

# Catalyzing Cloud-Fog Interoperation in 5G Wireless Networks: An SDN Approach

Peng Yang, Ning Zhang, Yuanguo Bi, Li Yu, and Xuemin (Sherman) Shen

## ABSTRACT

The growing storage and compute stacks in cloud data centers are expected to accommodate the majority of Internet traffic in the future. However, as the number of mobile devices significantly increases, getting massive data into and out of the cloud wirelessly inflicts high pressure on the bandwidth, and meanwhile induces unpredictable latency. Fog computing, which advocates extending clouds to the network edge, guarantees low latency and location-aware service provisioning. In this article, we consider fog computing as an ideal complement rather than a substitute for cloud computing, and we propose an SDN enabled framework for cloud-fog interoperation, aiming at improving quality of experience and optimizing network resource usage. Two case studies are provided to illuminate the feasibility and advantages of the proposed framework. Finally, potential research issues are presented for further investigation.

## INTRODUCTION

Recently, service providers (SPs) are enthusiastically exploring cloud computing for service provisioning. Typically, cloud vendors provide large-scale storage and compute resource pooling, enabling SPs to customize their level of service on a pay-as-you-go basis. SPs are then able to make faster innovations with reduced capital expenditure and operational expenditure. Consequently, more services are being delivered from the cloud. However, this trend will be shifted when mobile devices become the dominant user equipment. Cisco forecasts that by 2020, there will be 1.5 mobile devices per capita globally [1]. As the popularity of mobile devices surges, the superiority of cloud computing will be adversely affected since getting massive data into and out of the cloud wirelessly requires substantial spectrum resources and incurs unexpected latency, which lead to degraded quality of experience (QoE).

To provide users with a better experience in the coming 5G era, massive MIMO and adaptive access will be exploited for ubiquitous connection and higher data rates. Meanwhile, a cloud-assisted platform is also designed to enhance the operation and management of heterogeneous 5G networks [2]. Among others, *fog computing* offers an alternative approach to ease the wireless networking tensions [3]. Instead of migrating data between the cloud and mobile users, fog computing advocates extending data centers from the network center to the network edge, which greatly reduces the data volume traversed through the core network. Meanwhile, the round-trip latency can be shortened considerably. Moreover, by

implementing fog computing in the vicinity of mobile users, geo-location featured applications (e.g., vehicular networks, e-healthcare systems and smart home systems) can be better supported since data generated in those scenarios will most likely be consumed locally. Yet, fog computing is an ideal complement rather than a substitute for cloud computing. As mobile service types diversify, user requirements can hardly be met by a monotone computing paradigm. Certain services can be better delivered from the cloud while others from the fog. In this way, fog and cloud are interdependent and should be investigated together. However, harnessing them unitedly is challenging due to the following critical questions: What kind of tasks should be processed at the fog? How should the fog interoperate with the cloud to guarantee QoE and, at the same time, maximize the usage of fog resources?

Software defined networking (SDN), characterized by the decoupled control plane and data plane, provides fine-grained network control. Based on real-time global information, the SDN controller is able to make informed management decisions [4]. Because of this, SDN is witnessing growing popularity in complex network management. Liu *et al.* take the advantage of SDN to conduct fine-grained measurement and control in device-to-device based multi-tier LTE-A networks, aiming at providing better QoE [5]. In this article, we propose an SDN enabled architecture to catalyze the cloud-fog interoperation. Through the logical controller and programmable interfaces described in our architecture, the full potential of local pooling resources can be unleashed. Meanwhile, user requests can be dynamically steered among fogs and the cloud so that they can be processed with better QoE. Moreover, the proposed architecture will also benefit network operators in terms of agile management and cost-effectiveness.

The remainder of the article is organized as follows. We introduce fog computing and its applications in the next section, and clarify the importance of cloud-fog interoperation and its challenges in the following section. Then, we propose an SDN enabled architecture for cloud-fog interoperation. Two case studies are conducted later to highlight the feasibility and advantages of the proposed architecture. We present several potential research issues, and finally conclude the article in the last section.

## WHY FOG COMPUTING IN 5G SYSTEM?

Cloud computing provides outsourced infrastructures, platforms and softwares. Being able to deliver such full-scale services on demand, clouds are powerful and cost-effective. However, there are

Peng Yang and Li Yu (corresponding author) are with Huazhong University of Science and Technology.

Ning Zhang and Xuemin (Sherman) Shen are with the University of Waterloo.

Yuanguo Bi is with Northeastern University, China.

Digital Object Identifier: 10.1109/MNET.2017.1600078

inherent limitations that mobile applications will face from the cloud paradigm in the 5G era.

### CLOUD LIMITATIONS

**The Responsiveness:** The coming 5G system is envisioned to provide 1 millisecond (ms) round-trip time. This vision, however, might be hard to achieve since packets of mobile users have to traverse the radio access network (RAN), the core network (CN) and the Internet before reaching the cloud server. Current LTE networks achieve 10 ms round-trip latency, 5 ms in the RAN and CN and 5 ms in the Internet if the server is in the same country as the user [6]. Even if the time spent in the RAN and CN is managed to be within 1 ms in 5G systems, the time spent in the Internet turns out to be the dominant barrier. Moreover, bandwidth limitation and network uncertainty on the way to the clouds will further make latency unpredictable.

