Network Functions Virtualization: Challenges and Opportunities for Innovations

Bo Han, Vijay Gopalakrishnan, Lusheng Ji and Seungjoon Lee
AT&T Labs Research
Bedminster, NJ 07921, USA

***Abstract*—Network Functions Virtualization (NFV) was recently proposed to improve the flexibility of network service provisioning and reduce the time to market of new services. By leveraging virtualization technologies and commercial off-the-shelf programmable hardware, such as general purpose servers, storage and switches, NFV decouples the software implementation of network functions from the underlying hardware. As an emerging technology, NFV brings several challenges to network operators, such as the guarantee of network performance for virtual appliances, their dynamic instantiation and migration, and their efficient placement. In this article, we provide a brief overview of NFV, explain its requirements and architectural framework, present several use cases and discuss the challenges and future directions in this burgeoning research area.**

***Index Terms*—Network functions virtualization, virtual network appliance, dynamic service provisioning, Cloud RAN, Cloud EPC.**

I. Introduction

It is well-known that bringing in new services into today's networks is becoming increasingly difficult due to the proprietary nature of existing hardware appliances, the cost of offering the space and energy for a variety of middle-boxes, and the lack of skilled professionals to integrate and maintain these services. Network Functions Virtualization was recently proposed to alleviate these problems, along with other emerging technologies, such as Software Defined Networking (SDN) and cloud computing.[1]

NFV transforms how network operators architect their infrastructure by leveraging the full-blown virtualization technology to separate software instance from hardware platform and by decoupling functionality from location for faster networking service provisioning [4]. Essentially, NFV implements network functions through software virtualization techniques and runs them on commodity hardware (i.e., industry standard servers, storage and switches), as shown in Figure 1. These virtual appliances can be instantiated on demand without the installation of new equipment. For example, network operators may run an open-source software-based firewall in a Virtual Machine (VM) on an x86 platform. Recent trials have demonstrated that it is feasible to implement network functions on general purpose processor based platforms, for example, for physical layer signal processing [3].

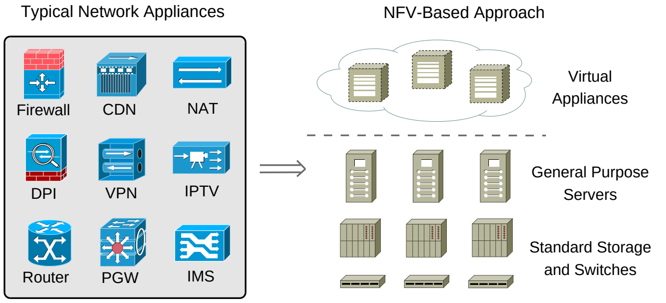

Fig. 1: From dedicated hardware-based appliances for network services, such as firewall, Content Delivery Network (CDN), Network Address Translation (NAT), Deep Packet Inspection (DPI), Virtual Private Network (VPN), IPTV, router, Packet Data Network Gateway (PDN-GW or PGW) and IP Multimedia Subsystem (IMS), to software-based NFV solutions.

As an innovative step towards implementing a lower cost agile network infrastructure, NFV can potentially bring several benefits to network carriers, dramatically changing the landscape of the telecommunications industry. It may reduce capital investment and energy consumption by consolidating networking appliances [2], decrease the time to market of a new service by changing the typical innovation cycle of network operators (e.g., through software-based service deployment), and rapidly introduce targeted and tailored services based on customer needs, just to list a few.

Along with the benefits of NFV, network operators also face several technical challenges when deploying virtual appliances. A frequently raised issue about Virtualized Network Functions (VNFs)[2] is their network performance. Previous work has shown that virtualization may lead to abnormal latency variations and significant throughput instability even when the underlying network is only lightly utilized [14]. Therefore, ensuring that network performance remains at least as good as that of purpose-built hardware implementations will be one of the key challenges in realizing NFV. Besides the network performance issue, another major problem network carriers are confronted with is how to smoothly migrate from the existing network infrastructure to NFV-based solutions, given the former's large scale and tight coupling among its components. Moreover, the separation of functionality from location also casts the problem of how to efficiently place

[1] We discuss the relationship between NFV, SDN and cloud computing in Section III.

[2] A VNF is the software instance in NFV that consists of some number or portion of VMs running different processes for a network function.

the virtual appliances and dynamically instantiate them on demand.

These facts all impose the need to investigate open research issues brought by NFV in order to ensure its successful adoption. However, there are very limited prior efforts in the literature to offer an overview of aspects to be considered and issues to be addressed when adopting NFV. Our goal is to bridge this gap by identifying critical research challenges involved in the evolution towards NFV.

