Software Defined Networking, Caching, and Computing for Green Wireless Networks

Ru Huo, Fei Richard Yu, Tao Huang, Renchao Xie, Jiang Liu, Victor C.M. Leung, and Yunjie Liu

ABSTRACT

Recent advances in networking, caching, and computing will have a profound impact on the development of next generation green wireless networks. Nevertheless, these three important areas have traditionally been addressed separately in existing works. In this article, we propose a novel framework that jointly considers networking, caching, and computing techniques in a systematic way to naturally support energy-efficient information retrieval and computing services in green wireless networks. This integrated framework can enable dynamic orchestration of different resources to meet the requirements of next generation green wireless networks. Simulation results are presented to show the effectiveness of the proposed framework. In addition, we discuss a number of challenges in implementing the proposed framework in next generation green wireless networks.

INTRODUCTION

Increasingly rigid environmental standards and rising energy costs have led to great interest in improving the energy efficiency of green wireless networks [1, 2]. Recent advances in networking, caching, and computing have been extensively studied in the development of green wireless networks. In the area of networking, software-defined networking (SDN) has attracted great interest in both academia and industry. SDN introduces the ability to program the network via a logically software-defined controller, and separates the control plane from the data plane. It is beneficial to extend SDN to wireless networks [3, 4]. Software defined wireless networks enable direct programmability of wireless network controls and abstraction of the underlying infrastructure for wireless applications, with improved energy efficiency and great flexibility in green wireless network management [5].

Information-centric networking (ICN) has been extensively studied in recent years. To promote the content to first-class citizenship in the network, in-network caching is used in ICN to speed up content distribution and improve network resource utilization. In ICN, requests no longer need to travel to the content source, but are typically served by a closer ICN "content node" along the path. Information-centric wireless networks enable content caching in both the air and the mobile devices, which has been recognized as one of the promising techniques for next generation green wireless networks [6, 7].

In the area of computing, cloud computing has been widely adopted to enable convenient access to a shared pool of computing resources [8]. Cloud computing will have a profound impact on green wireless networks as well. The introduction of cloud computing to wireless mobile networks enables mobile cloud computing (MCC) systems, and cloud computing for radio access networks leads to green cloud radio access networks (C-RAN) [9]. Nevertheless, as the distance between the cloud and the edge device is usually large, cloud computing services may not provide guarantees to low-latency applications, and transmitting a large amount of data (e.g., in big data analytics) from the device to the cloud may not be feasible or economical. To address these issues, fog computing [10] and mobile edge computing (MEC) [11] have been proposed to deploy computing resources closer to end users.

Although some excellent work has been done on networking, caching, and computing in wireless networks, these three important areas have traditionally been addressed separately in existing studies. In this article, we propose to jointly consider networking, caching, and computing techniques in order to improve the performance of next generation green wireless networks. Specifically, based on the programmable control principle originated in SDN, we incorporate the ideas of information centricity originated in ICN. This integrated framework can enable dynamic orchestration of networking, caching, and computing resources to meet the requirements of next generation green wireless networks.

The rest of this article is organized as follows. We describe an overview of SDN, ICN, cloud/fog computing, and MEC. We present the proposed framework that integrates networking, caching, and computing for next generation green wireless networks. Simulation results are presented and discussed. Some open research issues are discussed. Finally, we conclude this study.

OVERVIEW OF SOFTWARE-DEFINED NETWORKING, INFORMATION-CENTRIC NETWORKING, CLOUD/FOG COMPUTING, AND MOBILE EDGE COMPUTING

In this section, we briefly introduce SDN, ICN, cloud/fog computing, and MEC.

The authors propose a novel framework that jointly considers networking, caching, and computing techniques in a systematic way to naturally support energy-efficient information retrieval and computing services in green wireless networks. This integrated framework can enable dynamic orchestration of different resources to meet the requirements of next generation green wireless networks.

This work is jointly supported by the National High Technology Research and Development Program (863) of China (No. 2015AA016101), the National Natural Science Foundation of China (No. 61501042), Beijing Nova Program (No.Z151100000315078), and the Natural Sciences and Engineering Research Council of Canada (NSERC).

Ru Huo, Tao Huang, Renchao Xie, Jiang Liu, and Yunjie Liu are with Beijing University of Posts and Telecommunications and Beijing Advanced Innovation Center for Future Internet Technology; Fei Richard Yu is with Carleton University; Victor C. M. Leung is with the the University of British Columbia.

Digital Object Identifier: 10.1109/MCOM.2016.1600485CM

 0163-6804/16/$25.00

SDN is a new type of network architecture, realizing the separation of the data plane and the control plane, and programming on the control plane directly. The separation of the data plane and the control plane is conducive to abstract and manage the infrastructure and resources of underlying networks.

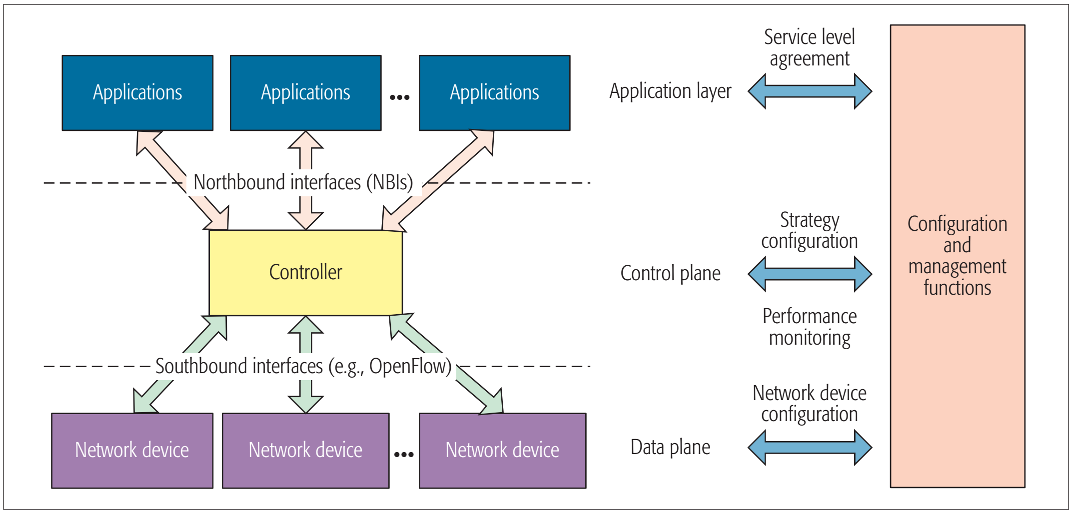

Figure 1. A typical architecture of SDN.