**The Backhaul Bottleneck:** Through densification, as well as advanced physical layer techniques, 5G systems are supposed to provide seamless coverage and gigabit data rates. However, even if those measures are adopted to keep users connected, high data rates might still be hard to achieve due to the backhaul bottleneck. Since user-requested content has to be delivered from cloud servers through backhaul, the capacity of backhaul should be at least comparable to that of RAN, which is challenging.

**Location-Aware Applications:** The strong momentum of location based services is another driving force toward fog computing. In those applications, services are often requested by geographically adjacent users, and data exchange mostly happens locally. Unnecessarily directing that traffic to the cloud will not only impair responsiveness, but also aggravate the backhaul tension. If that traffic can be handled around the application contexts, users can be served with better experience and cost effectiveness.

### FOG COMPUTING

Given those cloud limitations, researchers are extensively investigating providing services near mobile users. The measures they have taken are twofold: delivering content near users and offloading users' computation tasks to nearby powerful agents. Ao *et al.* propose to cache content at small base stations, and they present cooperative algorithms that achieve unprecedented content delivery speed at reduced backhaul cost [7]. Chen *et al.* investigate the distributed computation offloading game for mobile edge computing considering latency and energy consumption [8]. These works and references therein exploit either storage or compute resources around users to improve QoE. However, emerging applications such as virtual/augmented reality generate large volumes of data that need to be processed instantly, posing challenges to mobile devices in terms of storage, communication and computation.

Fog computing can help users meet those requirements. By pooling resources from near-user devices and infrastructures, fog computing carries out substantial storage and computation capacity with reduced communication cost. In this way, heavy applications can be delivered with shortened latency, energy efficiency and location awareness [9, 10]. Fascinating fog services can be expected,

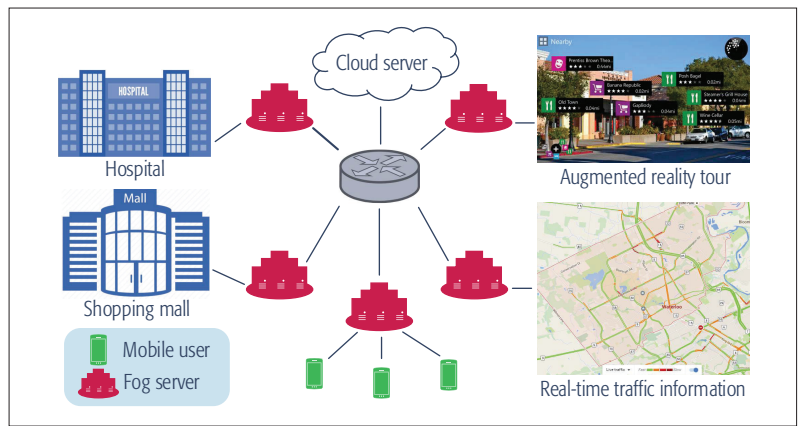


FIGURE 1. The prospective fog applications.

ed, e.g., caching popular content at hotspots can greatly improve responsiveness. Meanwhile, fog can also be harnessed for location-specific purposes. Figure 1 depicts several fog contexts: a fog server in a shopping mall can provide commodity information and navigation to nearby consumers; community-wide fog can be utilized to collect household utility usage for demand analysis by utility companies; and city-wide fog can be leveraged for gathering, analyzing and disseminating real-time road traffic information. The granularity of fogs can vary from body area to city wide.

## CLOUD-FOG INTEROPERATION AND CHALLENGES

### THE INTEROPERATION

In fact, cloud can easily host various fog applications with more than enough resources, but the unpredictable latency and bandwidth waste is unacceptable. Though fog outperforms cloud in terms of responsiveness and location awareness, its lightweight capability will be inadequate when handling heavy tasks. Therefore, fog computing can be complementary to cloud computing, rather than a substitute. Computationally intensive tasks, long-term large-size file storing and sharing, and delay-tolerant services are more appropriate to be handled by the cloud, while the massively deployed fogs are suited to carry applications such as lightweight computing and short lived location-specific content sharing.

More importantly, we are seeing huge potential for cloud-fog cooperation. Take augmented reality tour as an example. While traveling, visitors can enjoy the sights with their smart phones or glasses. Extra information on landmarks or restaurants is displayed on the screen immediately. In this application, images are continuously generated and need to be processed instantly. With cloud-fog cooperation, image features can be first extracted at the local fog, and thereafter the feature information is delivered to the cloud for image matching. In this way, bulk images are processed locally, which avoids bandwidth waste and unexpected latency. Moreover, as the cloud server stores massive data sets, retrieving images in the cloud will return better results.