In this article, we first present the key technical requirements of NFV (Section II). We then introduce its architectural framework (Section III) and standardization activities (Section IV). We also describe several use cases of NFV, including the virtualization of mobile base station, cellular core network and home network (Section V). Finally, we discuss the open research issues and point out future directions for NFV, focusing on the network performance of virtualized appliances, their efficient instantiation, placement and migration (Section VI).

II. Technical Requirements

In this section, we summarize the technical requirements when implementing virtualized network functions, including the network performance of VNFs, their manageability, reliability, security, and the coexistence with existing platforms.

A. Performance

When talking about software-based implementation of network functions through virtualization technologies on general purpose servers, the first question we may ask is whether the performance, such as throughput and latency, will be affected. The per-instance capacity of a VNF may be less than the corresponding physical version on dedicated hardware.

Although it is hard to completely avoid the performance degradation, we should keep it as small as possible while not impacting the portability of VNFs on heterogeneous hardware platforms. One possible solution is to leverage clustered VNF instances and modern software technologies, such as Linux New API (NAPI)[3] and Intel's Data Plane Development Kit (DPDK)[4]. When deploying VNF instances, we need to design efficient algorithms to split network load across a number of distributed and clustered VMs while keeping the latency requirement in mind. Moreover, the underlying NFV infrastructure should be able to gather network performance information at different levels (e.g., hypervisor, virtual switch and network adapter). We discuss the research challenges related to NFV performance in Section VI.

A bottom line when designing NFV systems is that we should understand the maximum achievable performance of the underlying programmable hardware platforms. Based on this information, we can make the proper design decisions.

B. Manageability

The NFV infrastructure should be able to instantiate VNFs in the right locations at the right time, dynamically allocate and scale hardware resources for them and interconnect them to achieve service chaining[5]. This flexibility of service provisioning poses new requirements to manage both virtual and legacy appliances. The manageability in NFV is quite different from that in data center networking where the hardware resources are almost equivalent, which makes their coordination easier. However, the cost and value of resources may vary significantly between network points of presence and customers' premises. The management functionality should take the variations into account and optimize resource usage across the wide area.

Since service unavailability is typically thought unacceptable, network carriers usually over-provision their services [5] and thus the utilization of resources allocated to these services is normally low, due to the offered redundancy for unexpected traffic increase or service element failure. If we share cloud resources across multiple services and their failure modes are independent, we can leverage the pool of spare resources to provide the necessary redundancy across them and dynamically create VNFs to appropriately handle traffic increase or failure. In addition, NFV can potentially improve resource utilization through the elasticity feature of cloud computing, for example, by consolidating the workload on a small number of servers during overnight hours and turning the rest off (or using them for services such as online gaming). The management functionality should be able to support the sharing of spare resources and the elastic provisioning of network services effectively.

Although NFV may make planned maintenance relatively easy [15], it presents new requirements for service quality management. Network operators should be able to obtain and process actionable information from various service impacting events, determine and correlate faults and recover from them, by monitoring compute, storage and network resource usage during the life cycle of a VNF. Since VNFs can be dynamically created/migrated, it brings an additional dimension of complexity in terms of keeping track of where a given VNF is running. Moreover, a VNF can behave erratically even if the underlying infrastructure is running fine, which makes the detection of issues non-trivial.

C. Reliability and Stability

Reliability is an important requirement for network operators when offering specific services (e.g., voice call and video on demand), no matter through physical or virtual network appliances. Carriers need to guarantee that service reliability and service level agreement are not affected when evolving to NFV. Purpose-built network equipment can provide the traditional five-nines reliability in telecommunications industry. To

[3] http://www.linuxfoundation.org/collaborate/workgroups/networking/napi

[4] http://dpdk.org/

[5] Service chaining describes a method for the delivery of network services based on their function associations and enables the ordering and topological independence of the network functions.

meet the same reliability requirement, NFV needs to build the resilience into software when moving to error-prone hardware platforms. Moreover, as we mentioned above, the elasticity of service provisioning may require the consolidation and migration of VNFs based on traffic load and user demand. All these operations create new points of failure that should be handled automatically.

In addition, ensuring service stability poses another challenge to NFV, especially when reconfiguring or relocating a large number of software-based virtual appliances from different vendors and running on different hypervisors. Network operators should be able to move VNF components from one hardware platform onto a different platform while still satisfying the service continuity requirement. They also need to specify the values of several key performance indicators to achieve service stability and continuity, including maximum non-intentional packet loss rate and call/session drop rate, maximum per-flow delay and latency variation, and maximum time to detect and recover from failures.