Software-Defined Networking

The idea of SDN, which originated in the Ethane project, was proposed by the Clean Slate Research Group of Stanford University. After that, first the Global Environment for Network Innovations (GENI) project conducted experimental development of OpenFlow-based SDN in campus networks and backbone-sized networks. Then some industrial companies established the Open Networking Foundation (ONF) organization, intending to promote the technology of SDN, which took OpenFlow as representative through the industry alliances. Furthermore, Google Inc. deployed the notable B4 network over its 12 global data centers based on SDN, for traffic engineering between the data centers. All these actions and practices have played important roles in a great boom in the global technology research and industrialization of SDN.

SDN is a new type of network architecture, realizing the separation of the data plane and the control plane, and programming on the control plane directly. The separation of the data and control planes is conducive to abstracting and managing the infrastructure and resources of underlying networks. Actually, SDN has a logically centralized and programmable control plane with open interfaces. Meanwhile, the control function is no longer confined to routers, or programmed and defined only by the manufacturers of equipments. Therefore, SDN achieves better flexibility and controllability.

A typical architecture of SDN is shown in Fig. 1. The control plane of SDN located in the middle of this architecture is implemented on the basis of software, and is responsible for monitoring global information and realizing network intelligence. Thus, the SDN network device acts as a logical switching device in terms of the upper applications and strategies, which greatly simplifies the control and operation of the network. Meanwhile, in terms of the underlying data forwarding plane, the network switching equipments do not have to support a large number of protocol standards, but only accept the instructions from the controller, which also fully simplifies the data forwarding plane. With OpenFlow, switching equipment and control equipment can communicate with each other. That means the flow table, which is a core data structure in the equipment of the data forwarding plane used to implement forwarding, is programmed by SDN through the network interaction protocol. A controller will send, delete, and modify the flow table in switches of the data forwarding plane, to guide data or traffic forwarding of switches according to the regulations defined by OpenFlow. According to the items in the flow table, a packet matching the rules will be handled following the responding actions.

Information-Centric Networking

With a variety of applications appearing on the Internet, one of the main applications is to request massive contents. Popular contents are transmitted repeatedly on the Internet, wasting resources and reducing quality of service (QoS). In order to meet the essential behavioral patterns of the existing network, which requests and acquires information, a set of clean slate network architectures, ICN, has been proposed to change the traditional communication mode based on end-to-end communications [6]. Among these architectures, here we choose named data networking (NDN) [12] as a typical representative to introduce ICN, because it uses content name to route and retrieve, which is a direct method of content delivery. The hourglass model of the IP protocol stack is retained in NDN, but the waist layer is changed with a hierarchical content naming structure. This naming structure is similar to URLs. Meanwhile, all the routing nodes of NDN are equipped with caches or repositories. Therefore, routing, forwarding, and caching are all implemented based on flexible naming. Besides, NDN integrates security into data itself by cryptographically signing every data packet to ensure integrity and authenticity.

In the communication process of NDN, there are an interest packet and a data packet. The interest packet contains content name, and the data packet contains content name, encrypted information, and data. The router of NDN contains the content store (CS), pending interest table (PIT), and forwarding information base (FIB). The data packets are cached in the CS

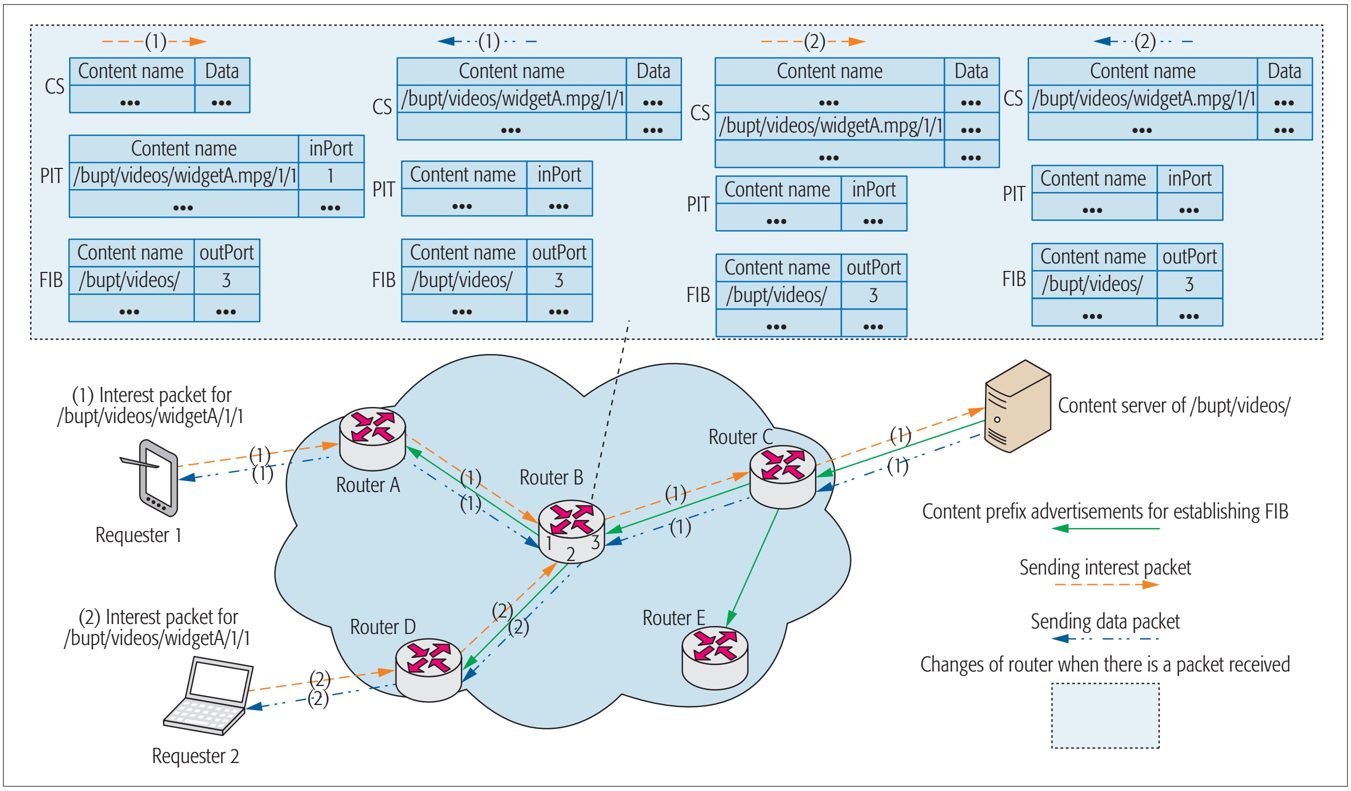

Figure 2. A typical architecture of ICN.