### THE CHALLENGES

Since requests and content will migrate frequently between fogs and clouds, high-level cloud-fog interoperation should be enabled for coordinated service provisioning. However, empowering the

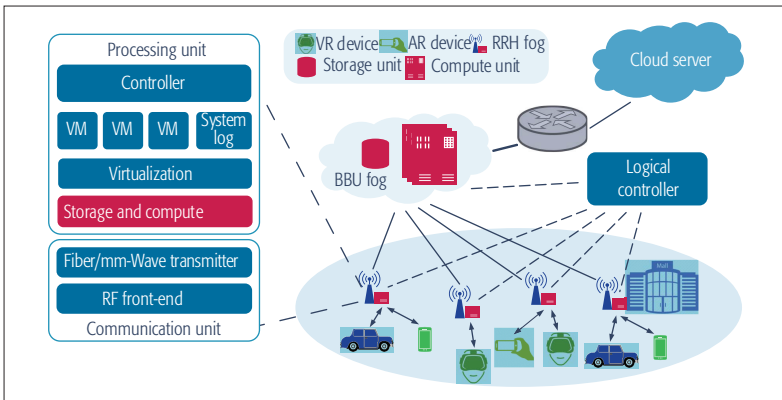


FIGURE 2. The SDN enabled cloud-fog interoperation architecture.

interoperation is a great challenge due to the following requirements.

It needs a *local coordinator*. As service requirements are dynamic and time-varying, certain application might need support from either the cloud or fog or both. Thus, we need a coordinator to preprocess requests. If the request can be cooperatively processed locally, the coordinator will need to decompose and assign tasks to available resources. If the request is more suitable to be processed in the cloud, the coordinator can act as a local hub to direct it to the cloud.

It needs *global knowledge*. The cloud and fog might be operated by several vendors, and it is necessary to coordinate how the SPs are accommodated. Therefore, gathering high-level and real-time information regarding these facilities is a key enabler of cloud-fog interoperation. Once the coordinator is well informed, it can make optimal decisions. Furthermore, a network state information renewal mechanism should be designed, as well as how to manage and make the best use of collected information.

It needs *open programmable interfaces*. Typical applications will extensively exchange information among mobile users, fog/cloud servers and service providers. In this way, each entity needs to make certain functions available to others through open interfaces, so that high-level policies from the controller can be dynamically enforced to different entities.

## SDN ENABLED CLOUD-FOG COMPUTING ARCHITECTURE

In legacy networks, the control plane and data plane are tightly coupled, and protocols running in switches and routers are immutable once they are installed. Hence the network is highly ossified and the operator can hardly innovate upon it. SDN, a new network paradigm that enables control-data separated operation, gives the controller a global view of the network via a programmable control plane. Consequently, it contributes to dynamic network deployment, agile network management, faster application innovation and efficient resource utilization [4, 5]. Cultivating SDN to catalyze the cloud-fog interoperation can well meet the requirements mentioned in the previous section. Therefore, we propose an SDN enabled architecture for cloud-fog interoperation.

**Components:** As illustrated in Fig. 2, the proposed architecture incorporates mobile users, fog servers and the cloud server, where storage and computing resources beyond the access network are referred to as the cloud. Fogs are deployed on top of cloud-RAN (C-RAN). C-RAN is an ideal carrier for fog deployment, as it divides the function of conventional base stations into two parts: remote radio head (RRH) for radio signal transceiving, and baseband unit (BBU) for high-speed baseband processing [11]. By stacking storage and computing resources on the BBU and RRHs, C-RAN can well carry the facilities for fog computing.

RRH fogs are widely scattered and attached to the centralized BBU fog (20-40 km away [11]) via fiber link or millimeter-wave. In addition to the communication unit, each RRH fog is equipped with a processing unit. The storage and computing resources are virtualized as isolated virtual machines (VMs) [12], which are managed by a local fog controller. The functionalities of RRH fogs are twofold: they perform baseband processing after radio signals are received, and they provide local storage and computing resources that can be dynamically scheduled by the controller at the BBU fog.

The BBU fog is stacked with more processing units, making it the most powerful fog within RAN. In addition to the functionalities of RRH fogs, the BBU controller also acts as a master controller (MC) that coordinates all the controllers in the RRHs. Note that this configuration is compatible with legacy C-RAN, and thus can be incrementally deployed.

**Interoperation:** With the coordination of the MC, local network information is continuously shared among controllers, leading to a logic controller with group intelligence. The controllers evaluate user requests and make corresponding responses through either fog-fog interoperation or cloud-fog interoperation.

Fog-fog interoperation is initiated when user demands are beyond the capability of an individual fog. Upon receiving such requests, the logical controller starts a crowdsourcing process [13]. Based on the available resources on each fog server, the MC decomposes the tasks and distributes them accordingly. Once the decomposed tasks are all completed, the MC recomposes the results and delivers them to mobile users.

Cloud-fog interoperation is activated when user demands cannot be met by fog computing alone. In this regard, the MC abstracts the user requirements and evaluates whether fog preprocessing will accelerate the service provisioning (recall the example of the augmented reality tour mentioned earlier). Then the MC either starts a crowdsourcing process through fog-fog interoperation or sends user requests directly to the cloud.

