting user privacy [9].

3 Related Work

3.1 Big Data Governance

Businesses and enterprises have recognized that increasing expenses on data management solutions are becoming unbearable. They need to use effective data governance methods to solve big data problems. Data governance involves the adoption of data models, data quality standards, data security and lifecycle management methods, as well as the processing procedures the application defines. However, data governance has not been well applied, due to the void of a particular enterprise repository, lacking structures and requiring broader support of organizations. Therefore, despite its importance, data governance is still under investigation.

Al-Ruithe et al. proposed six key dimensions that must be taken into consideration for cloud data governance, such as data governance structure, organizational factors, and technical/environmental factors [10]. A new technology requires a good data governance strategy for its successful implementation. Furthermore, as increasingly large amount of personal and confidential data are transferred to the cloud, related stakeholders' accountabilities have emerged as a critical issue that is related to data protection in cloud ecosystems. From this angle, Felici et al. introduced a conceptual model, consisting of attributes, practices and mechanisms [11], to form the basis for characterizing accountability relationship between cloud actors, and chains of accountability in cloud ecosystems as well. However, these two research efforts did not give specific solutions to the above mentioned problems, no matter from software development or implementation perspective.

3.1.1 Technologies for Big Data Governance

A. Corradi et al. pointed out that the discovery, aggregation and manipulation of distributed and diversified data sets play

an important role in supporting core business processes [12]. It has been agreed that the semantic method can effectively deduce the relation and dependency from the heterogeneous information set, but when the real joint data navigation is performed, the current de facto standard query language is not sufficient and there is no accurate knowledge of data distribution. Therefore, the authors set up a model to propose a lightweight federation ontology for crossing the organization mapping information source to add the current SPARQL limit based on a priori network knowledge. Then, a single query was conducted both on academia and on municipality endpoints, facilitating the development efforts of the overall solution. However, it has certain limitations in terms of endpoint time-outs and unavailability. This process is obviously inefficient and poorly extensible, because web services should be extended anytime while new data sources are added, to query new endpoints and combine results with old ones.

T. Priebe et al. presented a methodology to gather and structure data requirements to improve data-intensive projects and enable data governance [13]. The methodology facilitates data harmonization by introducing a semantic business information model as a central point of reference on top of physical and logical data models. In addition, M. Al-Ruithe et al. postulate that as the "smart" continuum continues to innovate and grow, so will the velocity and volume of data [14]. Their efforts add to body of research and frameworks are proposed to identify a roadmap to address data governance and security challenges in Internet-of-Things (IoT) cloud converged environments. Also, they propose a data governance and security layer, which describes roles, responsibilities and policies as key pillars. However, as more IoT cloud converged domains continue to evolve, their roles, responsibilities and policies will remain central to governance and security processes and procedures, that brings certain questions of whether this framework can continue to function or not. Furthermore, software development and implementation details are still missing.

When data moving across multiple systems, it may cause more mistakes or bad changes of related processes and systems. The lack of awareness of corporate data landscape impacts the ability to govern data, which in turn impacts overall data quality within organizations. R. J. DeStefano et al. propose tools and techniques for companies to better gain awareness of the landscape of their data, processes, and organizational attributes through the use of linked data, via the Resource Description Framework (RDF) and ontology [15]. The outcome of adopting such techniques is an increased level of data awareness within the organization, resulting in improved ability to govern corporate data assets, and in turn increased data quality. However, the application of such techniques into real-life big data systems still needs time.

3.1.2 Applications of Big Data Governance

It is all agreed that data has become a major need in nearly all businesses. However they must be accurate and valid. Organizations usually face data problems, like data duplication, inaccurate data, incomplete data, invalid data, and unavailable data. Yulfitri et al. [16] analyzed these problems that occurred in a governmental agency in Jakarta, Indonesia, and concluded an approach that was used in sequential stages, that is studying best practices regarding operational model of data governance, analyzing current conditions, reviewing organizational structure, analyzing business processes and human resources, and mapping out data governance activity. Another example is about operational model for clinical data governance. Thiel et al. [17] proposed a method to identify the legal and ethical challenges in Europe for clinical data governance in health informatics and to classify the various legal bases for sharing a dataset.

3.2 Big Data Security

Data security is one of the major challenges in the era of big data, including the protection against security breaches and data leakage, penetrability in public databases, and third party data sharing, etc. From research only perspective, there exists certain literature reviews reporting the recent progresses along this direction. However, from software development and implementation perspectives, open source initiatives are missing. For example, how to protect sensitive information from the security threats brought by data mining has become a hot topic in recent years. To solve the above threats, Xu et al. surveyed the privacy issues related to data mining by using a user-role based methodology [18]. They differentiate four different user roles that are commonly involved in data mining applications, i.e., data provider, data collector, data miner and decision maker. Furthermore, Ye et al. proposed three categories about security issues in big data infrastructure security, data privacy, and data management [19]. Finally, Tan et al. presented a survey of recent security advances in smart grid, centered around the security vulnerabilities and solutions within the entire lifecycle of smart grid data [20].

4 Open Source Initiatives for Big Data Governance

In the first two sections, we presented a technical overview of the governance and security of big data, including big data lifecycle governance, security protection, and data auditing. In the following sections, we will detail the realization of governance and security of big data. This section focuses on Apache Falcon and Apache Atlas, which play an important role in big data governance. Apache Falcon can perform data lifecycle management, including data collection, data processing, data backup and data cleansing, for big data platforms, as well as for fine scheduling of components of big data platforms. Apache Atlas can perform tasks including metadata management, data lifecycle auditing and visualization, lineage search,

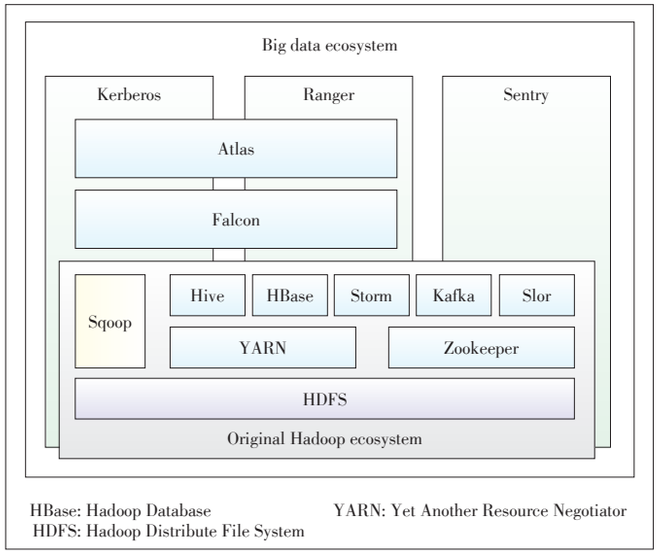
and data security and privacy, for big data platforms.