D. Security

When deploying virtualized network functions, operators need to make sure that the security features of their network will not be affected. NFV may bring in new security concerns along with its benefits. The virtual appliances may run in data centers that are not owned by network operators directly. These virtualized network functions may even be outsourced to third parties [10]. The introduction of new elements, such as orchestrators and hypervisors, may generate additional security vulnerabilities which increase the load of intrusion detection systems. The underlying shared networking and storage can also introduce new security threats, for example, when running a software router in a VM that shares the physical resources with other network appliances. Moreover, these software-based components may be offered by different vendors, potentially creating security holes due to integration complexity. All these changes require us to rethink security issues when designing and building NFV systems.

NFV can also enhance the security level of a wide spectrum of networking services. The creation, management and adjustment of security zones become easier, since network operators can automate the placement of virtualized firewalls, create dedicated software firewalls on-demand to protect specific network domains, and update the security rules of deployed firewalls remotely.

E. Interoperability and Compatibility

Another key issue for NFV is to design standard interfaces between not only a range of virtual appliances but also these virtualized implementations and legacy equipment. As one of the goals of NFV is to promote openness, network carriers may need to integrate and operate servers, hypervisors and virtual appliances from different vendors in a multi-tenant NFV environment. Their seamless integration requires a unified interface to facilitate the interoperability among them.

Network operators should support a smooth migration path from proprietary physical appliances to open standard based virtual ones, since they may not be able to update all their existing services and equipment to NFV-based solutions. The developed NFV solutions need to be compatible with existing Operation and Business Support Systems (OSS/BSS) and Element and Network Management Systems (EMS/NMS), and work in a hybrid environment with both physical and virtual network functions.

III. Design and Architectural Framework

Virtualization provides us the opportunity for a flexible software design. Existing networking services are supported by diverse network functions that are connected in a static way. NFV enables additional *dynamic* schemes to create and manage network functions. Its key concept is the VNF forwarding graph which simplifies the service chain provisioning by quickly and inexpensively creating, modifying and removing service chains. On one hand, we can compose several VNFs together to reduce management complexity, for instance, by merging the serving gateway (SGW) and PGW of a 4G core network into a single box. On the other hand, we can decompose a VNF into smaller functional blocks for reusability and faster response time. However, we note that the actual carrier-grade deployment of VNF instances should be transparent to end-to-end services.

Compared with the current practice, NFV introduces the following three major differences [12]:

- *Separation of software from hardware*: This separation enables the software to evolve independently from the hardware, and vice versa.
- *Flexible deployment of network functions*: NFV can automatically deploy network-function software on a pool of hardware resources which may run different functions at different times in different data centers.
- *Dynamic service provisioning*: Network operators can scale the NFV performance dynamically and on a grow-as-you-need basis with fine granularity control based on the current network conditions.

We illustrate the high-level architectural framework of NFV in Figure 2. Its four major functional blocks are the orchestrator, VNF manager, virtualization layer and virtualized infrastructure manager. The *orchestrator* is responsible for the management and orchestration of software resources and the virtualized hardware infrastructure to realize networking services. The *VNF manager* is in charge of the instantiation, scaling, termination and update events during the lifecycle of a VNF, and supports zero-touch automation. The *virtualization layer* abstracts the physical resources and anchors the VNFs to the virtualized infrastructure. It ensures that the VNF lifecycle is independent of the underlying hardware platforms by offering standardized interfaces. This type of functionality is typically provided in the forms of VMs and their hypervisors. The *virtualized infrastructure manager* is used to virtualize and manage the configurable compute, network and storage resources and control their interaction with VNFs. It allocates

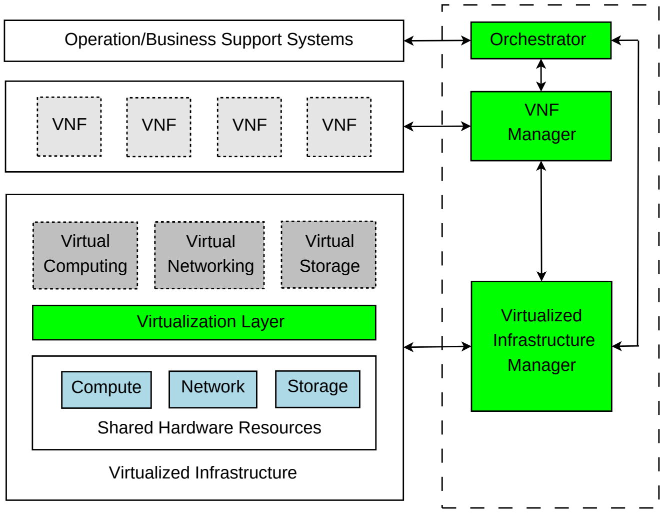

Fig. 2: NFV architectural framework [12].