With interoperation, more user requests can be handled locally, and fog resource efficiency is pushed to the maximum. Taking fog caching as an example, frequently requested files can be pushed to fogs from the cloud. Thus, mobile users can fetch content from neighboring servers through fog-fog interoperation. As fog storage is limited compared to the cloud, only selected content can be cached and dated files should be evicted continuously. Thus, to obtain a higher



local request hit ratio, cloud and fog have to interact frequently to maintain the optimality of fog content entry.

### LOGICAL CONTROLLER

Figure 3 depicts the framework of a logical controller geared to service provisioning. The logical controller is responsible for bridging mobile users and fog service providers. On one hand, it processes the requests from mobile users, and anonymously records them in the system log. On the other hand, it periodically updates the events and VM operating information of each fog in the system log, such as pending tasks, amount of users served and remaining resources. The controller thus has global knowledge of the network status and can always make informed scheduling decisions based on what it has learned from the system log.

As the local network operator, the logic controller also manages the access network including authentication, authorization and accounting (AAA), mobility management entity (MME), and resource allocation (RA). The system log is essential for the controller's decision making as it provides information of user requests and available resources. More importantly, those historical statistics can be exploited for learning and predicting the future network behaviors so as to take proactive actions. The prediction results are expected to be more accurate as the system log files expand.

### THE PROGRAMMABLE INTERFACES

Generally, delivering services involves three other entities in addition to the controller, i.e., mobile users, fog/cloud servers and SPs, and all of them are intertwined. As a practical scenario, music streaming company Spotify uses Amazon's simple storage service to store its music files while keeping its core computing infrastructure (or its backend) in Google's cloud platform to process and record user requests (<http://www.wsj.com/articles/google-cloud-lures-amazon-web-services-customer-spotify-1456270951>). Since information is extensively exchanged among those entities, standardized application programming interfaces (API) are pivotal. In computer networks, OpenFlow is the dominant software interface between the SDN controller and the underlying switches. An extension of OpenFlow can be made to support the communication between the logical controller and fog facilities [14]. Generally, APIs can be classified into the following two categories.

- Functional APIs: Make functionalities at each entity available to others for service delivering, such as data migration, VM configuration and scheduling.
- Management APIs: Used for AAA, mobility management, service billing, etc.

Once those APIs are elaborated and standardized, a user's requirements can be cooperatively fulfilled by heterogeneous SPs with minimum impact on performance.

### ADVANTAGES AND VISIONS

The innate SDN features of the proposed architecture are key enablers toward cloud-fog interoperation. Specifically, they offer the following advancements.

- Real-Time Knowledge: Real-time network

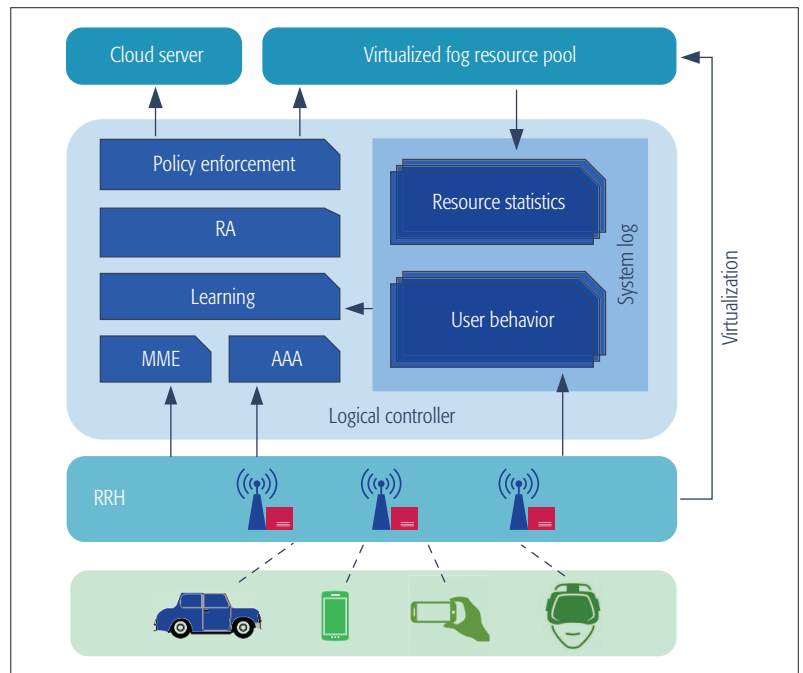


FIGURE 3. The elements of logical controller for service provisioning.

status information is essential to the controller's decision making. The dedicated control plane makes the update of local information reliable and flexible.

- Centralized Operation: The logical controller schedules and optimizes the utilization of all the resources within RAN. Eventually, the norm of wireless networks will be locally centralized and globally distributed.
- Fine-Grained Control: The dedicated control channel exposes traditionally inaccessible functional APIs to the controller. In turn, physical devices translate high-level policies into low-level configuration instructions, whereby fine-grained control is achieved.

