

A Review in the Core Technologies of 5G: Device-to-Device Communication, Multi-Access Edge Computing and Network Function Virtualization

Ruixuan Tu^{1*}, Ruxun Xiang^{2*}, Yang Xu³, Yihan Mei⁴

¹Sino-U.S. Program Attached Middle School to Jiangxi Normal University, Nanchang, China

²School of Communication Engineering, Chengdu University of Information Technology, Chengdu, China

³UoG-UESTC Joint School University of Electronic Science and Technology of China, Chengdu, China

⁴Lemania Altdorf Ecole, Shanghai, China

Email: cis2018@foxmail.com

How to cite this paper: Tu, R.X., Xiang, R.X., Xu, Y. and Mei, Y.H. (2019) A Review in the Core Technologies of 5G: Device-to-Device Communication, Multi-Access Edge Computing and Network Function Virtualization. *Int. J. Communications, Network and System Sciences*, 12, 125-150.
<https://doi.org/10.4236/ijcns.2019.129010>

Received: August 24, 2019

Accepted: September 27, 2019

Published: September 30, 2019

Copyright © 2019 by author(s) and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution-NonCommercial International License (CC BY-NC 4.0).
<http://creativecommons.org/licenses/by-nc/4.0/>



Open Access

Abstract

5G is a new generation of mobile networking that aims to achieve unparalleled speed and performance. To accomplish this, three technologies, Device-to-Device communication (D2D), multi-access edge computing (MEC) and network function virtualization (NFV) with ClickOS, have been a significant part of 5G, and this paper mainly discusses them. D2D enables direct communication between devices without the relay of base station. In 5G, a two-tier cellular network composed of traditional cellular network system and D2D is an efficient method for realizing high-speed communication. MEC unloads work from end devices and clouds platforms to widespread nodes, and connects the nodes together with outside devices and third-party providers, in order to diminish the overloading effect on any device caused by enormous applications and improve users' quality of experience (QoE). There is also a NFV method in order to fulfill the 5G requirements. In this part, an optimized virtual machine for middle-boxes named ClickOS is introduced, and it is evaluated in several aspects. Some middle boxes are being implemented in the ClickOS and proved to have outstanding performances.

Keywords

5th Generation, Network, Virtualization, Device-To-Device communication, Base Station, Direct Communication, Interference, Multi-Access Edge Computing, Mobile Edge Computing

*Co-first authors, alphabetically sorted by last name.

1. Introduction

Nowadays, mobile networking is becoming more and more popular as the main way for people to access the Internet. Thus, people start to use enormous mobile data. With the increasing number of users, various online services are emerging, for example, media streaming, social networking, and game. Also, new connection-based devices such as Internet of Things (IoT) and Machine-Type-Communications (MTC) are also developing. It brings a predictable giant growth of data of 8-fold from 2015 to 2020 and requires a much more exacting standard in communication and faster data transmission [1]. How to effectively increase network capacity, improve wireless spectrum utilization, and improve the users' experience in different communication modes has become an urgent task. As a result, the 4G communication will be no longer satisfy our increasing need in the efficiency. Thus, the concept of 5G was born.

The fifth-generation mobile communication standard, also known as the fifth-generation mobile communication technology, abbreviated as 5G, is an extension after 4G; it also technically named "IMT-2020", developed by the International Telecommunication Union (ITU). The digit 2020 means it is estimated to be widely published for commercial use in the year of 2020, and the 5G technology is now under experiments because it must fully satisfy standards of some key indicators: 1) user's experience rate: 0.1 - 1 Gb/s; 2) low latency: the end-to-end delay is reduced to milliseconds; 3) connection density: one million/km²; 4) high data rate: the theoretical downlink rate of 5G network is 10 Gb/s.

There is a large number of 5G required technologies, such as ultra-intensive heterogeneous network, micro base station, beam forming, device-to-device communication, software-defined network, multi-access edge computing, and network function virtualization.

In this paper, the study mainly focuses on the summary of three 5G related technologies: 1) Device-To-Device communication (D2D); 2) Mobile Edge computing (MEC); 3) Network Function Virtualization (NFV) and ClickOS. We then provide a tutorial overview of these fields respectively.

In the area of D2D communication, we first introduce the basic knowledge of D2D—a communication method that allows high-speed and direct communication between two devices without base stations in a range, and introduce four categories of D2D communication in detail along with challenges of D2D communication including interference management and security issues.

In the area of MEC, an introduction to MEC is presented in the aspects of computation offloading, distributed cloud delivery and caching, web performance enhancements and big-data. Next, we introduce the applications of MEC.

In the area of NFV, we briefly introduce the concept of a middlebox and NFV—a technology that moves middlebox from hardware to software through a kind of virtual machine. Also, an original virtual machine ClickOS is introduced, including the architecture and performance analysis of ClickOS, such as delay, boot time, temperature rate, etc. Finally, we introduce a real-world test of ClickOS running middleboxes.

2. Background

It is known that the architecture of 4G network consists of three parts: radio access network, bearer network and core network, and thus, we introduce the transformation of 5G through the aspects of radio access network and core network that relate to our research.

2.1. Radio Access Network

Normally, the base station of radio access network consists of three parts: an antenna, an RRU (Radio Remote Unit), and a Building Baseband Unit (BBU). During 3G and 4G, BBU and RRU are separated placed to decrease the carrier maintenance cost. While to further reduce the cost of the infrastructure of base station, the BBUs in different cabinets are placed centrally, to a station call Central Office (CO) and thus gather into the BBU baseband pool to realize a unified management and scheduling of resource. With the use of BBU baseband pool, the concept of NFV technology and Software Defined Network (SDN) can thus be introduced into the radio access network to support virtual machines (VM) and run software with BBU functions. The operators then only need to buy a general-purpose server, and the corresponding software can be installed on it to implement the function of the BBU without purchasing a BBU device anymore.

Also, with the use of MEC, the servers are now moving from their original data center to different rooms to achieve local content cache and the reduction of network latency. MEC is similar to NFV that it also emphasizes functional software and platform openness. Moreover, MEC can conduct optimization to NFV according to the radio access network environment, and deeply integrate the mobile access network with the Internet service.

In the year of 2016, ETSI extended the concept of MEC to Multi-Access Edge Computing, extending edge computing from telecommunications cellular network to other wireless access networks such as IEEE 802.11. At this point, the MEC can be seen as a cloud server running a specific task at the edge of the mobile network. And this is one of our stressing points in this paper.

The reason why MEC techniques introduced is also related to the key indicators of 5G, as we mentioned before that 5G requires a delay of milliseconds, and the propagation rate of fiber is 200 KM/ms. Data will be transmitted between terminals and core networks that are hundreds of kilometers apart. Obviously, the transmitting time is larger than milliseconds. As a result, we can only think of caching content to the access network side. It is also with this MEC that laid the foundation for the transformation of the core network.

2.2. Core Network

With the rise of high-definition video and applications on Virtual Reality (VR) and Augmented Reality (AR), large data traffic has brought huge challenges to the processing power of the core network. To carefully address these issues, in the 5G, the network architecture adopts a “full separation” approach, which di-

vides the S-GW and P-GW units in 4G into control plane and user plane. The function of user plane is processed in the access network (MEC servers), having reduced lots of overhead.

Another transformation in the core network is network slicing mechanism. 5G is internet-of-everything-oriented, except for data and voice services that 3G and 4G provide. Consequently, 5G divides the network into different virtual subnetworks according to different business needs, by doing so allows the network to better adapt and handle the business.

3. Device-to-Device Communication

Device-to-Device communication is an essential technology of 5G, which allows high-speed transmission of large amounts of data between two devices within a short range. In this section, three aspects of D2D communication will be discussed: overview, categorization and challenges of D2D communication.

3.1. Overview of D2D Communication

D2D communication technology is a communication method that enables direct communication between two devices within a certain distance. Compared with traditional communication, D2D can exchange information without base stations. **Figure 1** illustrates the difference between D2D communication and traditional one [2].

In traditional cellular network, the channel is divided into two links and users cannot directly communicate with each other. This centralized communication method is beneficial to resource management and interference control. But even if the user exchanges information face-to-face, the information is relayed through entities such as base stations, which greatly reduces the utilization of spectrum resources [2].