Fig. 1 shows the overall big data ecosystem that supports the data governance and security functionalities, based on the existing Hadoop ecosystem. Apache Falcon, Atlas, Kerberos, Ranger and Sentry are used. Falcon and Atlas components can interact with each other, while Atlas can be used as metadata sources for Atlas. Meanwhile, Hive, Sqoop, Falcon, and Storm can also be used as metadata sources. Then, Apache Ranger is used as a centralized security management solution for Hadoop that enables administrators to secure authentication mechanisms and configurations with Hadoop components such as Hadoop Distributed File System (HDFS), Hive, Hadoop Database (HBase), and Kafka. The components Kerberos, Ranger, and Sentry provide security protection to all Hadoop components. Furthermore, Kerberos and Ranger can interact with Falcon and Atlas, to provide data governance and security solutions simultaneously.

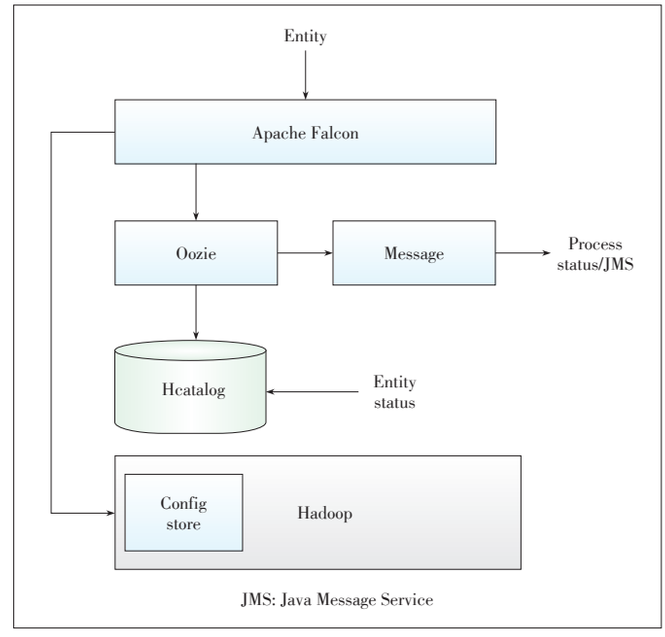
4.1 Apache Falcon

Apache Falcon solves the problems of Hadoop data replication, business continuity and lineage tracking by declaring data management and processing solutions. Falcon centrally manages the data lifecycle, facilitates quick data replication for business continuity and disaster recovery and provides a foundation for audit and compliance by tracking entity lineage and collection of audit logs. It also helps user set data management and the way of process and submit it for scheduling on Hadoop Cluster.

Apache Falcon is a management platform built on Hadoop data set and **Fig. 2** shows its processing flows. A user can submit an entity to the Apache Falcon through the Falcon client or the Rest API. Falcon generates the workflow entity based on the declaration information and stores it in the config store of



▲**Figure 1. Big data ecosystem that supports data governance and security.**



▲**Figure 2. Falcon architecture diagram [21].**

the Hadoop. As processing the workflow, Apache Falcon performs task scheduling mainly through Oozie and stores the entity processing status in HCatalog. During scheduled tasks, Oozie will return status information during execution as well as execute command messages and send them back to the Java Message Service (JMS) message announcement and return the results to Apache Falcon. Falcon essentially translates the user's data set and its process configuration into a series of repetitive activities through a standard workflow engine, without doing anything cumbersome. All functions and workflow state management requirements are entrusted to the workflow scheduler for scheduling. Because it does not do extra work on the workflow itself, the only thing Falcon has to do is to keep the dependencies and links between the data flow entities. This allows the developer to completely feel the Oozie scheduler and other underlying components when creating a workflow using Flacon so that they can focus on the data and processing itself without any unnecessary operation.

Although Falcon distributes the workflow to the scheduler (the default scheduler is Oozie; due to Oozie's limitations, Flacon also performs the scheduler functionality), Falcon also maintains communication (for example, JMS messages) with the scheduler to generate message traces for each workflow in the execution path, to ensure the progress of the current workflow task and the specific situation.

Falcon simplifies the development and management of data processing pipelines with a higher layer of abstraction, taking the complex coding out of data processing applications by providing out-of-the-box data management services. This simplifies the configuration and orchestration of data motion, disaster recovery and data retention workflows.

Falcon enables this simplified management by providing a framework to define, deploy, and manage data pipelines. As an open source project of data lifecycle management, Apache Falcon can provide the following services:

- Establishing relationship between various data and processing elements on a Hadoop environment
- Feeding management services such as feed retention, replications across clusters, and archival
- Onboarding new workflows/pipelines easily, with support for late data handling and retry policies
- Integrating with metastore/catalog such as Hive/HCatalog
- Providing notification to end customer based on availability of feed groups
- Enabling use cases for local processing in colo and global aggregations
- Getting lineage for feeding and processing.

In general, Apache Falcon meets enterprise data governance needs in three areas, as shown in **Table 1**.

In the workflow implementation, Apache Oozie is mainly responsible for task scheduling, and the entity execution status is stored in HCatalog. During the scheduled execution of the task, Oozie returns the status information during execution, executes the command message and returns the result to the Apache Falcon by sending it to JMS message announcement.

The default scheduler for Apache Falcon is Oozie. Since Falcon relies on Oozie for scheduling and workflow execution, which limits the natural return of feed. In order to achieve better scheduling capabilities, the current Apache Falcon project has also started with native scheduler development.

The scheduler functions include:

1) Submitting and scheduling Falcon to run the process regularly (no data dependencies are required). The program can be a PIG script, an Oozie workflow, or a Hive.

2) Monitor/query/modify scheduled processes: All used entity APIs and instance APIs remain in their original state. Falcon provides a data management function in the form of a life dataset that allows a user to submit a dataset location as a time-based partitioned directory in an HDFS included file.

Although the actual responsibility of the workflow is with the scheduler (such as Oozie), Apache Falcon still remains the execution path of the workflow by subscribing to messages that may be generated by each workflow. When Apache Falcon generates a workflow in Oozie, after that, it uses additional steps, including JMS messaging, to detect workflow execution. The Apache Falcon system itself subscribes to these control messages and, if necessary, performs operations such as retrying and handling the latest input data.