Consequently, the proposed architecture has great potential to enhance the network performance in the following aspects.

**Improved QoE:** Being able to deliver services in the vicinity of mobile users, the SDN enabled cloud-fog interoperation architecture will largely reduce latency and jitter compared to the standalone cloud solution. The architecture will offer users dramatically improved responsiveness and make the 1 ms latency achievable in the coming 5G era.

As the human-computer interface develops, users are now expecting to get information and entertainment in highly interactive ways. Exciting applications include deeply immersive augmented reality and virtual reality. In those contexts, users' devices interact constantly with application servers. For instance, as user's eyes roll, the virtual reality devices should show the exact sight that the user is looking at. Although computationally demanding, application servers have to make continuous prompt responses, otherwise the user experience will quickly deteriorate since humans are extremely sensitive to visual stutter. As shown later in the case study, the proposed architecture can incorporate those applications with improved QoE.

**Resource Pooling Economics:** In spite of the lightweight resources deployed in the fog, central-

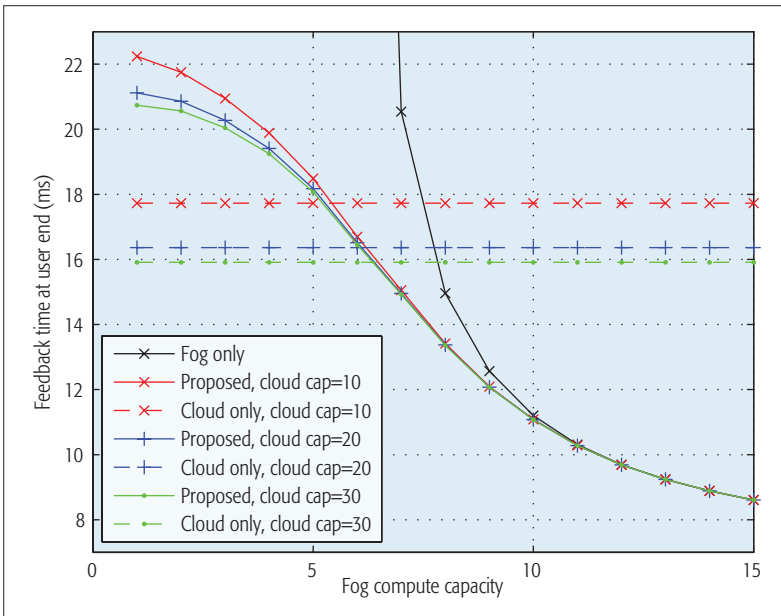


FIGURE 4. Average feedback time at user end versus fog compute capacity. The colored solid lines illustrate the results under the proposed architecture, while the black solid line indicates the results of fog only scenario, dashed lines depict the results of cloud only scenario.

ized operation and virtualization drive the multiplexing efficiency of resources to the maximum. In addition, since the traffic traversing backhaul is reduced, the expense on the backhaul facilities will also scale down. Moreover, wireless carriers have kept investing in deploying, operating and upgrading the network for decades. However, as value-added services are mostly delivered by the Internet SPs, wireless carriers have long been suffering from shrinking profits, while coping with the endless insatiable demands for higher data rates. This situation will be turned around if value-added services can be delivered within RAN, since carriers will have a clear advantage to provide high-quality services to mobile users.

**Agile Network Management:** Through a dedicated control channel, the network operator obtains timely global information. Via open programmable interfaces, informed decisions and policies can be dynamically enforced to the network components. Being separated from the underlying hardware, network management and orchestration are totally software based, which makes high-level operational automation possible. In this way, network operators are free of endless configuration, aimless debugging and maintenance.

## CASE STUDIES

In this section, the feasibility and advantages of the SDN enabled architecture are illuminated by two case studies, i.e., crowdsourcing task scheduling and popularity-aware content caching.

### CROWDSOURCING TASK SCHEDULING

Crowdsourcing is a new paradigm that solicits collective collaboration to support applications that require either broad participation or high performance computation [13]. Those requirements generally exceed the capacity of any single agent. As such, multiple agents need to contribute collectively. For example, neighboring fog servers on the road need to cooperate to perform road

traffic prediction [9]. In those scenarios, the MC needs to decompose and distribute tasks to adjacent fog servers in a crowdsourcing manner. So in this study, we evaluate how SDN enabled cloud-fog interoperation can improve the responsiveness of computational crowdsourcing tasks.

We make the following assumptions:

- The local crowdsourcing tasks are processed on a first in, first out (FIFO) basis.
- The cloud starts processing the task immediately upon receipt.
- The MC makes scheduling decisions according to the following rule: if the estimated fog response time (defined as the duration between the first run time and arrival time) of crowdsourcing is shorter than the request delivery time to the cloud, it decomposes the task and puts it in the FIFO queue; otherwise it sends the request to the cloud.