Unlike traditional communication architectures, D2D allows devices to communicate directly using cell resources without network infrastructure. The most significant difference between D2D and traditional cellular networks communication is that there is no need for relay of base stations. when devices are in a

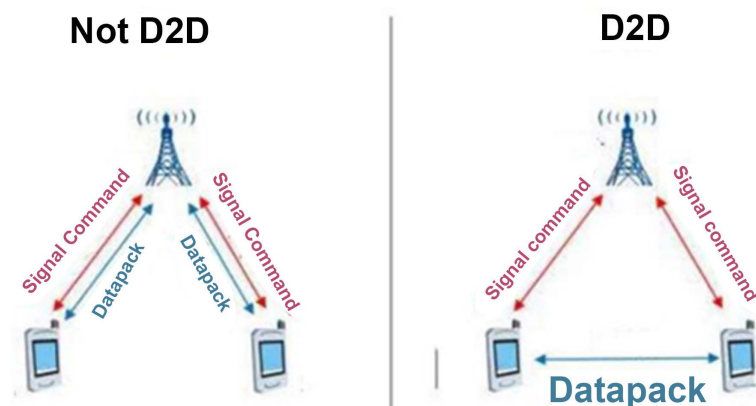


Figure 1. Illustration of traditional cellular network and D2D communication [2].

poor coverage area, devices are allowed to communicate with each other by creating a multi-hop, central-less, self-organized and wireless network which allows two terminals forward packets with other nodes. D2D communication can improve the communication rate, reduce the load of the base stations, reduce the communication delay, reduce battery consumption, and improve the quality of service (QoS) of the wireless network. Based on the characteristics of D2D communication, it can be used for emergency communication, Internet of Things enhancement and local services.

Compared with short-range communication technologies such as Bluetooth and WLAN, the advantage of D2D is that it works in licensed band, the interference environment is controllable, the data transmission is more reliable, a large amount of information interaction can be realized in a short time, and D2D communication does not require any operation from users, which providing a better user experience.

3.2. Categorization of D2D Communication

Different from the traditional cellular network system, D2D can perform resource allocation and link establishment under the control of the base station and can also exchange information when there is no network infrastructure. D2D communication can be divided into centralized control and distributed control.

Centralized control is similar to the traditional cellular network communication, with D2D link establishment being controlled by the base stations.

Distributed control is the type of D2D communication that establishes and maintains links autonomously without participation of base station.

Centralized control with the participation of base stations or other relay entities can manage resources and controls interference but increases the load on the base stations. The distributed control of device self-participation makes it easier to obtain all the link information between D2D devices but increases the complexity of The D2D device and increases the risk of interference.

Based on the degree of involvement of cellular operator (full/partial/no control over resource allocation), D2D communication is divided into four categories as the following figures shows [3].

3.2.1. Device Relaying with Operator-Controlled Link Establishment (DR-OC)

Figure 2 illustrates the communication process and link establishment of DR-OC communication. In this case, devices can communicate with the base station through the relay of other devices, and the operator partial or full involves in the link establishment [3].

This type allows the device to act as a relay device to achieve an expansion of the transmission range, which allows devices to achieve a higher quality of service. The involvement of operators and base stations enables the resource allocation and call setup, which reducing interference to some extent.

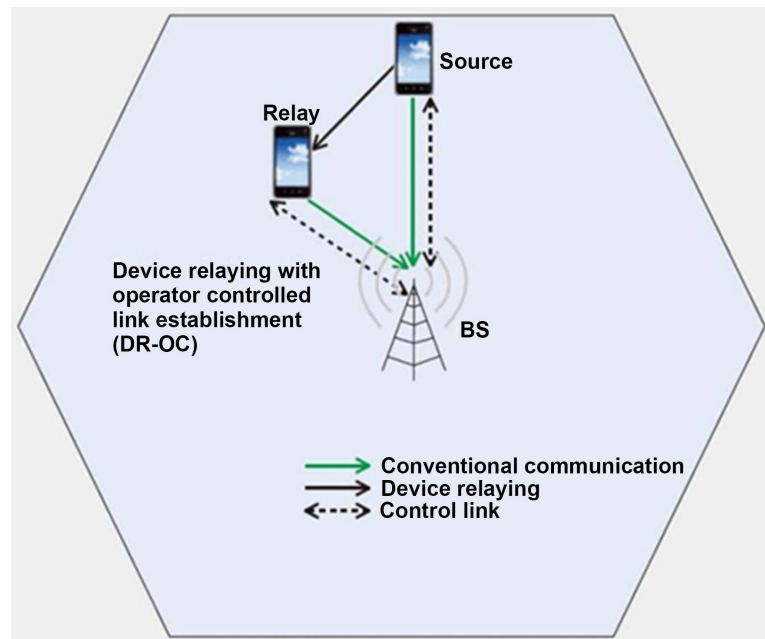


Figure 2. Illustration of DR-OC Communication [3].

3.2.2. Direct D2D Communication with Operator-Controlled Link Establishment (DC-OC)

As shown in **Figure 3**, the source and destination devices communicate with each other directly without base stations, and the link establishment is completed with the help of operator [3].

Compared with DR-OC, the difference between them is that DC-OC uses D2D communication. Devices under the same base station communicate with each other directly, without the relaying of the base station. This type of communication greatly reduces the workload of the base station, improves the efficiency of communication, and saves costs to a certain extent. In this case, the link establishment is performed by operators, which improves the efficiency of resource allocation and reduces interference.

3.2.3. Device Relaying with Device-Controlled Link Establishment (DR-DC)

Source and destination devices communicate using relays and establish the communication link by themselves as illustrated in **Figure 4** [3].

This type uses device-controlled link establishment method. Operators and base stations are not involved in the establishment of all links. Links established between the source device, the destination device and the relay devices.

This way gives the devices autonomy, greatly reduces the load of base stations and achieves high-speed and efficient transmission. At the same time, there are some problems. For DC-OC communications, there is no centralized entity to monitor the resource allocation between devices. Inevitably it will impact macrocell users operating in the same licensed band. Therefore, additional intelligent interference management policies are needed to address this issue.

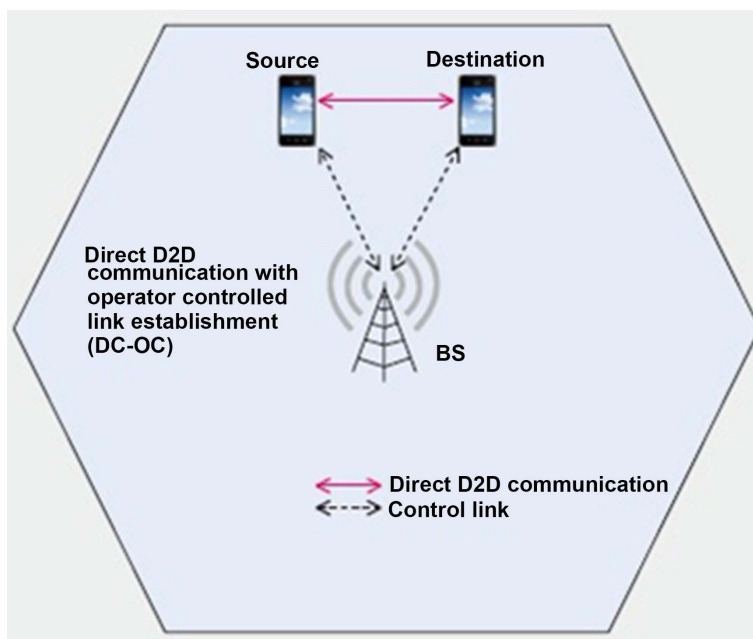


Figure 3. Illustration of DC-OC Communication [3].

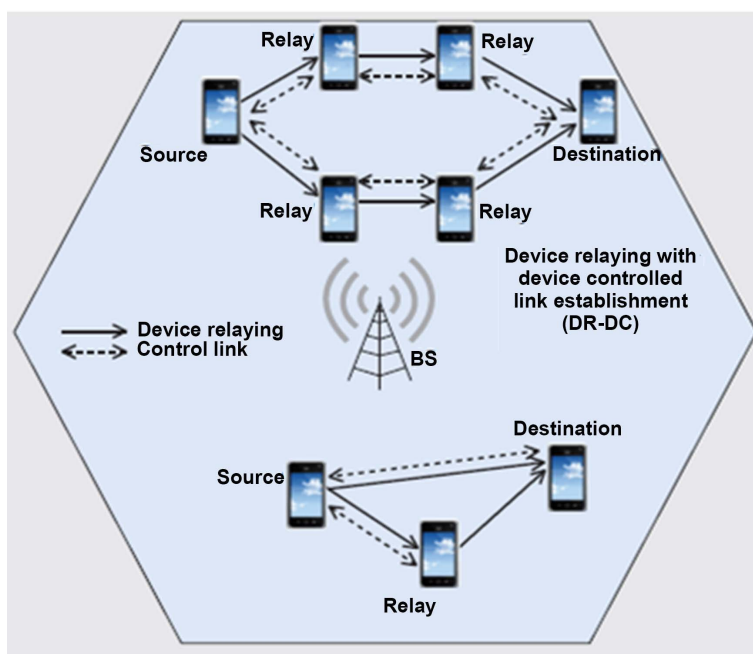


Figure 4. Illustration of DR-DC [3].