VMs onto hypervisors and manages their network connectivity. It also analyzes the root cause of performance issues and collects information about infrastructure fault and for capacity planning and optimization.

As we can see from this architectural framework, the two major enablers of NFV are industry-standard servers and technologies developed for cloud computing. A common feature of industry-standard servers is that their high volume makes it easy to find interchangeable components inside them with competitive price, compared with network appliances based on bespoke Application Specific Integrated Circuits (ASICs). Using these general purpose servers can also reduce the number of different hardware architectures in operators' networks and prolong the lifecycle of hardware when technologies evolve (e.g., running different software versions on the same platform). Recent developments of cloud computing, such as various hypervisors, OpenStack and Open vSwitch, also make NFV achievable in reality. For example, the cloud management and orchestration schemes enable the automatic instantiation and migration of VMs running specific network services.

NFV is closely related to other emerging technologies, such as SDN. SDN is a networking technology that decouples the control plane from the underlying data plane and consolidates the control functions into a logically centralized controller. NFV and SDN are mutually beneficial, highly complementary to each other, and share the same feature of promoting innovation, creativity, openness and competitiveness. These two solutions can be combined to create greater value. For example, SDN can support NFV to enhance its performance, facilitate its operation and simplify the compatibility with legacy deployments. However, we emphasize that the virtualization and deployment of network functions do not rely on SDN technologies, and vice versa.

IV. Standards Related Activities

European Telecommunications Standards Institute (ETSI) has created an Industry Specification Group (ISG) for NFV to achieve the common architecture required to support virtualized network functions through a consistent approach. This ISG was initiated by several leading telecommunication carriers, including AT&T, BT, China Mobile, Deutsche Telekom, Orange, Telefonica and Verizon. It has quickly attracted broad industry support and had over 150 members and participants by the end of 2013, ranging from network operators to equipment vendors and IT vendors.

The ETSI NFV ISG currently has four working groups: Infrastructure Architecture, Management and Orchestration, Software Architecture and Reliability & Availability; and two expert groups: Security and Performance & Portability. It has also developed a Proof of Concept (PoC) Framework to coordinate multi-vendor PoCs and build the confidence that NFV is a viable technology. Although it is not a standards development organization, it seeks to define the requirements that network operators may adopt and tailor for their commercial deployment. Part of this article (e.g., the architectural framework) is based on the NFV white paper [4] and several related specifications [12], [13] published by this ISG.

V. Use Cases

In this section, we describe several use cases of NFV, including the virtualization of cellular base station, mobile core network and home network. We focus on the problems of existing architecture and the benefits of NFV-based solutions. NFV is applicable to both data plane processing and control plane function. We refer interested readers to the specification of ETSI [13] for more use cases, such as the virtualization of CDN and fixed access network.

A. Virtualization of Cellular Base Station

The Radio Access Network (RAN) of traditional cellular networks is usually composed of stand-alone base stations which process and transmit wireless signal on behalf of mobile phones and forward their data to the core network through backhaul connections. This RAN architecture has several limitations. First, cellular operators provision their base stations to handle the maximum expected network load, but the traffic of a base station is fluctuating over time due to usage pattern and user mobility. Therefore, the processing power of base stations is usually not fully utilized. However, it is impossible to share the processing resources among them because they are geographically dispersed. Second, given the limited spectrum resource, base stations need to reuse radio frequency which makes the planning and optimization of base station deployment hard, especially in urban areas. Third, base stations require their own backhaul transmission equipment, environment surveillance system, cooling system and backup battery, which in turn need large space to host them.

Over the years, the RAN architecture has evolved from the all-in-one base stations to distributed base stations, which separate the radio function unit (a.k.a., Remote Radio Head or RRH) from the digital function unit (a.k.a., BaseBand Unit or BBU). Baseband wireless signal is carried over fiber links

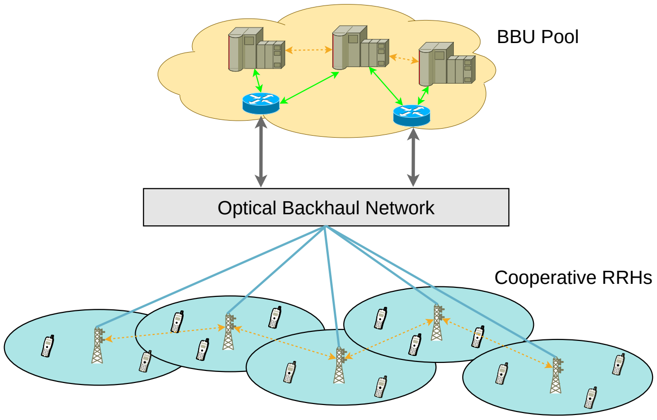

Fig. 3: The Cloud RAN architecture.

between RRH and BBU which also makes their physical separation possible (e.g., by a few kilometers). In this architecture, BBU implements the antenna array system functionality and the physical and MAC layers; while RRH obtains and converts the wireless signal and amplifies the power.