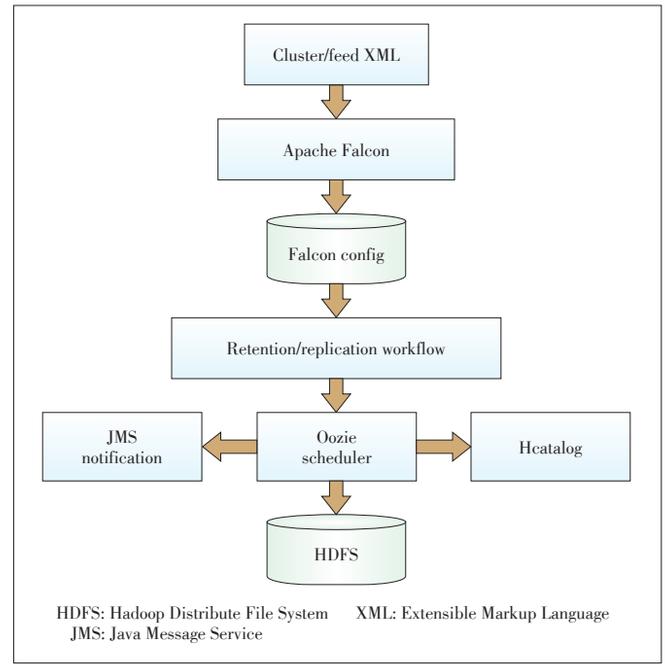
As shown in **Fig. 3**, the user submits and declares the cluster configuration information and data set information to the Apache Falcon through the Cluster XML cluster declaration file and feeds XML dataset declaration file. Falcon generates the cluster entity and feed data based on two files, sets the entities, and then stores them according to the Falcon configuration store, and finally generates the relational graphs. When a reservation or backup is required to operate related data set, Falcon reads the execution entity information in the configuration store and generates the corresponding workflow that will be dispatched by the Oozie scheduler. Oozie outputs the scheduling results to HDFS or Hive's Catalog Service and generates JMS message announcements for each action.

Therefore, Apache Falcon plays a role like Oozie's more advanced abstraction layer. It is the hub of the drives, such as Hive, Sqoop, Map Reduce and other series of Hadoop component tasks, which is not directly responsible for data processing in the actual data processing.

▼**Table 1. Apache Falcon requirements and features [22]**

Need	Feature
Unified management of data lifecycle	• Centralized definition and management of pipelines for data ingest, process and export
	• Ensuring disaster preparedness and business continuity
	• Out-of-the-box policies for data replication and retention
	• End-to-end monitoring of data pipes
Compliance and audit	• Visualization of data pipeline lineage
	• Tracking the data pipeline audit log
	• Taging data with business metadata
Database replication and archival	• Replication across on-premise and cloud-based storages targets: Microsoft Azure and Amazon S3
	• Data lineage with supporting documentation and examples
	• HDFS in heterogeneous tiered storage
	• Definition of data hot/cold storage layer within a cluster

HDFS: Hadoop Distribute File System



▲**Figure 3. The data set workflow.**

At present, Apache Falcon has been successful in a number of areas, including advertisement, healthcare, mobile communications applications, etc. For example, InMobi is one of the largest users of Falcon; it services the advertisement industry that has more than 200 complex big data pipelines and different data sources, and the data is still growing. InMobi can quickly deal with massive amount of data with the help of Apache Falcon, and keep up with the market ever-changing rhythm.

In addition, Expedia.com also carries out data management with Falcon. Expedia introduces the Falcon platform to integrate various data and rules for data processing in the Hadoop environment, sets the relationship of data and rules, perfectly solving the problems caused by the fast development of business. Falcon also provides a great deal of help for Expedia in terms of security. It provides security at the transport level to ensure data confidentiality and integrity.

4.2 Apache Atlas

Apache Atlas is also an important component for the management of big data. It supports metadata management, data lifecycle audit and visual display, data lineage collection, data security and privacy, and other content for big data. Apache Atlas play a very important role in big data governance.

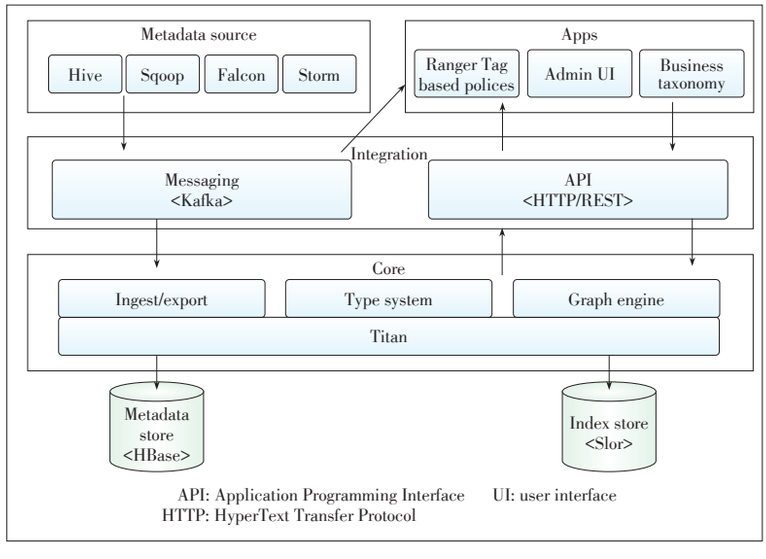
Apache Atlas is a scalable and extensible metadata management tool and provides core foundational governance services, including exchanging metadata with other components, changing the way of past metadata management, building a unified metadata definition standards, and integrating various components of the Hadoop ecosystem to establish a unified, highly scalable metadata management platform [23].

Metadata management can provide complete data definition information for data users, reduce data redundancy, help identify and search data, track data changes in the database, help users understand the data throughout the lifecycle, achieve a simple and efficient management of big data systems in the massive data, and find the value of data through the effective tracking of data resources. Atlas can efficiently integrate all ecosystem components of the enterprise platform with pre-defined requirements while enabling data visualization in Hadoop with pre-set models, providing easy-to-use functions data Audit, and enriching the business metadata by lineage search.

In the process of big data governance, data management tracks the entire lifecycle of data, including data sources, data modification and deletion, and the ability to quickly retrieve. The metadata model can better understand the data and lifecycle by combining labels and data attributes, which enables rapid modeling of data. Unified metadata standards provide a basis for establishing a unified meta-database that runs through the Hadoop ecosystem.

Apache Atlas provides five services: metadata exchange, data lineage, data lifecycle visualization, fast data modeling, and rich API in big data governance. Atlas also has the characteristics of data classification, centralized audit, search and data lineage, security and strategy engine, which plays an important role in the management of big data.

Fig. 4 shows the framework of Atlas. Its components can be grouped into four major categories: the Atlas core, integrations, applications (Apps), and metadata sources. Atlas supports ingesting and managing metadata from the following metadata sources: Hive, Sqoop, Falcon and Storm. After the metadata are acquired, both the API and message system can be used. In terms of metadata management, Atlas can be exposed to user through REST API, so that the user can perform corresponding operations. Meanwhile, the user can choose to integrate with Atlas using a messaging interface that is based on Kafka. Metadata managed by Atlas is used by various applications (Apps) to satisfy many governance use cases, including Admin User Interface (Admin UI), Ranger Tag Based Polices and Business Taxonomy. Atlas Admin UI is a web based application that allows data stewards and scientists to discover and annotate metadata. The Admin UI uses the REST API of Atlas for building its functionality. Tag Based Policies are used to integrate Ranger and Atlas. Ranger is notified by Atlas when metadata change. Meanwhile Atlas provides a business class taxonomy interface that allows the user to build a hierarchical set of terms for various terms in the business domain and integrate them into metadata entities that can be managed by Atlas. The core part of Atlas includes data import and export, type system, graph engine, and Titan. Atlas uses a graph model to represent metadata objects and then stores metadata objects through the Titan graph database. The Titan graph database uses metadata and index databases for data storage, and the metadata uses HBase and the index database uses Solr. Atlas de-



▲Figure 4. Architecture of Apache Atlas [24].

fines an original metadata model to represent various objects, providing the corresponding modules from these components to the metadata object. There are various metadata stored in the Atlas meta-database, and these metadata will be used by a wide variety of applications to meet the needs of a variety of display services and big data governance. Atlas can also be integrated with Apache Ranger, which allows administrators to customize the metadata-based security-driven policies for efficient management of big data.