3.2.4. Direct D2D Communication with Device-Controlled Link Establishment (DC-DC)

As shown in **Figure 5**, source and destination devices communicate directly with each other and establish communication link without control of operator [3].

In this type, direct communication achieves without the help of relay devices and operators. This simplest way achieves maximum communication efficiency.

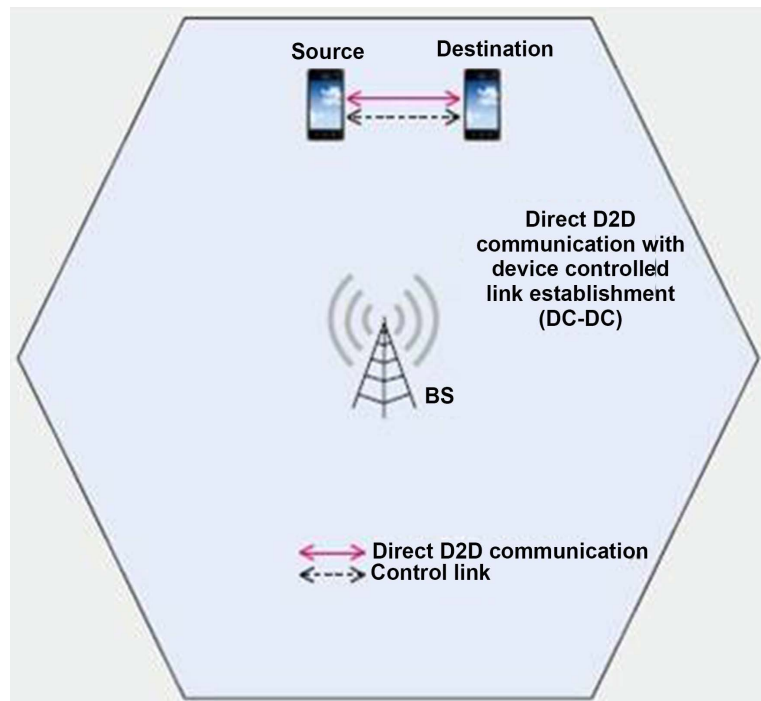


Figure 5. Illustration of DC-DC [3].

3.3. Challenges of D2D Communication in Two-Tier Cellular Network

Two-tier cellular network is an example of the application of D2D technology to mobile cellular networks. This combination can improve resource utilization and network capacity, but at the same time, it also creates some problems. While D2D shares wireless resources with the cellular network, it brings certain interference. When data is relayed through user devices, security issues need to be considered.

Thus, interference management and security problem are two challenges of D2D communication, and there are some corresponding solutions of them.

3.3.1. Interference Management and Solutions

There are two kinds of interference in two-tier cellular network.

First, there is interference between the two tiers. Without supervision of resource allocation between devices, devices in D2D tier inevitably interfere users' operating in the same licensed band in Macro-cell. This problem can be solved through the involvement of supervisors [3]. For example, in DR-OC and DC-OC, resource allocation and communication link set up are controlled by base station. Therefore, base station can alleviate the problem via centralized method.

Second, there is interference among users in the device tier. There are no entities for distributed D2D communication to perform macro-control and resource allocation, devices in the same frequency band must have interference. The employment of approaches such as resource pooling can alleviate resource allocation problem in this type of communication [3].

3.3.2. Security Problem and Solutions

For D2D communication, when two devices communicate with each other, some device terminals will function as transmission relays. In this case, security problem must be considered for the privacy of relays.

The use of *closed access* is a possible solution of this problem. There is a list of “trusted” devices in closed access. Only devices on the list can use device tier (D2D) to communicate with others. Without the list, any device can be used as relays without restrictions which may causes potential attacks and threats of system. Devices are encrypted between each other to avoid leakage of information during transmission and only trusted devices can exchange information freely [3].

4. MEC

In this section, we introduce multi-access edge computing (MEC) by differentiating it from other related technologies (*i.e.* Fog Computing) and talk about its applications (*i.e.* Computation Offloading, Distributed Cloud Delivery and Caching, Web Performance Enhancements, and Big-data), fundamentals (*i.e.* Cloud Computing, VMs, Containers, NFV, SDN, Network Slicing), and frameworks by several sections.

4.1. Mobile Edge Computing

With the widespread of smartphones, Mobile Network Operators (MNOs) are now facing efficiency problems on network resources, for the dramatic development of compact applications. To optimize perspectives like storage and calculation, Mobile Cloud Computing (MCC) has been put into use. In order to enhance more, edge-cloud, with technologies like Cloudlets, fog computing, Mobile Edge Computing (MEC) has been introduced.

4.1.1. Cloudlets

There is a three-layer architecture consisting of end devices, edge cloud platforms, and centralized data centers in network transportation as **Figure 6** shows. Cloudlets are parts of edge cloud platforms on the middle of the architecture. Cloudlets are VM-based, so it is just a micro data center, offering access to end users for managing their own VMs. Extensively, VM migration is possible.

4.1.2. Fog Computing

Fog computing and cloud computing are both computing implementations, so they both provide storage, applications, and data with end-users. However, there are still some differences between cloud computing and fog computing, and we can find some special features which are only available in fog computing: firstly, as **Figure 7** demonstrates, fog computing devices are closer to users, which bring low latency. Secondly, for the total number of users is not changed, fog computing devices are more widespread geographically. Thirdly, for the dense distribution of fog computing, fog computing is a medium weight and intermediate

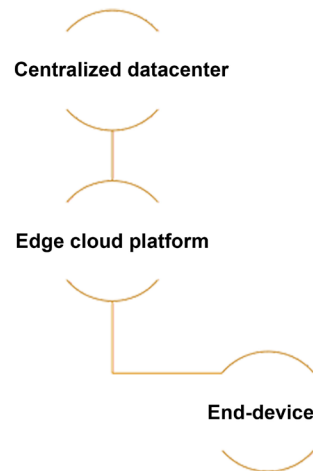


Figure 6. The position for cloudlets in a three-layer architecture.

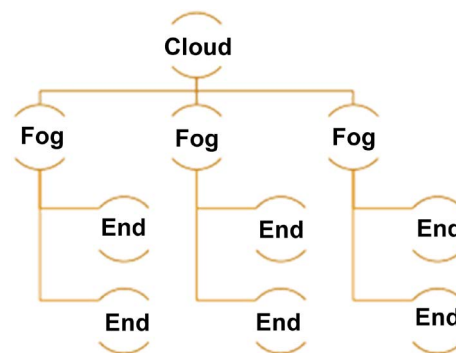


Figure 7. Fog computing, cloud computing, and end devices.

level of computing power compared with the heavyweight, powerful form of cloud computing.

4.1.3. MECs

There are 2 MECs: Mobile Edge Computing and Multi-Access Edge Computing. The two MECs both provide network features like proximity, low latency, reliability, and scalability with end-users. The abbreviations of them are the same, so the meanings of them are quite similar. The slight difference between them is that Multi-Access Edge Computing includes the access points that are also a part of network edges. However, Mobile Edge Computing, as its name says, does not include these access points. **Figure 8** shows that multi-edge computing contains more potential devices than mobile edge computing.

4.1.4. Fog Computing and MEC

Both fog and edge computing can run on mobile and wireless networks. The differences between them are: firstly, fog computing also uses wired connections to transmit data. Secondly, edge computing is much simpler than most of fog computing instances, for edge computing has only one layer, while fog has many

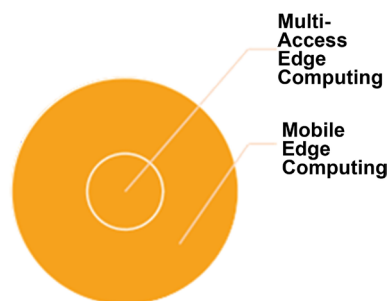


Figure 8. 2 MECs.

intermediate ones between edge computing devices and the cloud. Also, the differentiation is shown in **Figure 9**.

4.2. Applications

In this section, we talk about several applications of MEC: Computation Offloading, Distributed Cloud Delivery and Caching, Web Performance Enhancements, and Big-data. All of them have their own strategies to reduce the computing power that end-devices should have.