Distributed base stations have paved the way to further evolve the architecture to Cloud RAN by virtualizing BBUs in data centers and thus enabling dynamic service provisioning. Cloud RAN leverages many advanced technologies, including the common public radio interface in wireless communication, the coarse/dense wavelength-division multiplexing in optical communication and the real-time virtualization in cloud computing, as shown in Figure 3. The virtualization target is usually the BBU pool which typically runs in data centers. Moreover, for the traditional RAN architecture, the virtualization target can also be the evolved NodeB (eNodeB) in a 4G network, NodeB in a 3G network, or a legacy base station in a 2G network (e.g., running a part of eNodeBs in VMs).

The decoupling of baseband processing and radio units can potentially bring various benefits. By centralizing and virtualizing the BBUs, Cloud RAN can significantly reduce the operation, computing, energy and real-estate cost for cellular carriers, thanks to easy software/firmware upgrades, fewer site visits and lower site space leasing cost. For example, based on the analysis of real-world data, Bhaumik et al. [2] reported that Cloud RAN can reduce at least 22% computing resources by sharing the processing load among base stations and exploiting the load variations.

Cloud RAN can also enable advanced technologies such as Coordinated MultiPoint (CoMP) in 4G Long Term Evolution (LTE) networks and LTE-Advanced networks. CoMP dynamically coordinates the transmission and reception between user equipment and multiple eNodeBs, and thus improves reception performance, reduces interference levels and better utilizes network resources. It requires joint signal processing for both uplink and downlink data and coordinated scheduling/beamforming among base stations, which can be achieved by the virtualized BBU pool.

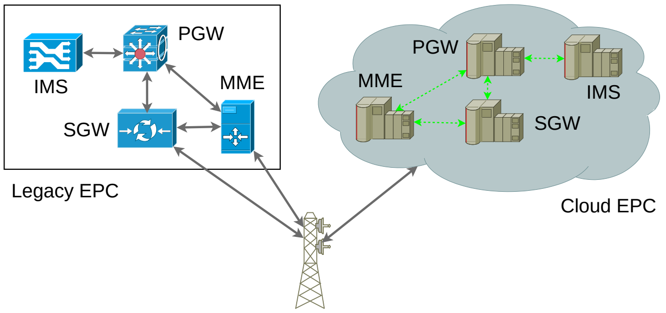

Fig. 4: Virtualization of EPC and its coexistence with legacy EPC.

B. *Virtualization of Mobile Core Network*

Today's mobile core networks suffer from a huge variety of expensive and proprietary equipment, as well as from inflexible hard-state signaling protocols. When a specific function is not available, cellular operators have to replace an existing equipment even if it is still sufficient for most purposes [7], which reveals the difficulty to scale up and down offered services rapidly as required. Moreover, the mobile core network leverages the tunneling mechanism over lower-layer transport protocols to and from a few centralized gateways (PGWs in case of 4G EPC) for the delivery of user data traffic. The same is true for 2G and 3G networks. These long-distance permanent tunnels are very expensive to control and maintain for cellular operators.

Cloud EPC can potentially address these problems by virtualizing the mobile core network to meet changing market requirements. The virtualization targets of EPC include Mobility Management Entity (MME), Home Subscriber Server (HSS), SGW, PGW and Policy and Charging Rules Function (PCRF). To better support Voice over LTE (VoLTE), cellular operators can also virtualize the components of an IMS, including various Call Session Control Functions (CSCFs) such as Proxy-CSCF, Serving-CSCF and Interrogating-CSCF, and Breakout and Media Gateway Control Functions. We illustrate the virtualization of EPC for 4G LTE networks and its coexistence with the legacy EPC in Figure 4. The coexistence is made possible through technologies such as MME pooling. We note that it is possible to virtualize only part of the mobile core network, such as SGW and PGW, and use physical appliance for other components.