Although Apache Atlas is still an Apache incubation project, it has been used in a production environment. Apache Atlas can efficiently integrate with all ecosystem components of the enterprise platform while meeting the enterprise's default requirements for the Hadoop ecosystem. At the same time, Atlas can use the pre-set model to visualize data in Hadoop, provide easy-to-use data auditing, and to enrich the metadata of enterprise's business through data collection. It also allows any metadata consumers to collaborate with each other without having to build a separate interface between them. In addition, the accuracy and security of metadata in Atlas is guaranteed by Apache Ranger, which prevents data access requests that do not have permissions at runtime.

4.3 New Progress of Falcon and Atlas Open Source Communities

With new feature requirements flowing in constantly, the Falcon project is making more frequent releases to ensure the features become available to the users as soon as possible. The latest in this string of releases is Falcon 0.10 that was announced on August 8, 2016, however the current stable version is still 0.9. There are many new features that the community is currently working on for product improvements in version 0.9, some of which are: 1) native time-based scheduling, 2) ability to import from and export to a database, and 3) additional API support in the Falcon unit.

Falcon huses Oozie as its scheduling engine. However, while Oozie works reasonably well, there are scenarios where Oozie scheduling is proved to be a limiting factor in version 0.9, e.g., simple periodic scheduling with no gating conditions, calendar based time triggers, scheduling based on data availability for periodic datasets, etc. To overcome these limitations, a native scheduler will be built and released over the next few releases of Falcon. In the 0.9 release, only time-based scheduling without data dependency is supported.

At present, the latest version of Atlas is 1.0 which was announced on May 2018. There are many new features in Atlas 1.0, some of which are: 1) new DSL implementation, using ANTLR instead of Scala; 2) removal of older type system implementation in atlas-type system library; 3) using fine grained authorization to add metadata security; 4) classification propagation via entity relationships; 5) adding HBase integration. Looking for the future, as the next version, Atlas has the aggressive design plans to support, e.g. Titan 1.0+, Spark Integration, NiFi Integration, etc.

5 Open Source Initiatives for Big Data Security

As the Hadoop ecosystem is becoming more and more mature, it is now able to support a complete data lake. A data lake is a storage repository that holds a vast amount of raw data in its native format until it is needed [25]. Enterprises can run multiple workloads in a multi-client environment on a Hadoop system. Enterprises need to support multi-user access to the data lake and data is an important asset for enterprises, so how to protect these different types of user data need to be solved. The solutions are big data distributed security frameworks, including Ranger, Sentry, and Kerberos.

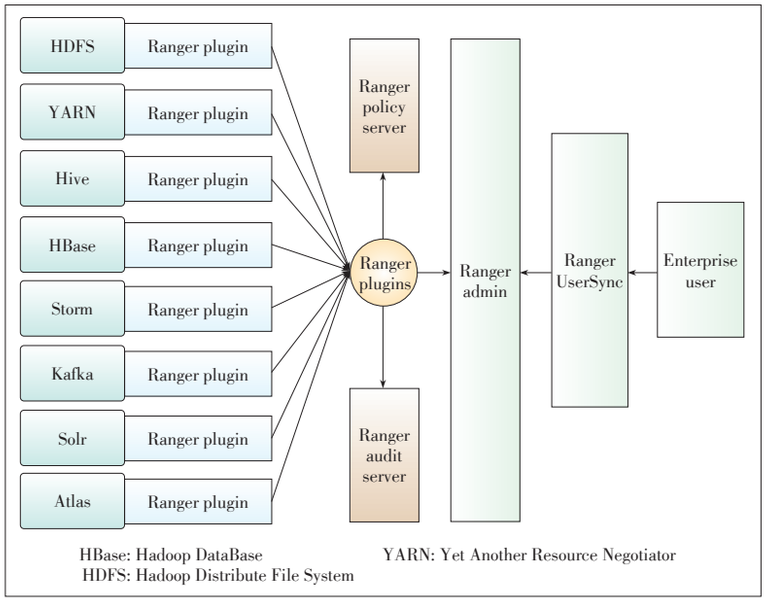
5.1 Apache Ranger

Apache Ranger is a centralized security management framework that provides centrally managed security policies and monitors user access. It supports fine grained authorization and auditing for Hadoop ecosystem components such as Hive and HBase. By operating the Ranger Web UI console, administrators can easily control the user's access rights by configuring policies. Compared to Apache Sentry, Ranger supports more services, including most of the Hadoop components like HDFS, HBase, Hive, Yarn, Strom, Kafka, Knox, Atlas, and Solr. Therefore, when a user needs to manage these frameworks, or when the entire big data ecosystem contains these frameworks, the user can use Ranger to control these frameworks' security.

As shown in **Fig. 5**, the Ranger architecture consists of three parts: Ranger Admin, Ranger Usersync and Ranger Plugins [26]. Ranger Admin is the core interface for security administration and is the center of the Ranger framework. Users can manage users' system rights on the Web UI provided by this service, and can create and update the authentication policies, which are stored in a policy database. Each component's plugin periodically monitors these policies. Ranger Admin also provides an audit service that collects data stored in HDFS or relational databases for auditing.

Ranger Plugins are the core of rights security management and is a lightweight Java program that can be embedded in each cluster component. For example, Ranger, for its highly supported Hive, provides a plugin that can be embedded in the Hiveserver2 service, which extracts the rights authentication policies for all Hives from the ranger admin service and stores them in a local file. When a user request comes through the component, the plugin intercepts the request and evaluates whether it meets the security policy. At the same time, the plugin can also collect data from user requests and create a separate thread to send the data back to the audit server.

Ranger Usersync is a very important tool for synchronizing users/groups from UNIX systems or Lightweight Directory Ac-



▲Figure 5. Ranger architecture.

cess Protocol (LDAP) to Ranger Admin. This stand-alone process can also be used as an authentication server to log into Ranger Admin by using a Linux user/password. The user or group information is stored in Ranger Admin for policy definition. Moreover, users can manually add/delete/modify user or group information to set permissions on these users or groups.