4.2.1. Computation Offloading

Computation offloading is a technology that enables heavyweight calculation-required tasks to fully or partially unload to cloud instances. To illustrate, in some computation-intense applications, such as video encoding, IoT information concentrating, and mobile gaming, can be moved into more widespread small instances, to provide energy-saving and low-latency services. In **Figure 10**, the intensity of calculation is lower than it is on backend servers, and the position of calculation is between end-users and the Internet, which speeds up the time required for tasks that are not suitable for both cloud platforms and end devices.

4.2.2. Distributed Cloud Delivery and Caching

This technology is mainly local caching which is used in additional applications besides video streaming and AR. The 2 instances have been chosen because of their widely-used and bandwidth-occupying features. **Figure 11** shows that this technology plays a role in a streaming network by more widely distributing and caching contents of data centers. As for videos, CDN usage can improve Quality of Experience (QoE) by up to 35% [4]. Media Cloud, which is distribution of flexible video streaming services, and interconnected cloud edges, which is a strategy to share cached content between edges, are also ways for effectively caching. On the other hand, for AR, QoS from MEC platforms and local caching for large objects can also enhance QoE.

4.2.3. Web Performance Enhancements

Caching can not only be used by streaming service providers, but also enhance web experience. There are 3 strategies: content optimization: MEC platforms can

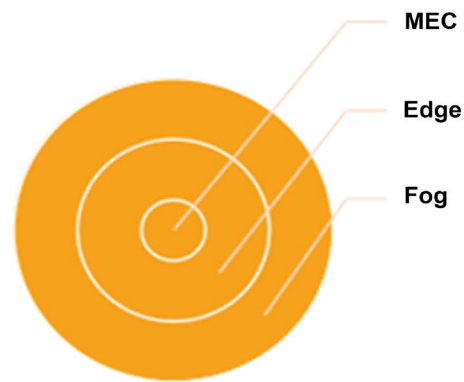


Figure 9. MEC, edge computing, and fog computing.

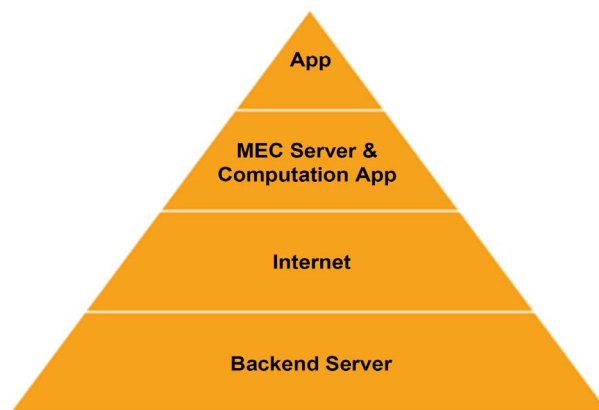


Figure 10. Illustration of computation offloading.

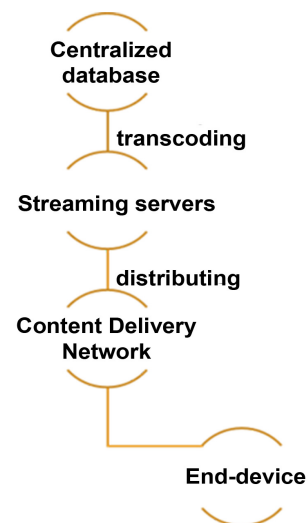


Figure 11. A streaming network with distribution and caching.

use users' cookie, history, location, or other information, to improve QoE, virtually without cloud instances. Accelerated browsing: edges can download contents with filters before processing. Web acceleration: an edge can perform like

CDN, to prestore large files on web pages.

4.2.4. Big-Data

There are 2 instances to use big-data on MEC platforms: IoT and smart city services. On the one hand, IoT instances can communicate with MEC services, in order to do data aggregation and analytics mostly locally. Thus, MEC gateway can filter requests between IoT devices and cloud servers. On the other hand, smart city services can get data from frontend pages and perform big data aggregation and analytics. For example, intelligent public spaces, public safety, censorship, energy saving are all promising applications for MEC-based big-data processing.

4.3. Fundamental Technologies

4.3.1. Cloud Computing, VMs & Containers

For cloud platforms, to classify them, there are 4 technology models: private, public, hybrid, and community. There is also another way of classification: there are 3 service models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). **Figure 12** shows the difference among 2 of the 4 technology models. These two technologies, VMs and containers, are ways to build cloud platforms. They are similar but there are still some significant differences. As for similarity, they both provide virtualization, abstraction, isolation, and security. However, as **Figure 13** [5] shows, VMs provide these features on a more complicated, more complete level. The detailed differences are listed below:

- VMs work at hardware abstraction level, while containers work at system call application binary interface layer.
- VMs provide abstraction for full guest OSs, while containers provide abstraction directly for the guest processes.
- VMs have slower provisioning.
- VMs consume more resource.
- VMs provide full isolation, which is more secure, while containers provide process level isolation, which is less secure.

4.3.2. Network Function Virtualization

The Network Function Virtualization (NFV) is an architectural concept which visualizes functions of network nodes into connectable components.

It includes 3 domains: virtualized Network Functions (VNFs): it allows the allocation of multiple instances on every one of the same virtualized environments. NFV Infrastructure (NFVI): it includes the resources, the physical and virtual components of network environments with VNFs. NFV Management and Orchestration (NFV MANO): it manages NFVI.

4.3.3. Software Defined Networks

Software Defined Networks (SDN) is a widely-used technology that enables network programmability and rapid deployment. It also supports flexible service

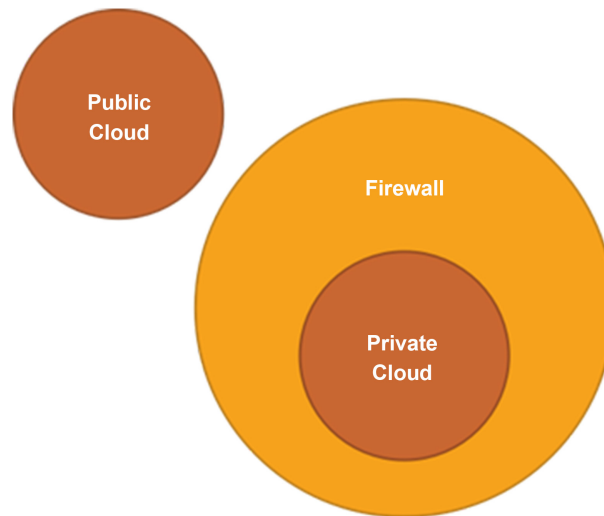


Figure 12. Classification by technology models.

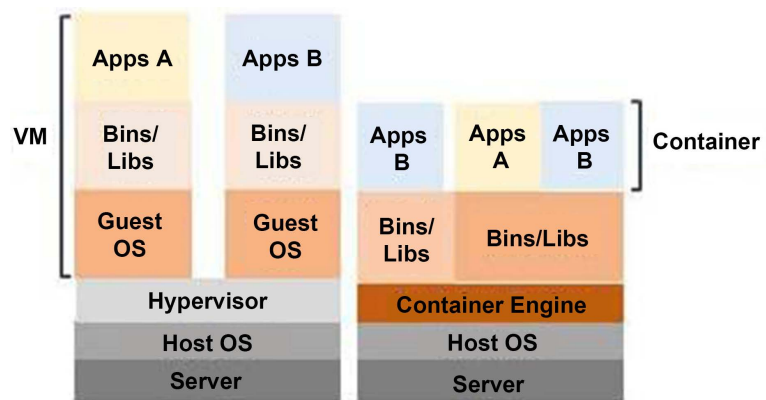


Figure 13. Differentiation of VMs and Containers [5].

chaining and cross-layer options, which can avoid IP translation and tunneling problems in 5G networks.

4.3.4. Network Slicing

Network slicing means slicing one network to multiple instances for respectively specific optimization. It supports multiple isolated, customized networks on a common physical infrastructure. **Figure 14** [5] shows an example of network slicing for different situations.