By virtualizing the aforementioned network functions, Cloud EPC allows us to move towards a more intelligent, resilient and scalable core architecture. It enables flexible distribution of hardware resources to eliminate performance bottlenecks and rapid launch of innovative services to generate new revenue sources (e.g., machine-to-machine communications). The virtualization of EPC frees distributed network resources from their geographic limitations to ensure service reliability and stability in the event of local resource failure and reduce the Total Cost of Ownership (TCO). It also makes the flexible deployment of SGW and PGW possible, for example, co-locating them with an eNodeB and thus eliminating the

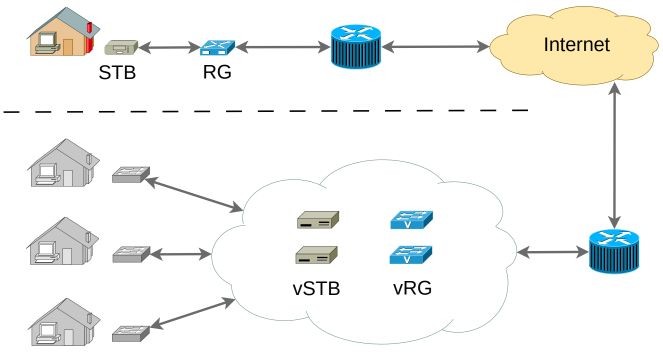

Fig. 5: Virtualization of home network.

long-distance tunnels. With Cloud EPC, cellular carriers can not only expand their current horizontal market business, but also capitalize on previously untouched vertical markets.

C. *Virtualization of Home Network*

Network service providers offer home services through dedicated Customer Premise Equipment (CPE) supported by network-located backend systems. Typical CPE devices include Residential Gateways (RGs) for Internet access and Set-Top Boxes (STBs) for multimedia services. Under this architecture, the delivery of time-shifted IPTV services is known to be complicated, due to the interactive stream control functions (e.g., rewind and fast-forward) [1]. The emerging NFV technology with the availability of high throughput last-mile access facilitates the virtualization of home network and brings down the complexity of IPTV services.

We depict the architecture of virtualized home networks in Figure 5. The virtualization targets are STBs and a range of components of RGs, such as firewall, DHCP server, VPN gateway and NAT router. By moving them to data centers, network and service operators need to provide only low cost devices to customers for physical connectivity with low maintenance requirements, demonstrated by the three gray boxes at the left hand bottom corner of Figure 5. These devices need to provide only the layer 2 functionality for Internet access, as the layer 3 and above functions of RGs are moved into the operators' network. We note that with this virtual architecture, it is possible to share some functionalities of RGs and STBs among customers. The concept of virtualizing home network is not actually new. Multiple-System Operators (MSOs) have been pushing the Cloud Digital Video Recorder (DVR) or Network DVR solutions for several years. Cloud DVR stores the recorded video programs at the MSO's central location (e.g., the video hub office) instead of the consumer's home and relieves the storage requirement on STBs.[6]

This virtualized architecture presents numerous advantages to network operators and end users. First, it reduces the operating expense by avoiding the constant maintenance and updating of the CPE devices and alleviating the call center and product return burdens. Second, it improves the quality of experience by offering near unlimited storage capacity and enabling access to all services and shared content from different locations and multiple devices, such as smartphones and tablets. Third, it allows dynamic service quality management and controlled sharing among user application streams which helps content providers programmatically provision capacity to end users via open APIs [11]. Finally, it introduces new services more smoothly and less cumbersome by minimizing the dependency on the CPE functions.

VI. Research Challenges and Future Directions

In this section, we discuss some of the research challenges and future directions for NFV, including the network performance of virtualization, the placement, instantiation and migration of virtual appliances and the outsourcing of VNFs.

A. *Network Performance of VNF*

The recent effort from the telecommunications industry has been centered on the software virtualization framework (e.g,. management and orchestration). However, it is challenging to offer guaranteed network performance for virtual appliances. Wang and Ng [14] measured the end-to-end networking performance of the Amazon EC2 cloud service. They found that the sharing of processors may lead to very unstable TCP/UDP throughput, fluctuating between zero and 1 Gbps at the tens of milliseconds time granularity, and the delay variations among Amazon EC2 instances can be 100 times larger than most propagation delays which are smaller than 0.2 ms, even when the network is not heavily loaded. The unstable networking characteristics caused by virtualization can obviously affect the performance and deployment of virtual appliances.

As we mentioned in Section II, it may be possible to leverage Linux NAPI and Intel's DPDK to improve the network performance of VNFs. NAPI is a modification of the packet processing framework in Linux device drivers, aiming at improving the performance of high-speed networking. It achieves this goal by disabling some interrupts when the network traffic load is high and switching to polling the devices instead, and thus avoids frequent interruptions sharing the same message that there are lots of packets to process. Another advantage of this polling-based approach is that when the kernel is overwhelmed, the packets that cannot be handled in time are simply dropped in the device queues (i.e., overwritten in the incoming buffer). Intel's DPDK is another software-based acceleration for high speed networking applications that also uses polling to avoid the overhead of interrupt processing. Recent work by Hwang et al. [6] extends the DPDK libraries to provide low latency and high throughput networking in virtualized environments.