The measurements of task workload and fog/cloud computing capacity are normalized. Assume the MC processes 1000 crowdsourcing requests, arriving according to a Poisson process with parameter 0.25, and each with uniformly distributed workload between 0 and 40. We measure the feedback time at the user end (including the round-trip time for request delivery, the response time and the processing time) under different scenarios, so as to evaluate the performance of the proposed architecture. The round-trip time of local crowdsourcing and cloud computing are set to 6 ms and 15 ms, respectively [6].

Figure 4 illustrates the trends of how the average feedback time at the user end varies with the increase in computing resources at the fog. It can be seen that under the proposed architecture, the average feedback time at the user end drops rapidly below the cloud-only scenario. When the fog computing capacity is higher than 5, it is capable of processing most of the requests, so the cloud capacity has little impact on feedback time. The results under the proposed architecture converge to the fog-only scenario, indicating that the fog is able to process all the requests after its computing capacity exceeds 10. The benefits of the proposed scheme can be even more striking if the proportion of lightweight tasks increases.

This study confirms that the proposed architecture can well catalyze cloud-fog interoperation, offering better QoE with lightweight fog resources (computing capacity between 5 and 10 in this case). The observation is that the MC should keep real-time information of available local resources so as to make optimal scheduling decisions. Meanwhile, more fog resources will absolutely improve QoE, but the capital expenditure and operational expenditure will also increase. Thus, the insight of this study can be used to guide fog resource deployment in practice.

### POPULARITY-AWARE CONTENT CACHING

Content sharing, especially video streaming, will generate the majority of mobile traffic in the future [1]. In this study, we focus on content caching, and evaluate how the SDN enabled architecture improves QoE while alleviating the backhaul bandwidth pressure.

Consider the scenario in which mobile users continuously request video clips from content servers. Without interoperation, the fog server

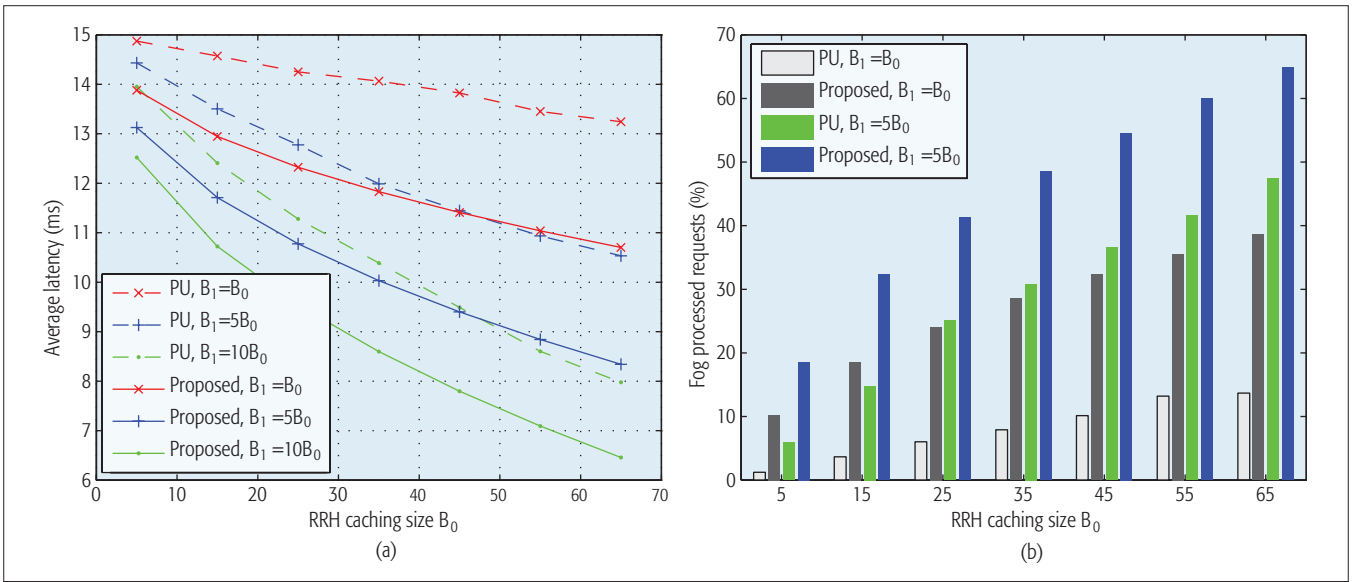


FIGURE 5. The comparison between interoperation enabled caching strategy and popularity unaware (PU) caching: a) average latency versus the RRH caching size; b) percentage of fog processed requests versus the RRH caching size.

can only cache content in a popularity-unaware manner, such as caching the latest requested content, while in the proposed architecture, since user requests are properly recorded in the system log, the controller can easily identify the hot video content and cache it from the cloud.