4.4. MEC Framework and Architecture

4.4.1. MEC Framework

The framework is a structure of MEC, which includes mobile edge system level, mobile edge host level, and networks level from top to down as **Figure 15** [6] shows. The networks level is the nearest physical part which provides fundamental communications. The mobile edge host level is the most important part, which is consisted of the mobile edge host which supports the virtualization infrastructure like NFVI and the mobile edge platform, and the mobile edge host

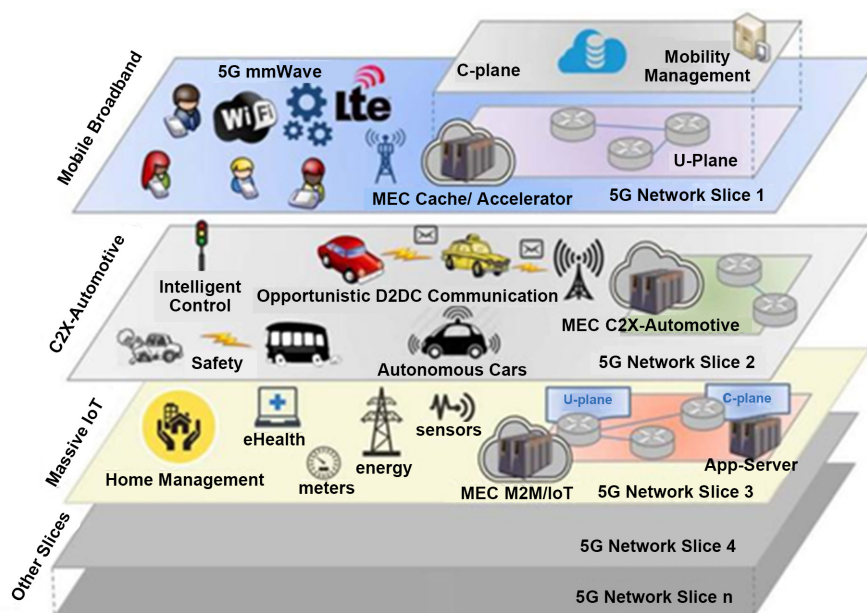


Figure 14. Network slicing in mobile, automotive, and IoT services [5].

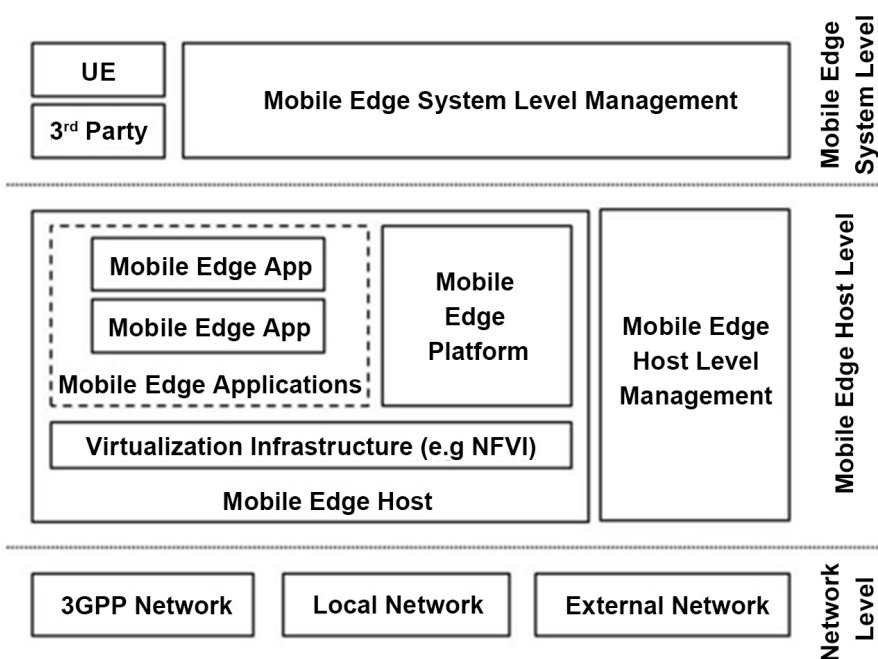


Figure 15. ETSI MEC framework [6].

level management. The top level, the mobile edge system level, exposes abstracted MEC services as APIs for UEs and third parties.

4.4.2. MEC Reference Architecture

The MEC reference architecture in Figure 16 [6] illustrates the mobile edge system level and the mobile edge host level in details. It shows how functions and interfaces in the MEC system connect with each other. Additionally, mobile edge applications (APPs) are running over the virtualized infrastructure.

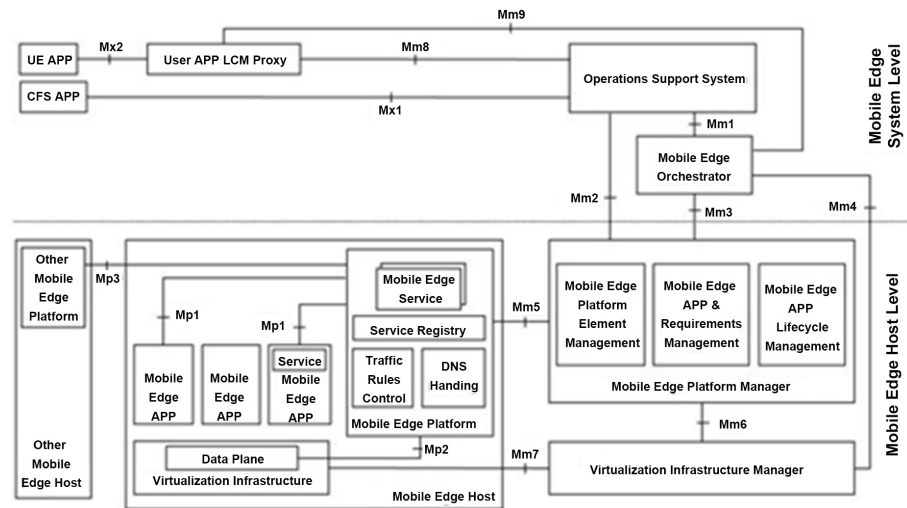


Figure 16. MEC reference architecture [6].

The mobile edge host level contains the mobile edge host, the mobile edge platform manager, and the virtualization infrastructure manager. The mobile edge platform manager provides management of APPs' life cycles, mobile edge platform elements, and APPs' policies such as DNS configuration and traffic rules. Besides, the Virtualized Infrastructure Manager (VIM) manages virtualized resources, fast installation images, and error reporting information.

4.4.3. Proofs of Concept (PoC)

With the MEC framework and architecture, the standardization group of ETSI MEC Industry Enabling Group (IEG) is adopting MEC technologies by PoCs. These 13 PoCs [7] are now developed:

- Video user experience optimization via MEC;
- Edge video orchestration and video clip replay via MEC;
- Radio-aware video optimization in a fully virtualized network;
- Flexible IP-based services (FLIPS);
- Enterprise Services;
- Healthcare dynamic hospital user, IoT and alert status management;
- Multi-service MEC platform for advanced service delivery;
- Video analytics;
- MEC platform to enable low-latency Industrial IoT;
- Service Aware MEC Platform to enable Bandwidth Management of RAN;
- Communication Traffic Management for V2X;
- MEC enabled OTT business;
- MEC infotainment for smart roads and city hot spots.

4.5. MEC Services and Network Orchestration

4.5.1. MEC Service Orchestration

In order to promote the operation efficiency of MEC, the following service-related points can be taken into consideration:

Resource allocation: Resources determine performance. VMs manage CPU, memory, storage, network bandwidth for tasks like computation offloading. An allocating method by semi-Markov decision is described in [8]. Service placement: The locations of MEC nodes are also critical for QoE. That is for the further the nodes are from users, the worse the delay and the capacity will be. The MEC nodes should be placed in gravity points where data demands concentrate, in order to diminish the cost and maximize the efficiency.

Edge selection: The most common strategy, connecting to the nearest edge, might be not the best because of movements and wireless conditions of users, and loads of edges. Methods which are better adopted to mobile environments should be created.

Reliability: To improve reliability, a technology named checkpoint can be used. It manages snapshots of applications in order to restore them when errors encounter. In mobile environments, the conditions may need frequent point checking, which decrease the scalability. A solution is to duplicate MEC instances.

4.5.2. MEC Service Mobility

Frequent edge changing is a usual problem in mobile networks. Redirecting user requests to a distant edge hosting the service may not be an optimum solution. Also, frequent MEC migrations are not a good idea. Besides, IP changes would lead to session breakdowns. In order to resolve these problems, technologies like DNS and NAT are introduced with a concept named follow-me, which forces data anchor replacing with service anchor with seamless IP transforming. There is also a solution based on SDN which uses a Location/ID Separation Protocol (LISP) to diminish triangular routing and provide faster migrations.

4.6. Discussion

MEC is an innovative technology that brings edge computing online with public APIs accessible by third parties, fast speed, and high flexibility. It enhances QoE not only physically, but virtually by exposing the network layer available for developers. MEC has been a key technology in 5G mainly because of its low-latency assurance and capacity optimization. However, although there are still many problems which are overcoming by researchers, MEC will definitely advance networks around us in the near future.