From the perspective of the rights model, Ranger controls the component's rights through centralized access controlling. Moreover, authorization is defined by "User – Resources – Permissions" between the three relations. Ranger abstracts this relationship and then extends users' own authority models. The "User–Resources–Permissions" has the following definitions:

1) User and group: User represents a user who is accessing the resource, while group refers to the one the user belongs to.
2) Resources: Using the tuple (Service, Resource), a policy only corresponds to a service, but can correspond to multiple resources.
3) Permissions: Expressed by the tuple (AllowACL, DenyACL), both of which contain two sets of AccessItem. AccessItem describes the relationship between a set of users and a set of accesses. AllowACL indicates that permission is allowed, and denyACL denies the permission.

Table 2 lists the model entity enumeration values for several common components.

In the recently released Ranger version, Apache Ranger has added Atlas components, integrating Atlas to support classification-based (tagging) and other dynamic policies (based on location, prohibition, and data lifecycle). This is the first time that Apache Ranger for security and Apache Atlas for data governance are integrated to authorize customers to define and implement security policies based on dynamic classification. Ranger's centralized platform enables data administrators to define security policies and apply the strategy to the entire hierarchy of data assets, including databases, tables, and columns, based on Atlas metadata tags or attributes. Ranger today has important features, but there are still some questions about how to adapt to the larger Hadoop security ecosystem. For example, some Ranger targets overlap with the targets of Apache Sentry (see the next section for details), and there seems to be little consensus about how the project synchronizes its work.

5.2 Apache Sentry

Apache Sentry, similar to Apache Ranger, performs fine-grained access control on the Hadoop ecosystem components, such as Hive and Impala. It also provides control and implementation of data for authenticated users and applications on the permission control function of Hadoop clusters. In the existing group mapping environment of the Hadoop ecosystem, it is easy to manage permissions by simply manipulating the unique role of Sentry.

Sentry mainly consists of three components: the server, data engine and plugin. The Sentry server manages the policy metadata, which supports the interface for safe retrieval and manipulation of metadata. The data engine is a data handler that requires authorization to access data or metadata, such as Hive and Impala. The data engine loads the Sentry plugin and intercepts all client requests that access the resource. Moreover, the requests are validated by the Sentry plugin. The Sentry Plugin runs in the data engine. It provides an interface to manipulate the authorization metadata stored in the Sentry server and includes an authorization policy engine that evaluates the access request by using the authorization metadata retrieved from the server.

In fact, the primary purpose of the Sentry server is to facilitate the management of metadata, and real authorization decisions are made by the policy engine in the Sentry plugin. The Sentry architecture (**Fig. 6**) has three important layers:

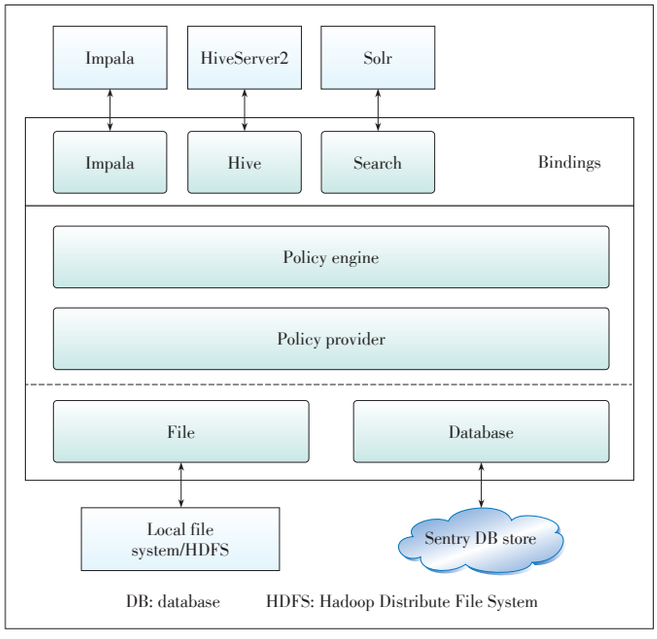
1) Bindings Layer

As mentioned earlier, Sentry's policy engine is part of the Sentry plugin, called by the Impala, Hive and Search components. The bindings layer is the bridge between the Sentry authorization and the invoking tools like Impala, HiveServer2,

▼Table 2. Ranger model entity enumeration values

Service	Resource	Authority
HDFS	Path	Read; Write; Execute
YARN	Queue	Submit; Admin
HBase	Table; Column Family; Column	Read; Write; Create; Admin
Hive	Database; Table; Column	Select; Update; Create; Drop; Alter; Index; Lock

HBase: Hadoop DataBase
HDFS: Hadoop Distribute File System
YARN: Yet Another Resource Negotiator



▲Figure 6. The architecture of Apache Sentry [27].

and Solr, which is responsible for converting the native format of authorization request to the request that can be processed by the Sentry policy engine.

2) Policy Engine

This is the heart of the Sentry, which obtains the privilege from the bindings layer and obtains the required privileges from the policy provider layer. It compares the requested and required permissions and determines whether the operation should be allowed.

3) Policy Provider

The policy provider is an abstraction that makes the authorization metadata available to the policy engine. It allows to use metadata regardless of how metadata is stored. Currently, Sentry supports file-based storage and relational database storage. A file-based scenario is to store metadata in a file format. The file can be stored in the local file system or HDFS. The file contains the group, role and privilege between the two groups of maps. However, it is difficult to use the program to modify the file, because there is competition for resources, and it is not conducive to maintenance. At the same time, Hive and Impala need to provide industry-standard SQL interface to manage the authorization strategy, requiring the use of programming the management.

Apache Sentry and Ranger are very similar in many functions, such as support for fine-grained access control, secure authorization mechanism, and support for multiple Hadoop components. The main difference is that Ranger was originally developed by Hortonworks, while Sentry was originally developed by Cloudera. However, both of them now belong to the Apache Foundation's incubation program. In contrast, Ranger is more comprehensive, which may be better for the industry. However, Cloudera has previously announced a major plan for security and Sentry is one of the beneficiaries of this "one platform" strategy. The prospects of Sentry are immeasurable.