We consider a medium-sized urban network where a BBU supports 100 RRHs [11], and each RRH serves 500 mobile users. Assume there are a total  $K = 1000$  video clips in the cloud server. Since we can crop the large files into small pieces, all video clips are considered with unit size for simplicity. Their popularity distribution follows Zipf's law with parameter  $\alpha$ , that is, the content ranking the  $i$ th is requested with frequency  $f_i = c/i^\alpha$ ,  $i = 1, \dots, K$ . Set  $\alpha = 0.56$  [7] and let the normalization constant  $c$  equal 1. Assume the storage size of the RRH is  $B_0$ , and that of the BBU is  $B_1$ . We assume the round-trip communication from mobile users to its nearest RRH fog, BBU fog and the cloud is 3 ms, 6 ms and 15 ms, respectively [6]. As interoperation is realized, we employ the following straightforward caching policy: the controller always caches the most popular  $B_0$  content from the cloud at the RRH, while it caches the less popular  $B_1$  content at the BBU. The requests that are not met at the RRH or BBU are sent to the BBU or cloud, accordingly.

Figure 5a shows that compared to the popularity-unaware scenario, our interoperation enabled strategy provides better latency performance. Figure 5b shows that fog servers are able to process more user requests under the proposed architecture, which in turn reduces traffic traversing the backhaul. Again, this study confirms the advantages of the proposed architecture. It can be seen that popularity-aware caching under our architecture can effectively improve QoE and alleviate backhaul pressure, especially under the scenario with fewer fog resources ( $B_1 = B_0$ ).

Moreover, as the popularity of video clips changes over time, content in fogs must be dynamically added or evicted. By employing machine learning techniques in the system log, the logical controller can capture the ascending or descending trends of certain content, thus pro-

actively adding or evicting instructions can provide better QoE with limited resources.

### REMARKS

As we can see from the case studies, the proposed architecture can improve QoE, and also benefit the network operator in terms of reduced backhaul pressure and agile network management. However, the cost for deploying this promising architecture should also be considered. First, network operators should investigate further the deployment of initial fog infrastructures, retraining their staff and customizing new business models within the new architecture. Second, SPs will also have to investigate the possible candidate services that can be better delivered from the fog, and balance the fog and cloud for reliability, responsiveness and cost effectiveness concerns.

### POTENTIAL RESEARCH ISSUES

Leveraging SDN for cloud-fog interoperation is promising yet challenging. In this section, we identify several specific research issues that can be further investigated to unleash the potential of the proposed architecture.

**Security and Privacy:** Generally, the fog is more secure than the cloud, since user requests are processed locally and the risk of exposing user data is reduced. However, security and privacy issues also arise from several factors. First, the centralized SDN paradigm may suffer from attacks on the control channel and the controller, where the network could be fatally comprised [4, 15]. Hence, designing a secure control channel and reliable controller is of paramount importance. Second, users' privacy is also at stake, especially in the crowdsourcing applications [13]. As the MC distributes user requests to multiple fog servers, users' private information disclosure at the fog server is inevitable. Hence, designing secure and reliable communication strategies is non-trivial.

**Cloud-Fog Resource Balancing:** The case studies indicates that mounting fogs with larger computing and storage capacity will significantly boost performance, but the resulting capital and

Fogs are necessarily managed by different network operators, who need to cooperate closely to make the fog-fog and cloud-fog interoperation possible. An optimal cooperation model and incentive mechanisms that stimulate more participation by operators should be designed.

operational expenditures also matter. In this way, we should investigate the optimal amount of fog resources that need to be deployed, especially when users' demands are fluctuating.

**Controller Design:** The controller should be knowledgeable and intelligent. With the coordination of the MC, the controllers are expected to work cooperatively so that the limited fog resources can be efficiently utilized. Moreover, as the system log expands, the learning pattern from the history statistics can well serve the purpose of traffic prediction and dynamic resource allocation. The controller can even capture the pattern of users' accessing the network. Certain APIs should be designed to support the controller's intelligence at different levels.

**Cooperation among Network Operators:** Fogs are necessarily managed by different network operators, who need to cooperate closely to make the fog-fog and cloud-fog interoperation possible. An optimal cooperation model and incentive mechanisms that stimulate more participation by operators should be designed.

## CONCLUSION

In this article, we have presented an SDN enabled architecture for cloud-fog interoperation, which directly contributes to improved QoE and agile network management. We first discuss the necessity of fog computing by arguing that cloud computing will introduce unwanted latency and backhaul bottlenecks and it is not well-suited for location based service provisioning. We then present how an SDN controller can be exploited to bridge the interoperation of cloud and fog computing. Two case studies are discussed to illuminate the feasibility and advantage of the SDN enabled cloud-fog architecture. Potential research issues are proposed to further harness SDN for cloud-fog interoperation.

## ACKNOWLEDGMENT

This research work is supported by the National Natural Science Foundation of China under Grant No. 61231010; the Research Fund for the Doctoral Program of MOE of China under Grant No. 20120142110015, and the Natural Sciences and Engineering Research Council (NSERC) of Canada. Peng Yang is also financially supported by the China Scholarship Council.