and reused by other relevant interest packets. The port where the interest packets originate is recorded in the PIT, and once the responding data packets arrive, the recording items are deleted. The same interest packets from different ports are recorded in one item to avoid repetitive transmitting of data, called aggregation of PIT. When content providers publish contents to the network, the routing table is established according to some rules in the FIB. A typical architecture of ICN is shown in Fig. 2. The content server first publishes advertisements, and then the routing information is established in routers. When an interest packet for */bupt/videos/widgetA/1/1* is sent from requester 1, it will be forwarded to the content server of */bupt/videos/*, and the data packet will be cached in routers A, B, and C. When the interest packet for */bupt/videos/widgetA/1/1* is sent from requester 2, it is forwarded to the content server of */bupt/videos/*. However, if there is a copy cached in router B, the data packet is sent directly from router B.

It is worth mentioning that the caching function or storage capacity in network devices is also an excellent method to avoid repetitive transmitting. Using caching or storage resource could reduce traffic redundancy and decrease delay and energy consumption, which is extremely beneficial to green applications of distributing and processing massive contents.

Cloud Computing, Fog Computing, and Mobile Edge Computing

Cloud computing has been widely adopted to enable convenient, energy-efficient, on-demand network access to a shared pool of configurable computing resources [13]. Cloud computing belongs to a new large-scale distributed computing paradigm and treats everything as a service (XaaS). It has different meanings to different people. For applications and users, it provides computing, storage, and applications over the Internet from centralized data centers (software as a service, SaaS). For developers, it is an Internet-scale software development platform and runtime environment (platform as a service, PaaS). For infrastructure providers and administrators, it is a massive and distributed data center infrastructure connected by IP networks (infrastructure as a service, IaaS).

Although cloud computing has been applied in diverse domains, most services need to be processed in data centers, which are usually far away from the end users. Therefore, cloud computing services may not provide guarantees to low-latency applications, and transmitting a large amount of data (e.g., in big data analytics) from the device to the cloud may not be feasible or economical. To address these issues, fog computing [10] (also called edge computing) has been proposed. The edge devices mainly refer to the local servers with weaker performance and different functions. These devices compose a distributed computing system, like personal cloud, private cloud, and enterprise cloud. Low latency and location awareness are two obvious characteristics of fog computing. Similar to the concept of edge/fog computing, MEC has recently been proposed, particularly for RANs [11], in close proximity to mobile subscribers. It is defined as running IT-based servers (called MEC servers) at the edge of a RAN, applying the concept of cloud computing, and combining computing and storage capacities with a mobile base station to accelerate contents, services, or applications.

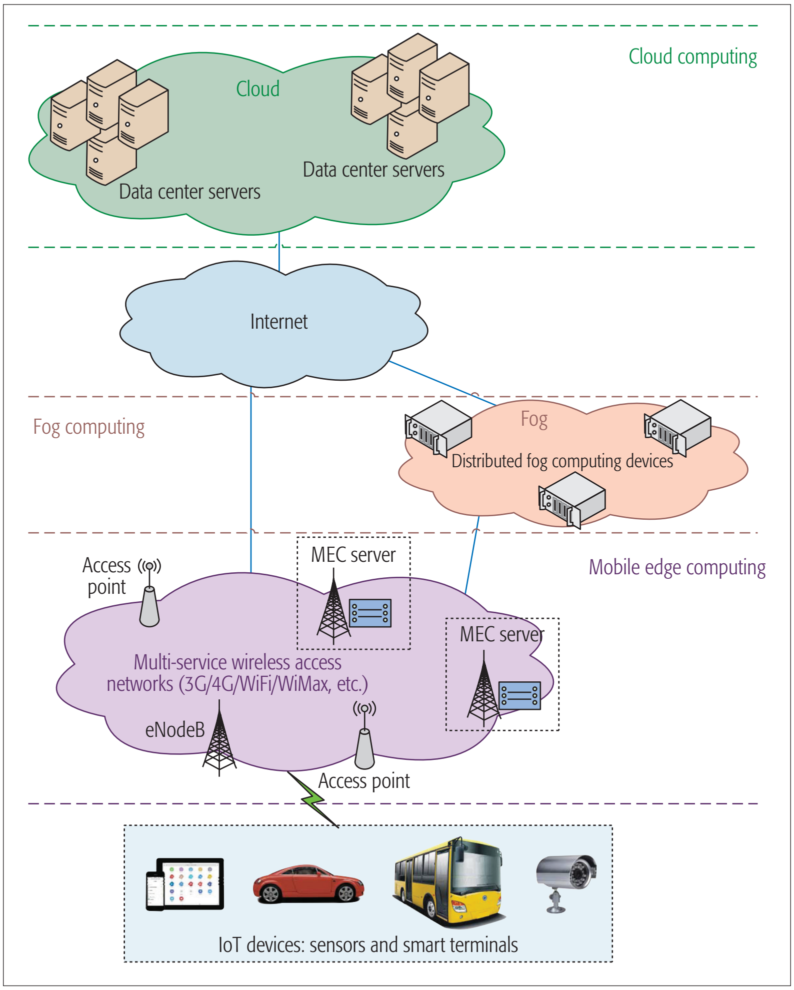

Figure 3. A typical architecture of combined cloud computing, fog computing, and mobile edge computing.