5.3 Kerberos

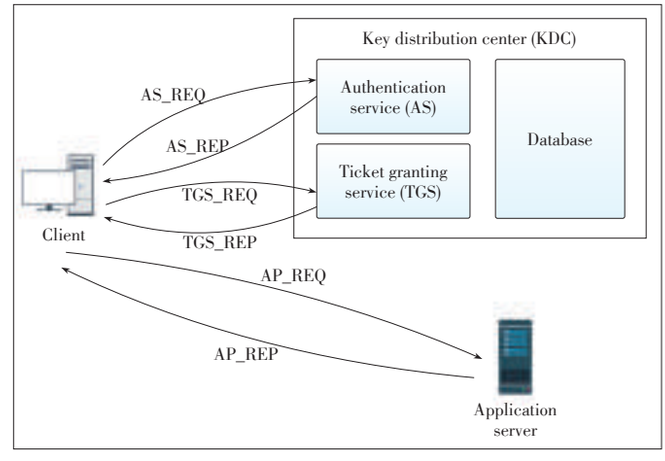
Kerberos is a computer network authorization protocol used to authenticate personal communications over non-secure network environments [28]. The implementation of the Kerberos authentication process does not depend on the authentication of the host operating system. It requires neither trust based on the host address nor the physical security of all the hosts on the network. Kerberos is based on the assumption that the data packets can be arbitrarily read, modified and inserted on the network. Kerberos supports the integration of Hadoop's multiple components, including HDFS, Yarn, Hive, Zookeeper, HBase, Sqoop, Hue, Spark, Solr, Kafka, Storm, Impala, etc.

The user uses the principal to authenticate through the Kerberos client. After the authentication is successful, the server will return the authenticated ticket to the user, who will use it for secure communication.

Kerberos supports Windows, Linux and Mac OS systems. It is often used in such systems as Web applications and enterprise networks, which require high security. Many companies such as Microsoft, Apple, and Red Hat have used Kerberos products; Kerberos also plays a very important role in the X-Box and cable television industry. Therefore, it can be said that Kerberos is one of the most widely used authentication methods in the history of computer networks.

Kerberos consists of the key distribution center (KDC) and the client and application server (**Fig. 7**). KDC provides the authentication service (AS) and ticket granting service (TGS). The specific process of certification is as follows:

1) A client sends an authentication request AS_REQ to AS. AS returns AS_REP, which includes a session key SK_TGS that is generated by the user and TGS and sends the ticket granting ticket(TGT) and SK_TGS encrypted with the user key.



▲Figure 7. The architecture of Kerberos.

2) The client sends TGS_REQ to TGS as requesting a service ticket (ST) for accessing an application server, and then sends TGT and the authenticator. The authenticator is used to verify that the user who sent the request is the user declared in the TGT.
3) If TGS judgment is correct, a new session key SK_Service will be generated for the user and the application server and then TGS_REP be sent to the user, including SK_Service and ST.
4) The user uses the session key SK_TGS to unlock the packet and get the session key SK_Service. SK_Service is then used to generate an authenticator andST and the authenticator are sent to the application server.
5) Authenticator is encrypted using the session key (SK_Service) between the user and the application server. The application server receives the key decryption ST or the session key SK_Service, and then use the session key (SK_Service) to decrypt the authenticator to verify that the user who sent the request is the user declared in the ticket.
6) The application server sends a packet to the user to prove his identity, which is encrypted using the session key (SK_Service). The client waits for the application server to send a confirmation message. Ifthe application server is not correct, it cannot unlock the ST, nor get the session key, so as to avoid the use of a wrong server. After that, the user and the application server can use SK_Service to communicate, and in the TGT validity period, the user will skip the first step of the authentication and directly jump to the second step by using TGT to prove their identity.

Although Kerberos is a high-performance security encryption system, it also has some problems if used improperly or its management is neglected. For example, if we observe the Kerberos authentication process, we can find that the Kerberos service is almost entirely dependent on the services on the KDC. Once the host of the KDC is down, all Kerberos-enabled services are not available.

5.4 Comparison of Security Frameworks

In this section, we compare the functionalities supported by Apache Ranger, Apache Sentry, and Kerberos frameworks.

As shown in **Table 3**, Ranger and Sentry have quite similar functionalities, both to provide audit log services, fine-grained authorization, unified authorization management strategy, and role-based management. They cannot support module authentication, nor the ticket grant services. However, Ranger supports more big data open source components than Sentry. Sentry only supported Apache HDFS, Apache Kafka, Apache Solr, Apache Sqoop, and Cloudera Impala by December 2016. As a comparison, Ranger also supports other components, i.e., HBase, Solr, Storm, and Atlas. Compared with Ranger and Sentry, Kerberos is mainly used to authenticate the above-mentioned components, but it does not support fine-grained permission control. Kerberos provides the ticket authorization service and the component authentication as well. It basically supports all big data open source components, as long as these components can use Kerberos authority for identification. To a certain extent, it is worth noting that Apache Ranger and Apache Sentry's functionalities overlap, and the users can choose them based on her own experience. However, Ranger supports more components than Sentry, and thus it may be a better choice in production environments.

▼Table 3. Comparisons between the three security frameworks

Services and Features	Ranger	Sentry	Kerberos
Audit log service	√	√	√
Fine grained authorization service	√	√	
Authentication service			√
Unified authorization policy	√	√	
Ticket granting service			√
Role-based management	√	√	
Supported components	9	5	12

5.5 New Progress of Ranger, Sentry and Kerboros

Ranger 0.6.0 was released in August 2016. It removes the support of database-based auditing, uses Solr as the index audit data, and HDFS to store audit data. The purpose of using HDFS is that the Ranger plugin can expand the audit log and index, when a new Ranger plugin is incorporated.

In early 2016, Sentry successfully graduated from the Apache incubation with six releases and continued to grow to provide unified authorization policy management across different Hadoop components. It is now targeting significant enhancements across the areas of 1) ease of Sentry enablement and management of permissions, 2) feature parity with access control capabilities of mature relational database systems, 3) attribute-based access control (ABAC), including permissions based on data sensitivity tags, and 4) integration with additional Hadoop ecosystem frameworks, so that existing permissions can be enforced across additional access paths [29].

Finally, Kerberos is designed to provide strong authentication for client/server applications by using secret-key cryptography. The availability of krb5-1.15.1 has been recently released. Now, the detached PGP signature is available without going through the download page, if one wishes to verify the authenticity of a distribution you have obtained elsewhere.

6 Future Work

Based on the above descriptions, the entire big data ecosystem can be somehow secured when we use Apache Falcon and Apache Atlas on the Hadoop ecosystem to govern the big data, use Apache Ranger and Apache Sentry on the application for security authentication and rights management, and use Kerberos to secure network transmission. However, there are still

some open problems and challenges in the process of big data governance and security.

6.1 Data Privacy Protection and Security

The existing data security protection methods are not effective enough to solve the multi-dimensional security of big data. The security frameworks described in this paper address neither the security of data semantics for access control, nor the access authorities for data owners. Moreover, conventional security scanning methods take too much time to process massive amount of data. Furthermore, current user data collection, storage, management and use are not standardized, and lack of supervisions, which mainly relies on self-disciplines of enterprises. Furthermore, end users cannot determine their own use of privacy information. Users have the right to decide how their information is used to achieve certain level of controllable privacy protections.