## REFERENCES

- [1] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015–2020," White Paper, Feb. 2016.
- [2] N. Zhang *et al.*, "Cloud Assisted HetNets Toward 5G Wireless Networks," *IEEE Commun. Mag.*, vol. 53, no. 6, 2015, pp. 59–65.
- [3] T. H. Luan *et al.*, "Fog Computing: Focusing on Mobile Users at the Edge," arXiv: 1502.01815, 2015.
- [4] D. Kreutz *et al.*, "Software-Defined Networking: A Comprehensive Survey," *Proc. IEEE*, vol. 103, no. 1, 2015, pp. 14–76.
- [5] J. Liu *et al.*, "Device-to-Device Communications for Enhancing Quality of Experience in Software Defined Multi-Tier LTE-A Networks," *IEEE Network*, vol. 29, no. 4, 2015, pp. 46–52.
- [6] GSMA Intelligence, "Understanding 5G: Perspectives on Future Technological Advancements in Mobile," White Paper, Dec. 2014.

- [7] W. Ao *et al.*, "Distributed Caching and Small Cell Cooperation for Fast Content Delivery," *ACM MobiHoc*, 2015, pp. 127–36.
- [8] X. Chen *et al.*, "Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, 2016, pp. 2795–2808.
- [9] X. Hou *et al.*, "Vehicular Fog Computing: A Viewpoint of Vehicles as the Infrastructures," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, 2016, pp. 3860–73.
- [10] F. Jalali *et al.*, "Fog Computing May Help to Save Energy in Cloud Computing," *IEEE JSAC*, vol. 34, no. 5, 2016, pp. 1728–39.
- [11] A. Checko *et al.*, "Cloud RAN for Mobile Networks—A Technology Overview," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, 2015, pp. 405–26.
- [12] C. Liang *et al.*, "Wireless Network Virtualization: A Survey, Some Research Issues and Challenges," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, 2015, pp. 358–80.
- [13] K. Yang *et al.*, "Security and Privacy in Mobile Crowdsourcing Networks: Challenges and Opportunities," *IEEE Commun. Mag.*, vol. 53, no. 8, 2015, pp. 75–81.
- [14] K. K. Yap *et al.*, "OpenRoads: Empowering Research in Mobile Networks," *ACM SIGCOMM Computer Commun. Rev.*, vol. 40, no. 1, 2010, pp. 125–26.
- [15] A. Akhunzada *et al.*, "Securing Software Defined Networks: Taxonomy, Requirements, and Open Issues," *IEEE Commun. Mag.*, vol. 53, no. 4, 2015, pp. 36–44.

## BIOGRAPHIES

PENG YANG [S] (yangpeng@hust.edu.cn) received his B.Sc. degree from the Department of Electronics and Information Engineering, Huazhong University of Science and Technology (HUST), Wuhan, China, in 2013. Currently, he is pursuing his Ph.D. degree in the School of Electronic Information and Communications, HUST. From Sept. 2015, he has also been a visiting Ph.D. student in the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His current research interests include next generation wireless networking, software defined networking and fog computing.

NING ZHANG [M] (n35zhang@uwaterloo.ca) earned his Ph.D. degree from the University of Waterloo in 2015. He received his B.Sc. degree from Beijing Jiaotong University and M.Sc. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2007 and 2010, respectively. He is now a postdoc research fellow at the BBCC Lab at the University of Waterloo. His current research interests include next generation wireless networks, software defined networking, green communication, and physical layer security.

YUANGUO BI [M] (biyuanguo@ise.neu.edu.cn) received his B.Sc. degree from Liaoning University, Shenyang, China, in 2003, and his M.Sc. and Ph.D. degrees from Northeastern University, Shenyang, in 2006 and 2010, respectively, all in computer science. From 2007 to 2009, he was a visiting Ph.D. student with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. In 2010, he joined Northeastern University as an associate professor. His current research interests include medium access control, QoS routing, and mobility management in vehicular networks.

LI YU [M] (hustlyu@hust.edu.cn) received her B.Sc. and Ph.D. degrees from the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan, China, in 1992 and 1999, respectively. She joined Huazhong University of Science and Technology in 1999, where she is now a professor in the School of Electronic Information and Communications. Her current research interests include image and video coding, multimedia communications, social networks and wireless networking.

SHERMAN (XUEMIN) SHEN [F] (xshen@bbccr.uwaterloo.ca) is a professor and University Research Chair, Department of Electrical and Computer Engineering, University of Waterloo, Canada. He was the Associate Chair for Graduate Studies from 2004 to 2008. His research focuses on resource management in interconnected wireless/wired networks, wireless network security, social networks, smart grid, and vehicular ad hoc and sensor networks. He served as Technical Program Committee Chair/Co-Chair for IEEE GLOBECOM'16, IEEE INFOCOM'14, IEEE VTC-Fall'10 and IEEE GLOBECOM'07; Symposia Chair for IEEE ICC'10; and Tutorial Chair for IEEE VTC-Spring'11 and IEEE ICC'08. He also serves or has served as editor-in-chief of *IEEE Network*, *Peer-to-Peer Networking and Applications*, and *IET Communications*. He is a registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and IEEE Communications Society.