5. Network Function Virtualization

In this section, we introduce another technique related to 5G communication. As we mentioned above, 5G requires a much higher throughput rate. Specifically, we need the networking devices to be much more effective in processing packets in a form called middlebox. In the following subsections we will introduce the concept of middleboxes, a network function virtualization method and a vitalize platform ClickOS. Also, results of detailed tests to evaluate this platform and a real-world implementation of middleboxes on ClickOS are introduced and included [9].

5.1. Middlebox Introduction

A middlebox is a computer networking device that transforms, inspects, filters, or manipulates traffic for purposes other than packet forwarding. It can be a firewall, an IDS to support security, or a load balancer to support traffic controlling. Nowadays, middleboxes are indispensable and basic parts in consisting of the present operational networks. Despite their great use, they have some defects: many of them are based on hardware, which is expensive in prices and in management, hard to manage for maintenance staffs, and due to the fixed form, their functionality is also hard to make alterations like expanding or reducing the functions.

5.2. Network Function Virtualization

To solve the issues mentioned above, an alternative method Network Function Virtualization (NFV) is proposed, by introducing it the operators can shift the hardware-based middleboxes processing to software running on inexpensive, commodity hardware. In order to fully replace the hardware-based middle boxes, NFV must satisfy a few requirements in functionality: 1) multi-tenancy; 2) accommodate a wide range of OSes, APIs, and software packages.

There are existing virtual platforms that might be feasible as candidates: hypervisor-based technologies such as Xen or KVM. There are some benefits such as they offer security and their performance is isolated and will not be disturbed. However, they do not support a large number of tenants; also, their network performance is not optimized for middlebox processing. Thus, a virtual platform ClickOS is proposed. ClickOS is a Xen-based software platform that is especially optimized for middlebox processing. Also, modifications in Xen's I/O subsystem are implemented in order to achieve the speedup of networking in middleboxes. The experimental results show that for simple packet generation, the throughput on Linux platform increases from 6.46 GB/s to 9.68 GB/s for 1500B packets and from 0.42 GB/s to 5.73 GB/s for minimum-sized packets.

5.3. ClickOS Virtual Machine

To realize flexibility, isolation, multi-tenancy and scalability, based on Xen, ClickOS virtual machine is built to provide a low-delay, high-throughput networking services. For a general view of ClickOS, it consists of three major parts: 1) Click modular route software, it makes it convenient to reuse the middlebox functionalities, and with more than 300 stocked elements, it also makes it easy to construct middleboxes; 2) MiniOS, it is a tiny operating system (built in Xen) that allows us to build efficient and virtualized middleboxes without unnecessary expenses; 3) Xen I/O sub-system optimization, which will allow a faster networking for traditional VMs. The ClickOS architecture is presented in **Figure 17** [9].

Based on Xen, ClickOS is split into a Dom0 and DomUs where Dom0 is a privileged virtual machine acts as the driver domain and DomUs are the domains of guests or users comprising their virtual machine. For the drivers, originally,

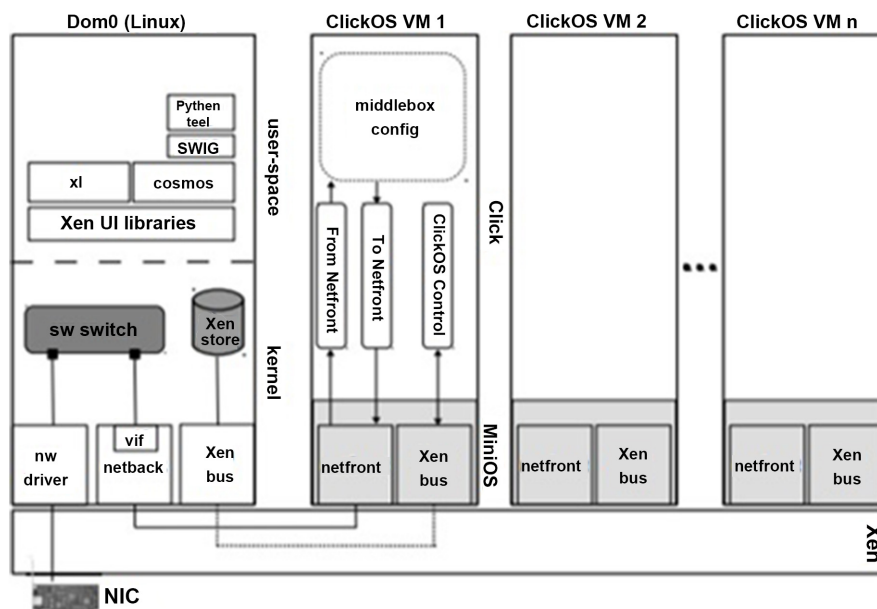


Figure 17. ClickOS architecture [9].

Xen has a split driver, where the back half runs in the driver domain and the front half in the guest VM. In the following part, we will present the modifications toward this driver model especially for ClickOS. MiniOS running on it realizes all the basic functionalities needed in a virtual machine. Furthermore, each ClickOS VM consists of the Click modular router software running on top of MiniOS, and with the use of Xen store database, guest domains are able to share control information.

Analysis of Xen network is implemented to test its bottlenecks. During the test, an Intel Xeon E3-1220 3.1 GHz 4-core CPU is used as a server, and a 16 GB memory along with an Intel x520-T2 dual Ethernet port 10 Gb/s card are used. Also, the server had Xen 4.2. The result is shown in **Figure 18** [9].

To evaluate the performance in throughput rate with the use of the original driver model, experiments are conducted and the result shows in **Figure 18** that the netfront driver yields poor throughput rate, especially for Rx, which barely handle 8 Kp/s, some modifications are put to the netfront driver to reuse the memory grants, and the Rx rate has increased to 344 Kp/s, though, still far from the 10 Gb/s line rate figure of 822 Kp/s. Next, the experimenter modified the software switch, replacing Open vSwitch with VALE (communicate using the netmap API), and the transmit rate has increased to 1.2 Mp/s for 64-byte packets. Thus, according to the experiments, ClickOS Network Input and Output is redesigned as **Figure 19** [9] shows.

There are three major steps in the re-design of I/O: 1) replace the standard but sub-optimal Open vSwitch back-end switch with the high-speed VALE so that the NIC connects directly to the switch, and they increase the maximum number of ports on the switch from 64 to 256; 2) remove the netback driver from the pipe; 3) change the netfront driver to map the ring buffers into its memory

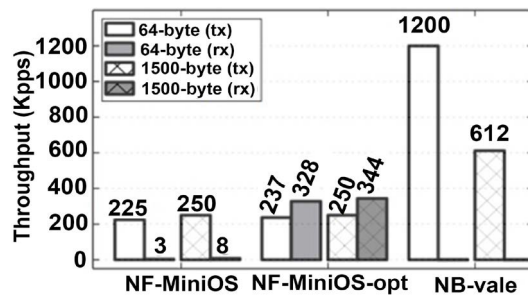


Figure 18. Throughput test [9].

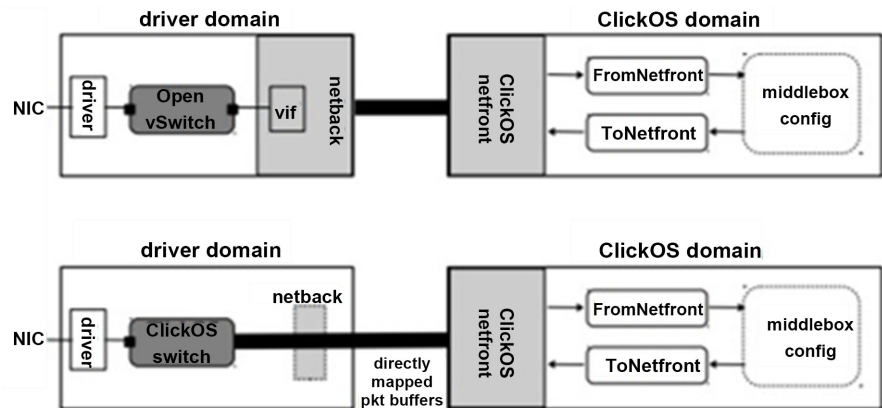


Figure 19. Standard Xen I/O pipe and ClickOS one [9].

spaces. Also, there are some other changes like allowing asynchronous transmit to speed to transmit throughout, granted re-use, Linux support and Click modifications.

5.4. ClickOS Accomplishment

In this part we introduce the accomplishment of ClickOS in 7 aspects: 1) ClickOS switch Performance; 2) boot time; 3) delay; 4) throughput; 5) state insertion; 6) chaining; 7) scaling out.