6.2 Secured Data Storage of Both Relation and Non-Relational Data

The data scale can easily reach the size of PB level, and therefore, massive data storage system should also have the appropriate level of scalability. The Internet has enforced the data to develop toward heterogeneous, unstructured, and other heterogeneous data such as images, video, audio, and text, growing at an alarming rate every day. The increasing heterogeneous data make the secured storage a challenging problem. That is, traditional Relational Database Management System (RDBMS) and emerging NoSQL databases have different security protection methods, at different scales, and of different applicability. However, certain degree of transparency is highly expected for companies because they do not care about which security protection method is used and how to store/retrieve the data. This requirement demands a unified middleware that supports secured data storage of both relational and non-relational databases.

6.3 Credibility of Big Data

There can be a general view of the big data that the data itself can explain everything, and the data itself is the fact. However, the reality is that the data will also be deceived. This is a threat to the security of big data. One of the threats to the credibility of big data is forgery or deliberate manufacturing data, while erroneous data often lead to erroneous conclusions. If a data application scenario is clear, someone may deliberately create data, create a false impression, and induce analysts to come to the conclusion that is beneficial to them. However, false information is often hidden inside the data, which prohibits the accurate identification of its authenticity and thus makes false judgments. Furthermore, the emergence and fast spread of false information through online social networks significantly increases the difficulties of identification, which cannot be solved by current security techniques. Another threat to the credibility of big data is that data may be distorted in the process of transmission. This is because a data acquisition process usually involves human intervention that may introduce errors, data distortion and deviation, and ultimately affect the accuracy of data analysis results. Therefore, it is highly expected that standardized transmission specification is enforced.

6.4 Optimizing Big Data Access Control

Access control is an effective way for data-controlled sharing, since data may be used for a variety of different scenarios. The key challenge comes from how to preset the roles, or to divide the different roles before the data system runs. Due to the wide range of big data applications, data are usually accessed by different organizations or individuals, with different purposes. However, their specific authorization requirements and access control rights are usually unknown as a priori.

7 Conclusions

This paper first gives a definition of data governance and security, and proposes that the management of big data can be carried out from three aspects: governance principle, governance scope, and implementation and evaluation of governance. The audit of big data governance mainly focuses on the supervision and evaluation of big data management in five aspects: the audit of big data governance, the audit of big data management content, the audit of big data management, the big data security audit, and the big data lifecycle audit. In this paper, we introduce the data lifecycle management framework (i.e., Apache Falcon), life cycle management and metadata management (i.e., Atlas), and three security authentication frameworks (i.e., Ranger, Sentry, and Kerberos). Detailed analysis of all these frameworks have been made. Moreover, we discuss how these frameworks carry out the data Lifecycle management, and protection of data security and privacy in the Hadoop ecosystem. Finally, we suggest the future works of the data governance and security and conclude this paper.

References

[1] Z. J. Dong, "Improving performance of cloud computing and big data technologies and applications," *ZTE Communications*, vol. 12, no. 4, pp. 1–2, Dec. 2014.
[2] L. Douglas. (2001, Feb. 6). *3D data management: controlling data volume, velocity and variety* [Online]. Available: https://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf
[3] S. Iwata, "Big data era," *Journal of Information Processing and Management*, vol. 55, no. 8, pp. 543–551, Jan 2012. doi:10.1241/johokanri.55.543.
[4] K. Sarvakar, "big data security and privacy using data transformation with role based access control," *International Journal of Computer Science & Communication (IJCSC)*, vol. 7, no. 2, pp. 90–94, Jul. 2016. doi: 10.090592/IJCSC.2016.115.
[5] S. B. Scruggs, K. Watson, and A. I. Su, "Harnessing the heart of big data," *Circulation Research*, vol. 116, no. 7, pp. 1115–1119, Mar. 2015. doi: 10.1161/CIRCRESAHA.115.306013.

[6] M. Jensen, "Challenges of privacy protection in big data analytics," in *IEEE BigData Congress*, Santa Clara, USA, Jul. 2013, pp. 235–238. doi: 10.1109/BigData.Congress.2013.39.
[7] F. Cang, M. Zhang, and Y. Wu, "Preventing data leakage in a cloud environment," *ZTE Communications*, vol. 11, no. 4, pp. 27–31, Dec. 2013. doi: 10.3969/j. issn.1673-5188.2013.04.004.
[8] K. Setty and R. Bakhshi, "What is big data and what does it have to do with it audit?," *ISACA Journal*, vol. 3, no. 14, pp. 1–3, 2013.
[9] M. Anup, R. Nimje, V. T. Gaikwad, and H. N. Datir, "A review of various trust management models for cloud computing storage systems," *International Journal of Engineering and Computer Science*, vol. 3, no. 2, pp. 3924–3928, Feb. 2014.
[10] M. Al-Ruithe, E. Benkhelifa, and K. Hameed, "Key dimensions for cloud data governance," in *IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud)*, Vienna, Austria, Sept. 2016, pp. 379–386. doi: 10.1109/FiCloud.2016.60.
[11] M. Felici, T. Koulouris, and S. Pearson, "Accountability for data governance in cloud ecosystems," in *IEEE 5th International Conference on Cloud Computing Technology and Science*, Bristol, UK, Dec. 2013, pp. 327–332. doi: 0.1109/CloudCom.2013.157.
[12] A. Corradi, L. Foschini, A. Zanni, et al., "A federation model to support semantic SPARQL queries for enterprise data governance," in *Eleventh International Conference on Digital Information Management (ICDIM)*, Porto, Portugal, Sept. 2016, pp. 96–100. doi: 10.1109/ICDIM.2016.7829778.
[13] T. Priebe and S. Markus, "Business information modeling: a methodology for data-intensive projects, data science and big data governance," in *IEEE International Conference on Big Data*, Santa Clara, USA, Dec. 2015, pp. 2056–2065. doi: 10.1109/BigData.2015.7363987.
[14] M. Al-Ruithe, S. Mthunzi, and E. Benkhelifa, "Data governance for security in IoT & cloud converged environments," in *IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, Agadir, Morocco, Dec. 2016, pp. 1–8. doi: 10.1109/AICCSA.2016.7945737.
[15] R. J. DeStefano, L. Tao, and K. Gai, "Improving data governance in large organizations through ontology and linked data," in *IEEE 3rd International Conference on Cyber Security and Cloud Computing (CSCloud)*, Beijing, China, Jun. 2016, pp. 279–284. doi: 10.1109/CSCloud.2016.47.
[16] A. Yulfitri, "Modeling operational model of data governance in government: case study: government agency X in Jakarta," in *International Conference on Information Technology Systems and Innovation (ICITSI)*, Bandung, Indonesia, Oct. 2016, pp. 1–5. doi: 10.1109/ICITSI.2016.7858207.
[17] R. Thiel, K. A. Stroetmann, and P. D. Singleton, "Clinical data governance: legal and ethical challenges," in *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, Valencia, Spain, Jun. 2014, pp. 597–600. doi: 10.1109/BHI.2014.6864435.
[18] L. Xu, C. Jiang, J. Wang, J. Yuan, and Y. Ren, "Information security in big data: privacy and data mining," *IEEE Access*, vol. 2, pp. 1149–1176 Oct. 2014. doi: 10.1109/ACCESS.2014.2362522.
[19] H. Ye, X. Cheng, M. Yuan, et al., "A survey of security and privacy in big data," in *16th International Symposium on Communications and Information Technologies (ISCIT)*, Qingdao, China, Sept. 2016, pp. 268–272. doi: 10.1109/ISCIT.2016.7751634.
[20] S. Tan, D. De, W. Z. Song, J. Yang, and S. K. Das, "Survey of security advances in smart grid: a data driven approach," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 397–422, Oct. 2016. doi: 10.1109/COMST.2016.2616442.
[21] Apache. (2016, Sept. 20). *What Falcon does* [Online]. Available: https://falcon.apache.org/FalconDocumentation.html
[22] Hortonworks. (2016, Sept. 20). *Falcon* [Online]. Available: https://zh.hortonworks.com/apache/falcon
[23] Apache. (2016, Oct. 16). *Data governance and metadata framework for hadoop* [Online]. Available: http://atlas.apache.org
[24] Apache. (2017, Mar. 16). *Atals architecture* [Online]. Available: http://atlas.apache.org/Architecture.html
[25] M. Rouse. (2015, May 23). *Data-lake* [Online]. Available: http://searchaws.techtarget.com/definition/data-lake
[26] Hortonworks. (2015, Dec. 1). *How Ranger works* [Online]. Available: https://hortonworks.com/apache/ranger
[27] Cloudera. (2016, May 23). *Sentry* [Online]. Available: http://www.cloudera.com/content/cloudera/en/products-adn-services/cdh/sentry.html
[28] J. Kohl and C. Neuman, "The kerberos network authentication service," Internet RFC 1510, Sept. 1993.
[29] Apache. (2016, Mar. 25). *Apache Sentry* [Online]. Available: https://blogs.apache.org/sentry/entry/sentry_graduates_to_a_ top?platform=hootsuite