B. *Placement of Virtual Appliances*

Ideally network operators should place VNFs where they will be used most effectively and least expensively. Although

[6] In terms of the legal considerations, a major MSO in the US won a court battle against content providers regarding the technology of sharing a stored program in the cloud among multiple users (http://en.wikipedia.org/wiki/Cartoon_Network,_LP_v._CSC_Holdings,_Inc.).

the virtualization of certain network functions is straightforward, there are a number of network functions that have strict delay requirements. For example, network functions offered by middle-boxes usually depend on the network topology and these boxes are placed on the direct path between two end points. When virtualizing these functions and moving their software implementations into data centers, data traffic may go through indirect paths, causing a potential delay of packets. Therefore, the placement of VMs that carry VNFs is crucial to the performance of offered services. For these services, it would be advantageous and efficient to run some network functions at the edge of the network [8].

Using mobile core network as an example, we could place a PGW, which currently sits in the cellular core network, right next to an eNodeB, and forward user traffic to the Internet as early as possible. However, the co-location of PGW and eNodeB will make the mobility management difficult, as neighboring eNodeBs will no longer share the same PGW as the anchor point. A possible solution would be to install virtualized PGWs that handle traffic for a small geographical area at the Mobile Telephone Switching Office (MTSO) or some other network points of presence in the metro area. Future work regarding low latency operation should be based on the investigation of the redirection architecture and the carrier's footprint of data centers.

The placement of virtual appliances, such as VPN gateways, can also enhance the security features of networking services. Today's VPN gateways are usually installed at locations very deep into the core network. By moving virtualized VPN gateways to the network edge and closer to end users, we can better isolate VPN traffic from other Internet traffic and reduce the complexity of core networks. Clearly this approach may lead to the support of more VPN gateways than the current practice. Thus, there is a need to optimize the number of instantiated virtual VPN gateways.

C. Instantiation and Migration of Virtual Appliances

Network infrastructure will become more fluid when deploying VNFs. To consolidate VNFs running in VMs based on traffic demand, network operators need to instantiate and migrate virtual appliances dynamically and efficiently. The native solution of running VNFs in Linux or other commodity OS VMs has a slow instantiation time (around several seconds) and a relatively large memory footprint. The carrier-grade deployment of VNFs requires a lightweight VM implementation. For instance, Martins et al. [9] recently proposed ClickOS, a tiny Xen-based VM to facilitate NFV. ClickOS can be instantiated within around 30 milliseconds and requires about 5 MB memory when running. However, optimizing the performance of this type of lightweight simplified VMs, especially during the wide-area migration, is still an open research issue.

Take virtual routers as an example, by enabling their free movement, carriers can separate the logical configurations (e.g., packet-forwarding functions) from physical routers, and simplify management tasks, such as planned maintenance [15]. However, it is challenging to keep the packet forwarding uninterrupted and the migration disruptions minimized; while at the same time guarantee the stringent throughput and latency requirements. Another interesting research topic is the design of a hypervisor [15] that splits the software of control plane from its state, such as routing information bases.

D. VNF Outsourcing

The end-to-end principle of initial Internet architecture that does not modify packets on-the-fly is no longer valid in current networks with the deployment of a variety of middle-boxes. Based on a study of 57 enterprise networks with different sizes, ranging from fewer than 1,000 hosts to more than 100,000 hosts, Sherry et al. [10] found that the number of middle-boxes in a typical enterprise is comparable to its number of hosted routers. In the last five years, surveyed large networks had paid more than a million US dollars for their middle-box equipment. Moreover, a network with about 100 middle-boxes may need a management team of 100-500 personnel for tasks such as configuration, upgrades, monitoring, diagnostics, training and vendor interaction [10].

By advocating the split of network functions and their locations, NFV makes the outsourcing of middle-boxes to a third-party [10] easier, which may release network carriers from some of the cumbersome operation and maintenance tasks. With the help of *VNF Service Providers* (e.g., cloud service providers or their partners), end users and small businesses may also be able to enjoy more diverse networking services which are previously not affordable due to their associated complexity and costs. However, the charging rules and policy interactions between carrier network infrastructure and outsourced VNFs need to be carefully investigated before taking actual actions. Another open question along this direction is to identify what types of VNFs can be outsourced to third parties and how to do it efficiently.