Cloud/fog computing and MEC can be widely used in the Internet of Things (IoT) [10]. A typical architecture of combined cloud/fog computing and MEC in the IoT scenario is shown in Fig. 3. Users can use cloud/fog computing and MEC depending on their situation. Energy-efficient sensors or data acquisition devices acquire and deliver the information to servers with computing and storage capacities to process.

Software Defined Networking, Caching and Computing for Next Generation Wireless Networks

In this section, we propose a framework that integrates networking, caching and computing for next generation wireless networks with heterogeneous access methods. We name this framework software defined networking, caching, and computing. The motivations, design, and implementation details of this framework are described as follows.

Motivations

With the popularity of smart mobile devices, wireless networks have developed rapidly, and a variety of wireless access technologies (e.g., cellular networks and WLANs) have been deployed. While these heterogeneous access methods are complementary to each other, there is a tendency to integrate these access technologies in next generation wireless networks. However, facing the heterogeneous access methods, network operators have difficulty managing and controlling them uniformly [14]. To tackle this issue, software defined wireless networks have been proposed to enable direct programmability of wireless network controls and abstraction of the underlying infrastructure for wireless applications, with improved energy efficiency and great flexibility in wireless network management and configuration [14].

Moreover, different from traditional wireless networks that only have networking resource, recent studies also consider developing caching and computing services in the underlying infrastructure, which will be beneficial for content retrieval and data processing for different applications. From the application's point of view, network, cache, and compute are underlying resources enabling upper layer applications. However, with a large number of underlying resources distributed in different heterogeneous wireless networks, how to manage and control these resources becomes a problem worthy of study to improve the performance of the upper layer applications.

Most existing works on wireless networks consider networking, caching, and computing separately, which could result in suboptimal performance. Therefore, it is desirable to have an integrated framework that enables the network, cache, and computing to be abstracted as a common resource pool for the upper applications. In addition, this framework could enable the resources to be scheduled based on the requirements of different applications, no matter what kind of radio access methods is used.

Therefore, in this article, we propose a novel framework that integrates networking, caching, and computing in a systematic way to naturally support data retrieval and computing services for next generation green wireless networks. Based on the programmable control principle originating in SDN, we incorporate the ideas of information centricity originating from ICN. This integrated framework can enable dynamic orchestration of networking, caching, and computing resources to meet the requirements of different applications.

Framework Design of Software Defined Networking, Caching, and Computing

Integrating networking, caching, and computing resources for next generation green wireless networks is not trivial. Such integration of heterogeneous wireless networks needs modifications of several aspects. The first one is from the aspect of communication protocols. As there are different kinds of access methods, different access methods have different communication protocols. If these three kinds of resources are integrated in heterogeneous wireless networks, the communication protocols for adapting the integration into every type of wireless network could be reconsidered. The second one is from the aspect of communication devices. The reconsideration of communication protocols leads to

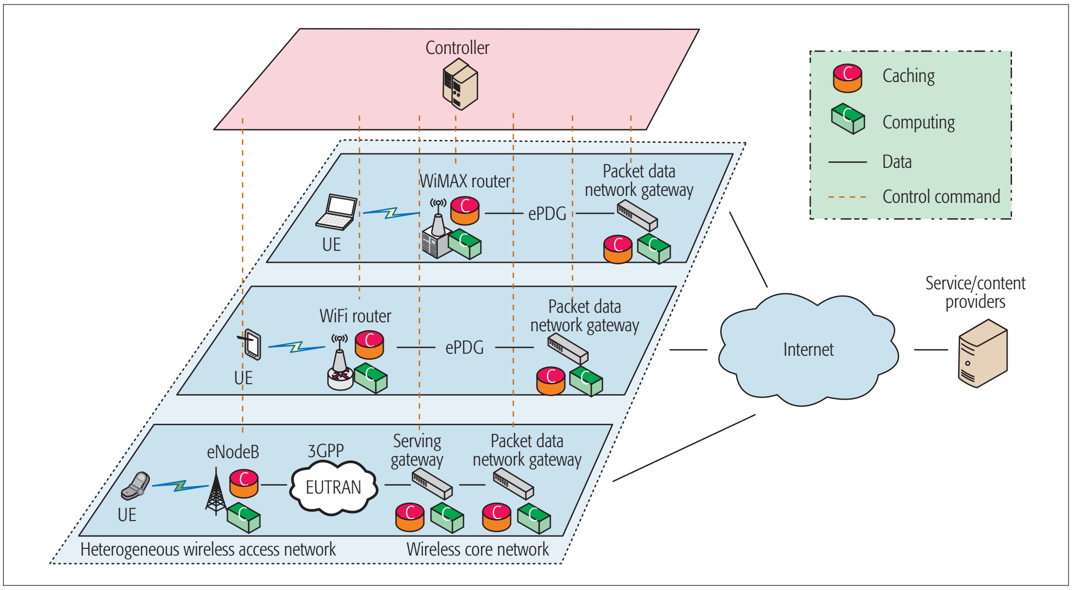

Figure 4. A framework of software defined networking, caching, and computing for next generation green wireless networks.

Integrating computing resources is very desirable for processing the massive content or data, reducing transmision delay greatly and guaranteeing the timeliness of applications. Thus, wireless networks not only provide essential communication capability, but also caching capability and computing capability.

change of communication interfaces. The interfaces should support the interaction among the integrated resources. It means that there is a redesign of the device. The third one is from the aspect of incentives. More modifications will cause more expenses, and take a long time to complete the development. To make our integration more flexible and achievable, we adopt a software-defined approach to integrate networking, caching, and computing resources for next generation green wireless networks with heterogeneous access methods.

In this article, we choose three types of wireless access methods as examples: cellular networks, WLANs, and WiMAX networks. We equip every network node with caching and computing capabilities, and place a controller in the green wireless networks. This controller can manage the topology of the whole wireless network and conduct the packet forwarding strategies. The architecture of the proposed framework is shown in Fig. 4. Multiple user equipments (UEs) respectively access the wireless access points (APs) (e.g., eNodeB, WiFi router and WiMAX router). The controller could manage all the nodes equipped with caching and computing capacities. The caching resource is used for caching massive contents including the data collected by terminals and popular contents requested by users. Therefore, once users want to access contents or some applications want to analyze and process the data, the contents maybe hit from the in-network caching of wireless access networks or wireless core networks. This could be easy for users or applications to access contents, alleviating traffic redundancy and in-network burden. In addition, integrating computing resources is very desirable for processing the massive content or data, reducing transmision delay greatly and guaranteeing the timeliness of applications. Thus, wireless networks provide not only essential communication capability, but also caching capability and computing capability.