The experimental settings are as follows: 1) A low-end Intel Xeon E3-1220 server with four 3.1 GHz core and 16 GB RAM which is used in most test; 2) A mid-range Intel Xeon E5-1650 server with three 3.2 GHz core and 16 GB RAM. As for the operating system, Linux 3.6.10 is used for dom0 and domU and Xen 4.2.0 in all situations, Xen 4.2.0 is used to generate packets and measure the rates. For the software, pkt-gen application is used to generate one 10 Gb/s Ethernet port.

5.4.1. ClickOS Switch Performance

Experimenters have generated two 10 Gb/s Ethernet ports to test the scalability of the switch. As Figure 20 [9] shows, for the single port, the switch saturated the pipe for all packets sizes, from 64 bytes to 1024 bytes, and for the two ports, the switch as well saturated the pipes for all packets sizes while achieving 70% of line rates except for the 64 bytes packets.

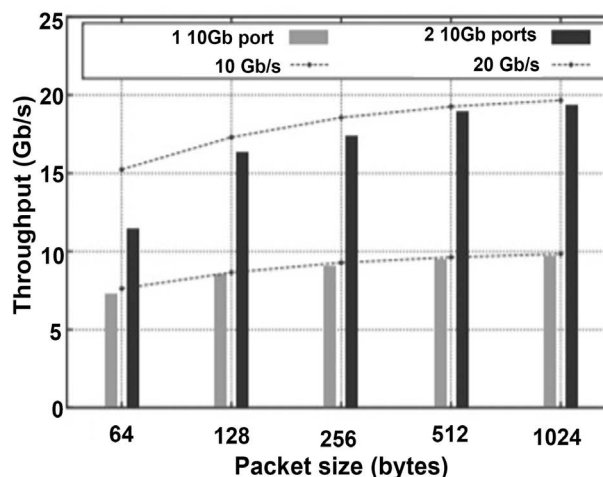


Figure 20. Performance using one and two 10 Gb/s ports [9].

5.4.2. ClickOS VM Boot Time

Figure 21 [9] shows that, for one ClickOS VM, it realized a time of 20.8 msecs, with 6.6 msecs to install a Click configuration, the total amount of time equates to approximately 28.8 msecs to the middlebox is up and running. Next, they measured how the booting number of ClickOS VMs can affect the boot time, and they booted large numbers of VMs (to 400) on the same system, and results show that it can up to a maximum of 2019 msecs. The increase in the boot time could be attributed to the contention on the Xen store and thus could be improved.

5.4.3. Delay

Virtualization technologies are sometimes notorious for the introducing of extra layers, which may bring additional delay, so we can see how ClickOS' network I/O pipe perform. When testing with disengaged ClickOS VMs, Figure 22 [9] presents a low delay of 45 μ secs and the number increased to 64 μ secs with 12 running VMs. Separately, Dom0 has a small delay of 41 μ secs. Consequently, compared with the unoptimized drivers of Xen and para-virtualized KVM drivers, ClickOS performs competitively in the measurement of delay.

5.4.4. Throughput

In this part the ClickOS is compared with the MiniOS, as we can see in the figure(c) of Figure 23 [9], ClickOS' transit performance is comparable to that of figure(a), which means that the ClickOS actually do not add much overhead. And the same is true for the receive performance except for the 64 bytes packets that drops from 12.0 Mp/s to 9.0 Mp/s.

5.4.5. State Insertion

To test feasibility of ClickOS, it must allow quick reads and writes in order to enable middleboxes to be quickly configured. Using the method of python and cosmos, the result in Figure 24 [9] shows that for cosmos, the read time is about 9.4 msecs and write time is about 0.1 msecs. And for experimental comparison

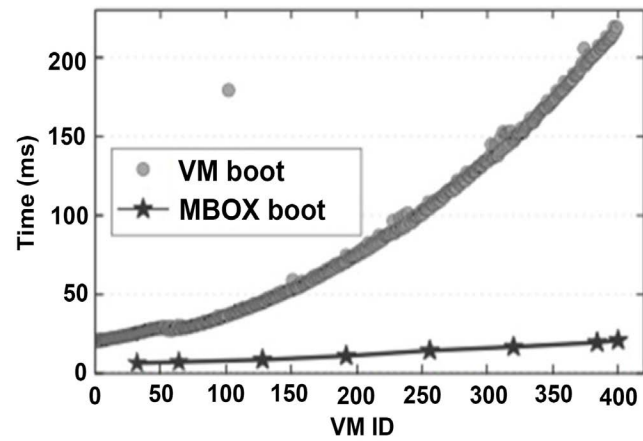


Figure 21. ClickOS VM boot time [9].

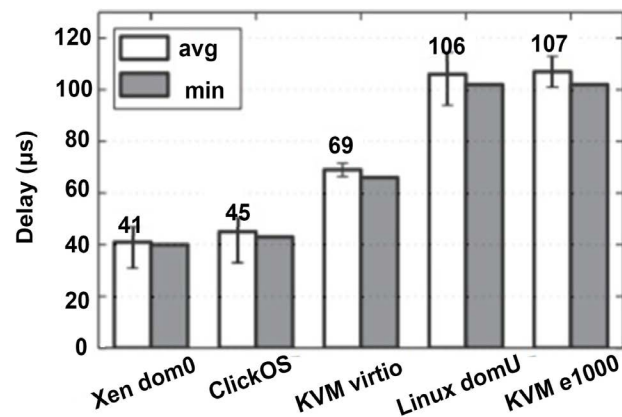


Figure 22. Idle VM ping delays for ClickOS, a Linux Xen VM, Dom0, and KVM using the e1000 or virtio drivers [9].

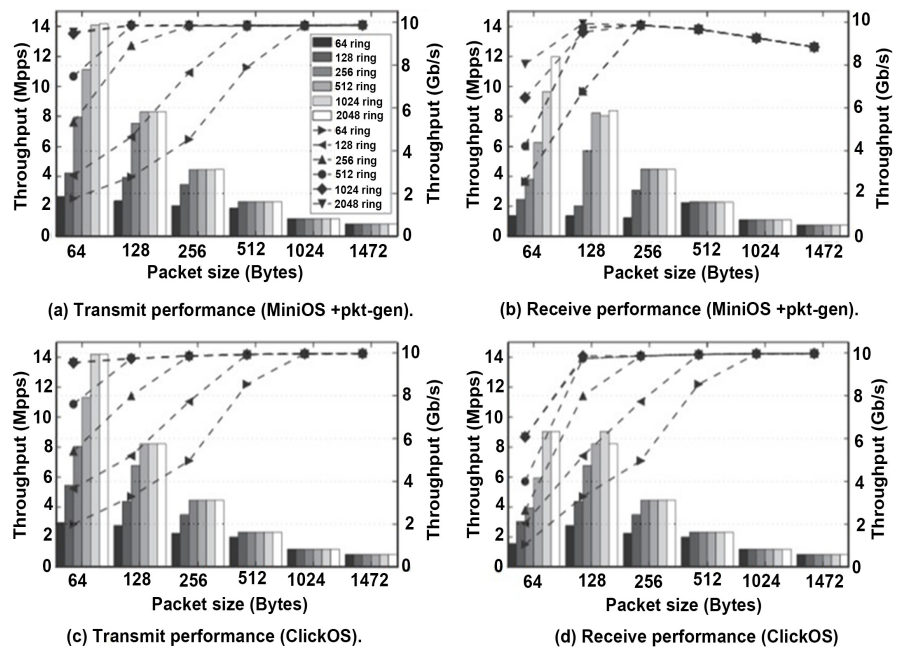


Figure 23. Baseline throughput measurement [9].

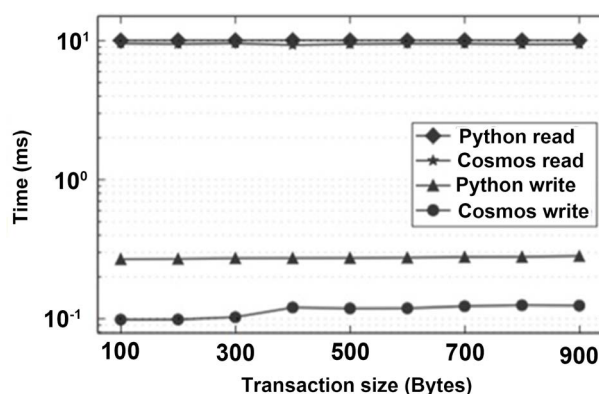


Figure 24. ClickOS middlebox write and read [9].

and completeness, the read time is 10.1 msec and the write time is 0.3 msec for python method.