There are also several other open research issues for NFV. For example, using dedicated hardware appliances, it is relatively easy to identify which component is malfunctioning and isolate it when a failure occurs. When deploying network functions in software at different locations, *troubleshooting* and *fault isolation* become harder. Moreover, as the creation of VMs is easy, when the number of VNFs increases the so-called VM Sprawl could happen. There may be a large amount of VNFs sprawling across the network even if they are seldom used. As a result, the same management inefficiency problem that NFV was proposed to solve may recur. The efficient *management* and *orchestration* of VNFs, especially in the wide area, is another challenging issue.

VII. Conclusion

In this article, we presented an overview of the emerging network functions virtualization technology, illustrated its architectural framework, summarized several use cases and discussed some interesting future research directions. NFV extracts the functionality in specialized appliances and replicates it in the virtual form. It is envisioned that NFV, along with cloud computing and SDN, will become a critical enabling technology to radically revolutionize the way network operators architect and monetize their infrastructure. NFV is prospectively the unifying revolution among the three, offering more revenue opportunities in the services value chain. We are looking forward to more initiatives from the networking research community to tackle various challenging issues introduced by NFV and its widespread and successful adoption.

References

[1] V. Aggarwal, V. Gopalakrishnan, R. Jana, K. K. Ramakrishnan, and V. A. Vaishampayan. Optimizing Cloud Resources for Delivering IPTV Services Through Virtualization. *IEEE Transactions on Multimedia*, 15(4):789–801, June 2013.

[2] S. Bhaumik, S. P. Chandrabose, M. K. Jataprolu, G. Kumar, A. Muralidhar, P. Polakos, V. Srinivasan, and T. Woo. CloudIQ: A Framework for Processing Base Stations in a Data Center. In *Proceedings of MOBICOM 2012*, pages 125–136, Aug. 2012.

[3] China Mobile Research Institute. C-RAN The Road Towards Green RAN. China Mobile White Paper, Oct. 2011.

[4] M. Chiosi et al. Network Functions Virtualisation: An Introduction, Benefits, Enablers, Challenges & Call for Action. ETSI White Paper, Oct. 2012.

[5] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel. The Cost of a Cloud: Research Problems in Data Center Networks. *ACM SIGCOMM Computer Communication Review*, 39(1):68–73, Jan. 2009.

[6] J. Hwang, K. K. Ramakrishnan, and T. Wood. NetVM: High Performance and Flexible Networking Using Virtualization on Commodity Platforms. In *Proceedings of NSDI 2014*, pages 445–458, Apr. 2014.

[7] X. Jin, L. E. Li, L. Vanbever, and J. Rexford. SoftCell: Scalable and Flexible Cellular Core Network Architecture. In *Proceedings of CoNEXT 2013*, pages 163–174, Dec. 2013.

[8] A. Manzalini, R. Minerva, F. Callegati, W. Cerroni, and A. Campi. Clouds of Virtual Machines in Edge Networks. *IEEE Communications Magazine*, 51(7):63–70, July 2013.

[9] J. Martins, M. Ahmed, C. Raiciu, V. Olteanu, M. Honda, R. Bifulco, and F. Huici. ClickOS and the Art of Network Function Virtualization. In *Proceedings of NSDI 2014*, pages 459–473, Apr. 2014.

[10] J. Sherry, S. Hasan, C. Scott, A. Krishnamurthy, S. Ratnasamy, and V. Sekar. Making Middleboxes Someone Else’s Problem: Network Processing as a Cloud Service. In *Proceedings of SIGCOMM 2012*, pages 13–24, Aug. 2012.

[11] V. Sivaraman, T. Moors, H. H. Gharakheili, D. Ong, J. Matthews, and C. Russell. Virtualizing the Access Network via Open APIs. In *Proceedings of CoNEXT 2013*, pages 31–42, Dec. 2013.

[12] The European Telecommunications Standards Institute. Network Functions Virtualisation (NFV); Architectural Framework. GS NFV 002 (V1.1.1), Oct. 2013.

[13] The European Telecommunications Standards Institute. Network Functions Virtualisation (NFV); Use Cases. GS NFV 001 (V1.1.1), Oct. 2013.

[14] G. Wang and T. S. E. Ng. The Impact of Virtualization on Network Performance of Amazon EC2 Data Center. In *Proceedings of INFOCOM 2010*, pages 1163–1171, Mar. 2010.

[15] Y. Wang, E. Keller, B. Biskeborn, J. van der Merwe, and J. Rexford. Virtual Routers on the Move: Live Router Migration as a Network-Management Primitive. In *Proceedings of SIGCOMM 2008*, pages 231–242, Aug. 2008.