Implementation Details of Software Defined Networking, Caching, and Computing

As we equip every in-network node with caching and computing capability, how to use the software defined approach to implement our system is urgent to be solved. Here we give an example implementation of our proposed framework, as shown in Fig. 5. There are some network managers who mainly administrate the controller through the program in the network applications layer, to implement tasks or deploy policies ordered from managers, such as traffic engineering or in-network computing and caching. In other words, these network applications send requests to the logical controller, and the specific behaviors and rules are designated by the control plane for the data plane. The goals of these applications could be reducing the energy consumption, enhancing QoS, improving the utilization of network resources, and achieving other optimization objectives. Therefore, these network applications could include routing algorithms, intrusion detection systems, load balancing, traffic engineering, developing ICN and MEC, and so on. The northbound interface is used for implementing network control and operation logic provided by the set of applications. Particularly, the policies defined by the applications will be translated to another kind of instructions presented below, to program the behavior of the underlying infrastructure.

The control plane can provide some basic services (e.g., monitoring network state, generating network configuration according to the policies of network applications, discovering devices, managing network topology) and important abstraction functions for the underlying infrastructure, including terminal hypervisor, switches hypervisor, network hypervisor, topology hypervisor, and so on. All the infrastructure could be abstracted as isolated resource slices. These hypervisors could manage the resource slices dynamically, realizing on-demand resource allocation. The terminal hypervisor is responsible for managing

The northbound interface is used for implementing network control and operation logic provided by the set of applications. Particularly, the policies defined by the applications will be translated to another kind of instructions presented below, to program the behavior of the underlying infrastructure.

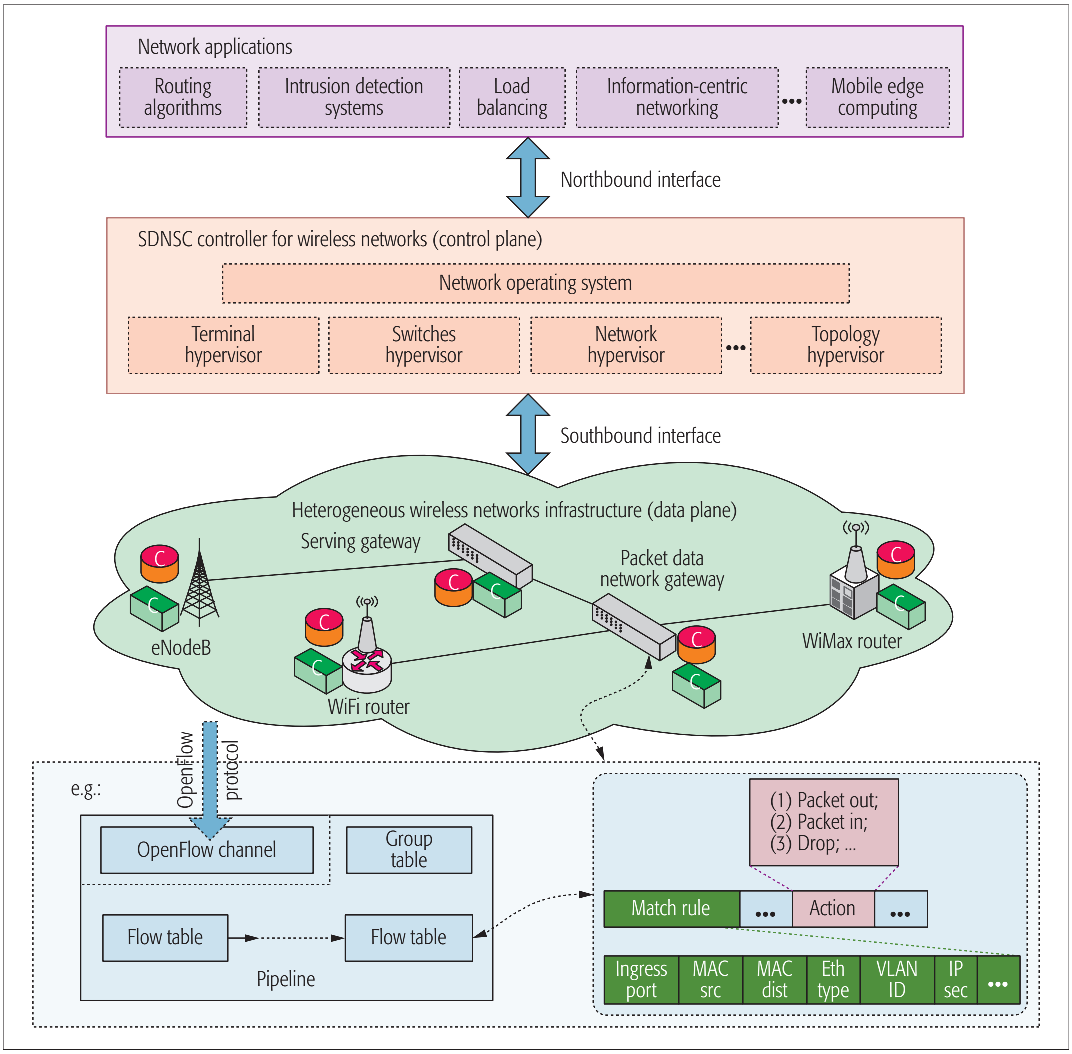

Figure 5. Implementation of software defined networking, caching and computing.

edge nodes. The switch hypervisor mainly implements and administrates the communication between switches and controller. The network hypervisor is used to monitor the networking status, such as congestion. The topology hypervisor masters all the physical nodes, links, and ports through regular monitoring. These hypervisors will map the abstracted resource slices to the physical infrastructure. Based on the information mastered by these hypervisors, the controller could implement some operations or strategies from the network applications layer, and ensure the isolation. Furthermore, the controller could guide packet forwarding of the devices in data plane, as well as perform the commands of communicating, computing, and accessing according to these information lists.