Manuscript received: 2017-06-15

Biographies

HU Baiqing (baibenny@foxmail.com) received his B.Eng. degree in software engineering from Wuhan Textile University, China in 2016. He is pursuing an M.Eng. degree at Beijing Institute of Technology, China, with a major in software engineering. His research interests include cloud computing, big data, and the Internet of Things.

WANG Wenjie (wangwj1203962899@gmail.com) received his B.Eng. degree in software engineering from Chongqing University, China in 2016. He is pursuing an M.Eng. degree at Beijing Institute of Technology, China, with a major in software engineering. His research interest is the security of big data.

Chi Harold Liu (chiliu@bit.edu.cn) received his B.Eng. degree in electronic and information engineering from Tsinghua University, China in 2006, and Ph.D. degree in electrical engineering from Imperial College, UK. He is currently a full professor and Vice Dean of School of Computer Science, Beijing Institute of Technology, China. His research interests include big data and the Internet of Things.

ZTE Communications Guidelines for Authors

Remit of Journal

ZTE Communications publishes original theoretical papers, research findings, and surveys on a broad range of communications topics, including communications and information system design, optical fiber and electro–optical engineering, microwave technology, radio wave propagation, antenna engineering, electromagnetics, signal and image processing, and power engineering. The journal is designed to be an integrated forum for university academics and industry researchers from around the world.

Manuscript Preparation

Manuscripts must be typed in English and submitted electronically in MS Word (or compatible) format. The word length is approximately 3000 to 8000, and no more than 8 figures or tables should be included. Authors are requested to submit mathematical material and graphics in an editable format.

Abstract and Keywords

Each manuscript must include an abstract of approximately 150 words written as a single paragraph. The abstract should not include mathematics or references and should not be repeated verbatim in the introduction. The abstract should be a self–contained overview of the aims, methods, experimental results, and significance of research outlined in the paper. Five carefully chosen keywords must be provided with the abstract.

References

Manuscripts must be referenced at a level that conforms to international academic standards. All references must be numbered sequentially in–text and listed in corresponding order at the end of the paper. References that are not cited in–text should not be included in the reference list. References must be complete and formatted according to ZTE Communications Editorial Style. A minimum of 10 references should be provided. Footnotes should be avoided or kept to a minimum.

Copyright and Declaration

Authors are responsible for obtaining permission to reproduce any material for which they do not hold copyright. Permission to reproduce any part of this publication for commercial use must be obtained in advance from the editorial office of *ZTE Communications*. Authors agree that a) the manuscript is a product of research conducted by themselves and the stated co–authors, b) the manuscript has not been published elsewhere in its submitted form, c) the manuscript is not currently being considered for publication elsewhere. If the paper is an adaptation of a speech or presentation, acknowledgement of this is required within the paper. The number of co–authors should not exceed five.

Content and Structure

ZTE Communications seeks to publish original content that may build on existing literature in any field of communications. Authors should not dedicate a disproportionate amount of a paper to fundamental background, historical overviews, or chronologies that may be sufficiently dealt with by references. Authors are also requested to avoid the overuse of bullet points when structuring papers. The conclusion should include a commentary on the significance/future implications of the research as well as an overview of the material presented.

Peer Review and Editing

All manuscripts will be subject to a two–stage anonymous peer review as well as copyediting, and formatting. Authors may be asked to revise parts of a manuscript prior to publication.

Biographical Information

All authors are requested to provide a brief biography (approx. 100 words) that includes email address, educational background, career experience, research interests, awards, and publications.

Acknowledgements and Funding

A manuscript based on funded research must clearly state the program name, funding body, and grant number. Individuals who contributed to the manuscript should be acknowledged in a brief statement.

Address for Submission

http://mc03.manuscriptcentral.com/ztecom
12F Kaixuan Building, 329 Jinzhai Rd, Hefei 230061, P. R. China

ZTE COMMUNICATIONS
中兴通讯技术(英文版)

ZTE Communications has been indexed in the following databases:

- Abstract Journal
- Cambridge Scientific Abstracts (CSA)
- China Science and Technology Journal Database
- Chinese Journal Fulltext Databases
- Inspec
- Ulrich's Periodicals Directory
- Wanfang Data

ZTE COMMUNICATIONS
Vol. 16 No. 2 (Issue 62)
Quarterly
First English Issue Published in 2003

Supervised by:
Anhui Science and Technology Department

Sponsored by:
Anhui Science and Technology Information Research Institute;
Magazine House of ZTE Communications

Published and Circulated (Home and Abroad) by:
Magazine House of ZTE Communications

Editorial Correspondence:
Add: 12F Kaixuan Building, 329 Jinzhai Road,
Hefei 230061, P. R. China
Tel: +86-551-65533356
Fax: +86-551-65850139
Email: magazine@zte.com.cn

Annual Subscription: RMB 80

Printed by:
Hefei Tiancai Color Printing Company

Publication Date: June 25, 2018

Publication Licenses:

ISSN 1673-5188
CN 34-1294/ TN