The intelligence shared between the control plane and the data plane is through the southbound interface, which enables the heterogeneous wireless communication devices to be programmed by the controller dynamically. In this article, we adopt the widely used OpenFlow protocol as the representative communication protocol of the southbound interface, which provides event-based messages, flow statistics, and packet-in messages to the controller. According to the topology information and network applications, the controller defines how packets should be handled and sends them to the flow tables. If the devices receive packets, they look up in the first table until executing the corresponding packet out action (forwarding packets to the next switch or destination) or packet in action (encapsulating the packet and sending to the controller to handle) or dropping (no rule is found), as illustrated in Fig. 5. There could be some other optional actions and detailed matching rules, and we will not describe them in detail here. With the management of the controller and the execution of devices, the packets are transmitted, cached, and computed in the green wireless networks.

Simulation Results and Discussions

In this section, we present some simulation results to show the effectiveness of our proposed software defined networking, caching, and computing for next generation green wireless networks. The proposed framework can decrease latency and save energy by jointly considering all the three resources including network, cache, and computing together in the heterogeneous wireless access networks and core networks.

In the simulations, we consider one eNodeB, one WiFi router, and one WiMAX router, each of which has 10 active users randomly distrib-

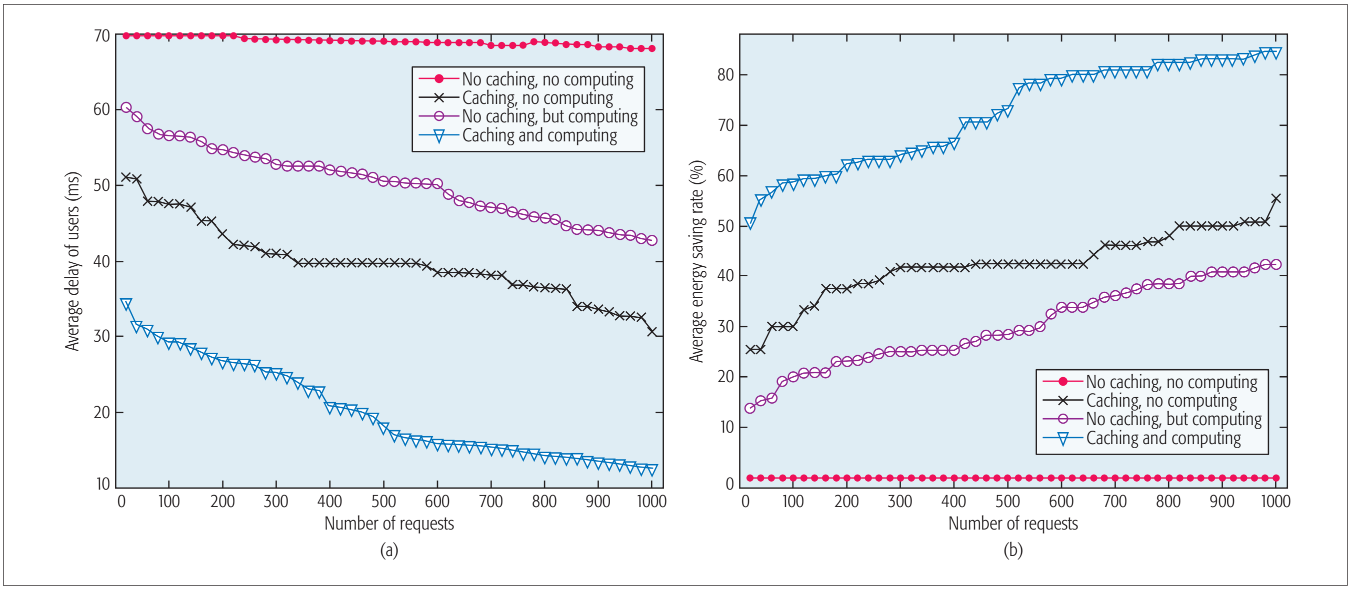

Figure 6. Performance evaluation of software defined networking, caching, and computing for green wireless networks: a) average delay of users' requests response; b) average energy saving rate (AESR).

uted. There is a server connected through the Internet with five routers, which provides caching and computing capacity on the cloud. There are 100 popular contents in the server that users request. Assume that there are totally 100 popular contents requested by all 30 active users from heterogeneous access networks, and there are totally 1000 requests for the popular contents, caching, and computing from all 30 users randomly. The link latency is assumed to be random, ranging from 10 to 20 ms per transmission hop, and the average hops count is 3 in the wired network. Here, we deploy the same cache and compute in every node of wireless networks, every cache size is 20 percent of the total content size (the caching delay is neglected), and the computational rate of the CPU F_T is set as 10^{10} cycles/s. We carry out the trace-driven simulation to evaluate the performance with the number of requests increasing, and consider the average response delay of users' requests and the average energy saving rate (AESR) as our performance metrics. Here, AESR is defined as the average ratio between the saved energy by using different resources and the consumed energy without using caching and computing resource.

The energy consumption model includes transmitting energy E_{tr}, caching energy E_{ca}, and computing energy E_{co}. The transmitting energy of user i for request k is expressed as $E_{tr}^{i,k} = (h_{i,k}P_l + 2h_{i,k}P_n)s_k$, the caching energy of user i for request k is expressed as $E_{ca}^{i,k} = W_{ca}s_kX_k^{ca}t_{ca}$, and the computing energy of user i for request k is expressed as $E_{co}^{i,k} = W_{co}s_kX_k^{co}t_{co}$. Here $h_{i,k}$ represents the hops from user i to the destination where request k can be served, P_l represents the energy density of a link (Joules per bit), P_n represents the energy density of a node to process and forward a request in wireless networks or wired networks (Joules per bit), s_k represents the size of the content or service request k wants (bit), W_{ca} represents the caching power efficiency (Watts per bit), X_k^{ca} represents whether request k is served with cache or not, t_{ca} represents the caching time for request k, W_{co} represents the computing power efficiency (Watts per bit), X_k^{co} represents whether request k is served with compute or not, and t_{co} represents the computing time for request k . Therefore, the energy consumption can be expressed as $\Sigma_{AllUsers} \Sigma_{AllRequests} (E_{tr}^{i,k} + E_{ca}^{i,k} + E_{co}^{i,k})$. Referring to [15], we set our parameters as $P_l = 0.15 \times 10^{-8}$ J/bit, $P_n = 2 \times 10^{-8}$ J/bit, $W_{ca} = 10^{-9}$ W/bit, and $W_{co} = 2.5 \times 10^{-9}$ W/bit.

Figure 6 shows the performance of different schemes. We can observe that the traditional wireless networks that only have networking resources but no caching and computing resources have higher average delay and lower AESR compared to other schemes. Enhancing the traditional wireless networks with caching resources or computing resources can improve the system performance metrics. As there is a portion of requests for popular contents, the popular content requested before is cached in all the nodes of wireless networks, and the same request for the popular content can be accessed even at the first hop. Thus, this avoids redundant transmission traffic to reduce the delay and save the energy of using networking resource. Meanwhile, computing capability depends on F_T, and all the nodes in wireless networks can compute the tasks collaboratively to decrease the computation in the remote cloud. Our proposed framework integrates all three resources, and exhibits the best and most stable performance, especially when the distribution of all the requests comes to the steady state. Due to the integration of both caching service and computing service in wireless networks, our proposed scheme can enable joint optimization to meet the demands of users and the requesting services. With the advances in caching techniques and computing techniques, these resources can achieve better performance at a reasonable price. Since our approach can provide users a good delay experience and AESR, it can enable next generation green wireless networks.

As we jointly consider networking, caching and computing techniques in our proposed framework, it is not trivial to develop this framework in practice. It is possible that Internet service providers (ISPs) will take the responsibility to develop this framework due to the improved users experience and energy efficiency. Nevertheless, it is a significant challenge for ISPs to develop this framework.

Open Research Challenges

Despite the potential vision of software defined networking, caching, and computing, there are some research challenges that need to be overcome. In this section, we discuss some of these challenges.

Scalability

In the proposed framework, we use the software defined approach to centrally manage and control networking, caching, and computing resources. This means the controller in the control plane should supervise all the devices and resources in the data plane. Since there are various access devices, gateway devices, and network nodes in heterogeneous wireless networks, the controller has to maintain a large central database. Therefore, the performance of the controller could be degraded due to the the frequent and rapid flow table update requests as well as caching and computing. In order to promote large-scale development of our scheme in real wireless networks, it is desirable to design a controller that can handle larger flow tables. Further research is needed on the design of a scalable controller in the proposed framework.

Networking/Caching/Computing Resource Allocation Strategies

In traditional SDN, the optimization problem mainly focuses on the networking resources. In contrast, in the proposed framework, there are networking, caching, and computing resources. These resources should be carefully managed to maximize the resource utilization and enhance the user experience. Therefore, it is important to design the networking/caching/computing resource allocation strategies to make a tradeoff between the deployment and operation costs (e.g., energy consumption) and performance benefits (e.g., decreasing latency). Research to study the interrelationship and mutual impacts among different factors are needed, such as the relationship among the number/locations of resource nodes, energy consumption, traffic, latency, and so on. In addition, there will be different strategies corresponding to different optimization objectives.

Security

The controller in the proposed framework could be attacked, resulting in a single point of failure in the system. Once an attacker acquires the permissions of the controller, the entire wireless network will be exposed to danger. Therefore, it is desirable to build an attack-tolerant system, which is designed by a fault-tolerant design approach and can operate correctly despite the existence of attacks. For instance, an attack-tolerant system may provide services meeting a service level agreement (SLA) even under an attack by triggering automatic mechanisms to regain and recover the compromised services and resources. Other descriptions used for similar themed research include survivability, resilience, trustworthy systems, and autonomic self-healing systems. How to use the software-defined characteristics to realize a tolerant system is a new direction that needs to be addressed by future research efforts.

Cooperation Incentives among Stakeholders

As we jointly consider networking, caching, and computing techniques in our proposed framework, it is nontrivial to develop this framework in practice. It is possible that Internet service providers (ISPs) will take the responsibility to develop this framework due to the improved user experience and energy efficiency. Nevertheless, it is a significant challenge for ISPs to develop this framework. Therefore, it is desirable to design some cooperative incentives among different stakeholders, including ISPs and content providers. It is important to provide scalable and flexible interfaces for these stakeholders to interact and cooperate to achieve attractive benefits.

Conclusions and Future Work

In this article, we review recent advances in networking, caching, and computing. We propose to integrate networking, caching, and computing in a systematic framework for next generation green wireless networks. We develop the architecture of the proposed framework for software defined networking, caching, and computing. We detail its key components of data, control, and management planes. Simulation results have been presented to show that our proposed framework can improve users' experience and energy efficiency. In addition, we discuss some open research challenges, including scalable controller design, networking/caching/computing resources allocation strategies, and security issues. Future work is in progress to address these research challenges.

Acknowledgment

We thank the reviewers for their detailed reviews and constructive comments, which have helped to improve the quality of this article.

References

[1] J. Wu *et al.*, "Green Communications and Computing Networks [Series Editorial]," *IEEE Commun. Mag.*, vol. 54, no. 5, May 2016, pp. 106–07.

[2] S. Bu *et al.*, "When the Smart Grid Meets Energy-Efficient Communications: Green Wireless Cellular Networks Powered by the Smart Grid," *IEEE Trans. Wireless Commun.*, vol. 11, Aug. 2012, pp. 3014–24.

[3] H. Hu *et al.*, "Software Defined Wireless Networks: Part 1 [Guest Editorial]," *IEEE Commun. Mag.*, vol. 53, no. 11, Nov. 2015, pp. 108–09.

[4] K. Wang *et al.*, "Virtual Resource Allocation in Software-Defined Informationcentric Cellular Networks with Device-to-Device Communications and Imperfect CSI," *IEEE Trans. Vehic. Tech.*, vol. PP, no. 99, Feb. 2016, p. 1.

[5] S. Zhou *et al.*, "Software-defined Hyper-Cellular Architecture for Green and Elastic Wireless Access," *IEEE Commun. Mag.*, vol. 54, no. 1, Jan. 2016, pp. 12–19.

[6] C. Fang *et al.*, "A Survey of Green Information-Centric Networking: Research Issues and Challenges," *IEEE Commun. Surveys Tutorials*, vol. 17, no. 3, 3rd qtr., 2015, pp. 1455–72.

[7] C. Liang, F. R. Yu, and X. Zhang, "Information-Centric Network Function Virtualization over 5G Mobile Wireless Networks," *IEEE Network*, vol. 29, no. 3, May, 2015, pp. 68–74.

[8] M. Armbrust *et al.*, "A View of Cloud Computing," *Commun. ACM*, vol. 53, no. 4, Apr. 2010, pp. 50–58.

[9] N. Saxena, A. Roy, and H. Kim, "Traffic-Aware Cloud RAN: A Key for Green 5G Networks," *IEEE JSAC*, vol. 34, no. 4, Apr. 2016, pp. 1010–21.

[10] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the Suitability of fog Computing in the Context of Internet of Things," *IEEE Trans. Cloud Computing*, vol. pp, no. 99, Oct. 2015, p. 1.

[11] ETSI, "Mobile-Edge Computing – Introductory Technical White Paper," Sept. 2014.

[12] L. Zhang *et al.*, "Named Data Networking," *ACM SIGCOMM Comp. Commun. Rev.*, vol. 44, no. 3, July 2014, pp. 66–73.

[13] G. Pallis, "Cloud Computing: The New Frontier of Internet Computing," *IEEE Internet Computing*, vol. 14, no. 5, Sept. 2010, pp. 70–73.

[14] T. Chen *et al.*, "Software Defined Mobile Networks: Concept, Survey, and Research Directions," *IEEE Commun. Mag.*, vol. 53, no. 11, Nov. 2015, pp. 126–33.

[15] C. Fang *et al.*, "An Energy-Efficient Distributed In-Network Caching Scheme for Green Content-Centric Networks," *Computer Networks*, vol. 78, Feb. 2015, pp. 119–29.

BIOGRAPHIES

RU HUO (EthelyHuo@gmail.com) received her B.S. degree in information engineering from Harbin Engineering University, Heilongjiang, China, in 2011. She is currently working toward her Ph.D. degree in the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications (BUPT), China. From September 2015 to September 2016, she studied at the University of British Columbia (UBC), Vancouver, Canada, as a visiting Ph.D. student. Her current research interests include wireless networks, software-defined networking, information-centric networking, and resource management and allocation.

FEI RICHARD YU (richard.yu@carleton.ca) received his Ph.D. degree in electrical engineering from UBC in 2003. From 2002 to 2006, he was with Ericsson, Lund, Sweden, and a start-up in California. He joined Carleton University in 2007, where he is currently a professor. He received the IEEE Outstanding Service Award in 2016, IEEE Outstanding Leadership Award in 2013, Carleton Research Achievement Award in 2012, Ontario Early Researcher Award (formerly Premier's Research Excellence Award) in 2011, Excellent Contribution Award at IEEE/IFIP TrustCom 2010, Leadership Opportunity Fund Award from the Canada Foundation of Innovation in 2009, and Best Paper Awards at IEEE ICC 2014, IEEE GLOBECOM 2012, IEEE/IFIP TrustCom 2009, and the International Conference on Networking 2005. His research interests include cross-layer/cross-system design, security, green IT, and QoS provisioning in wireless-based systems. He serves on the Editorial Boards of several journals, and is a Co-Editor-in-Chief for *Ad Hoc & Sensor Wireless Networks*, and is a Lead Series Editor for *IEEE Transactions on Vehicular Technology* and *IEEE Communications Surveys & Tutorials*. He has served as a Technical Program Committee (TPC) Co-Chair of numerous conferences.

TAO HUANG (htao@bupt.edu.cn) received his B.S. degree in communication engineering from Nankai University, Tianjin, China, in 2002, and his M.S. and Ph.D. degrees in communication and information systems from Beijing University of Posts and Telecommunications in 2004 and 2007, respectively. He is currently an associate professor at Beijing University of Posts and Telecommunications. His current research interests include network architecture and software-defined networking.

RENCHAO XIE (Renchao_xie@bupt.edu.cn) received his Ph.D. degree from the School of Information and Communication Engineering, BUPT, in 2012. From July 2012 to September 2014, he worked as a postdoctoral researcher at China Unicom. From November 2010 to November 2011, he visited Carleton University as a visiting scholar. He is an associate professor at BUPT. His current research interests include content delivery network, information-centric networking, and 5G networks. He has published more than 30 journal and conference papers. He has served on the Technical Program Committees (TPCs) of Chinacom 2016 and the 2012 IEEE Vehicular Technology Conference (VTC)-Spring. He has also served for several journals and conferences as a reviewer, including *IEEE Transactions on Communications*, *ACM/Springer Wireless Networks*, the *EURASIP Journal on Wireless Communications and Networking*,(Wiley) *Wireless Communications and Mobile Computing*, *IEEE Communications Letters*, 2011 IEEE GLOBECOM, and so on.

JIANG LIU (liujiang@bupt.edu.cn) received his B.S. degree in electronics engineering from Beijing Institute of Technology, China, in 2005, his M.S. degree in communication and information systems from Zhengzhou University, China, in 2009, and his Ph.D. degree from BUPT in 2012. He is currently an assistant professor at BUPT. His current research interests include network architecture, network virtualization, software-defined networking, information-centric networking, and tools and platforms for networking research and teaching.

VICTOR C. M. LEUNG [S'75, M'89, SM'97, F'03] (vleung@ece.uvc.ca) is a professor of electrical and computer engineering and holder of the TELUS Mobility Research Chair at UBC. His research is in the areas of wireless networks and mobile systems. He has co-authored more than 800 technical papers in archival journals and refereed conference proceedings, several of which have won best-paper awards. He is a Fellow of the Royal Society of Canada, the Canadian Academy of Engineering, and the Engineering Institute of Canada. He serves on the Editorial Boards of *IEEE JSAC-SGCN*, *IEEE Wireless Communications Letters*, and several other journals. He has provided leadership to the technical program committees and organizing committees of numerous international conferences. He was the recipient of the 1977 APEBC Gold Medal, NSERC Postgraduate Scholarships from 1977 to 1981, a 2012 UBC Killam Research Prize, and an IEEE Vancouver Section Centennial Award.

YUNJIE LIU (liuyj@chinaunicom.cn) received his B.S. degree in technical physics from Peking University, Beijing, China, in 1968. He is currently the Academician of the China Academy of Engineering, Chief of the Science and Technology Committee of China Unicom, and Dean of the School of Information and Communication Engineering, BUPT. His research interests include next generation networks, and network architecture and management.