5.4.6. Chaining

It is normal for middle boxes to be chained back to back, for example, the fire-wall can be followed by a flow monitor. It is necessary to test the throughput while the middleboxes are in a chain. **Figure 25** [9] shows that a longer chaining leads to a lower rate: from 21.7 Gb/s with a chain of length 2 to 3.1 Gb/s with a chain of 9. The decrease in throughput rate can be attributed to the overload of single CPU because the experiment is implemented on one CPU. In addition, some extra copy operations and the load of Dom0 are to blame.

5.4.7. Scaling Out

To test the scaling out ability, many VMs are launched on one CPU core and one 10 Gb/s port. The result in **Figure 26** [9] shows that however, the number of the VMs are, a cumulative throughput rate equals to line rate for 512 bytes packets, 1024 bytes packets and 1472 bytes packets, and a rate of 4.85 Mp/s for 64 bytes packets. Then, when testing the scalability for additional CPU cores and 10 Gb/s ports. The result in **Figure 27** [9] shows that up to 4 ports, the line rate for maximum-sized increased steadily, while invalid for 5 and 6 ports.

5.5. Middlebox Implementation on ClickOS

Finally, to evaluate ClickOS performance under the context of real situation, different actual middleboxes are implemented on the ClickOS, and they are: Wire (WR); Ether Mirror (EM); IP Router (IR); Firewall (FW); Carrier Grade NAT (CN); Software BRAS (BR); Intrusion Detection System (IDS); Load Balancer (LB); Flow Monitor (FM).

For this implementation test, two of the low-end servers are used, one of which is used to generate packets to the other server. A total four CPU cores are involved in, and the ClickOS virtual machine is allotted with a single CPU core, and the other three are distributed to dom0.

Under these settings, the throughput rate for each middlebox is shown in **Figure 28** [9]. In general, ClickOS performs well enough for requirements of

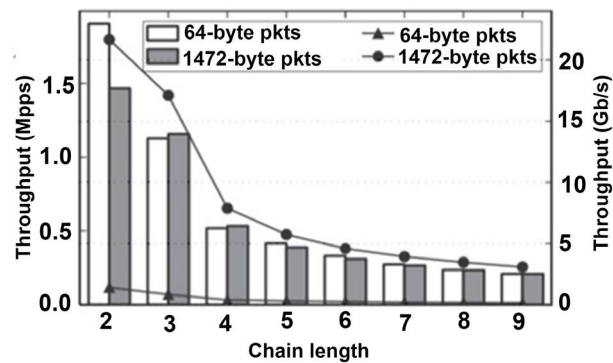


Figure 25. Performance when chaining ClickOS VMs back-to-back [9].

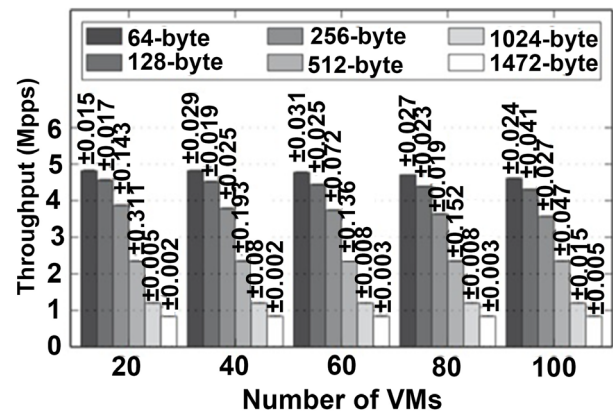


Figure 26. Running many VMs on one core and one 10 Gb/s port [9].

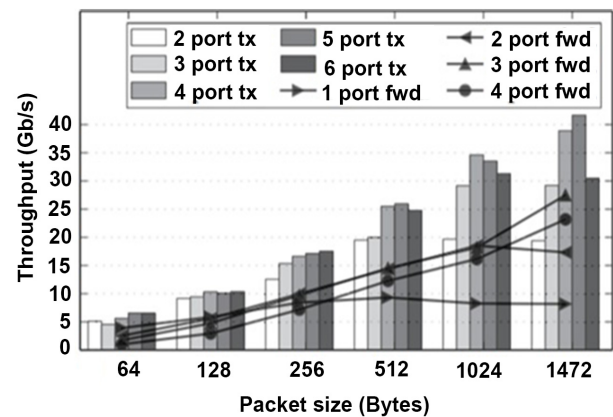


Figure 27. Cumulative throughput using multiple 10 Gb/s ports and one VM per port [9].

large packets, which saturated almost line rate for all middlebox configurations. Although for requirements of smaller packets, the rates drop from line rate, ClickOS is capable of processing the packets in millions/sec.

To sum up, as one of 5G techniques, the network function virtualization addresses the problems brought by hard-ware based middleboxes effectively. And

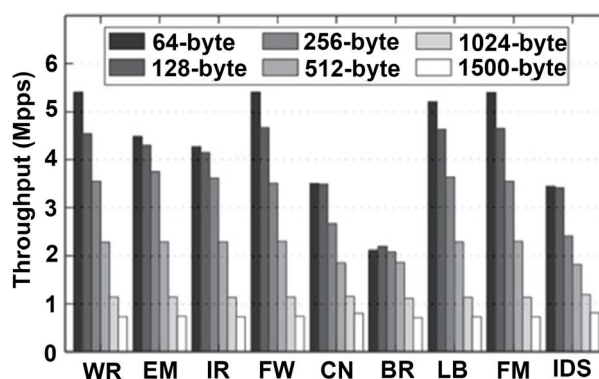


Figure 28. Performance for different middleboxes [9].

the virtual machine ClickOS provides an ideal platform for middlebox implementations with specialized optimized configurations for middlebox, low delay, excellent state insertion, chaining and scaling out performance and validated high throughput rates for packet transmitting and receiving. Absolutely, the more research is being conducted, the better 5G network will be.

6. Conclusions

5G changes mobile networking by MEC, NFV, edges and middleboxes. Not only can fog computing platforms, computation offloading, widespread computing, network slicing in MEC and middleboxes and polished ClickOs in NFV enhance QoE, but other promising technologies such as D2D communication will also advance networks around us. D2D technology greatly improves the speed and efficiency of communication. The use of D2D-based two-tier cellular networks also meets different communication situations.

From the increase of transmission rate, the reduction of transmission delay and the improvement of spectrum utilization efficiency, to the Internet of Everything, multi-network convergence of heterogeneous networks, the changes brought by 5G will be significant and thorough. It is absolutely that the more we research, the better 5G networks will be.

Our future work includes: 1) solving the D2D peak communication latency issue when large amounts of devices use D2D communication simultaneously; 2) upgrading traditional cellular network by realizing the compatibility of D2D and cellular networks; 3) exploring other technologies, for example, millimeter waves, massive MIMO, NOMA (non-orthogonal multiple access); 4) finding a quick way to widely deploy MEC nodes. 5) conducting research on the connection of software defined network—emphasized on the network structure and NFV—emphasized on redefining the structure of the network element device and their application on the 5G; 6) introducing a 5G network cloudization frame based on SDN and NFV that meets the future of 5G networks with flexibility, intelligence, integration and openness; 7) analyzing the technical advantages and creative applications of SDN and NFV, specifically on the dynamic configuration of the wireless network resource and standardization of hardware equipment.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] White Paper (2016) Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015-2020.
- [2] Xiao, Z.J. (2017) The First Time Someone Made 5G So Simple.
https://m.sohu.com/a/319547645_465176/?pvid=000115_3w_a
- [3] Tehrani, M.N., Uysal, M. and Yanikomeroglu, H. (2014) Device-to-Device Communication in 5G Cellular Networks: Challenges, Solutions, and Future Directions”. *IEEE Communications Magazine*, **52**, 86-92.
<https://doi.org/10.1109/MCOM.2014.6815897>
- [4] Intel Corporate (2013) White Paper: Small cell Revolutionize Service Delivery. Santa Clara, CA, USA.
- [5] Taleb, T., *et al.* (2017) On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration. *IEEE Communications Surveys & Tutorials*, **19**, 1657-1681.
<https://doi.org/10.1109/COMST.2017.2705720>
- [6] Mobile Edge Computing (MEC) (2016) Framework and Reference Architecture, v1.1.1. ETSI GS MEC Standard 003.
- [7] ETSI (2016) MEC Proofs of Concept. Sophia Antipolis, France.
<https://www.etsi.org/technologies/multi-access-edge-computing/mec-poc>
- [8] Liu, Y., Lee, M.J. and Zheng, Y. (2016) Adaptive Multi-Resource Allocation for Cloudlet-Based Mobile Cloud Computing System. *IEEE Transactions on Mobile Computing*, **15**, 2398-2410. <https://doi.org/10.1109/TMC.2015.2504091>
- [9] Martins, J., *et al.* (2014) ClickOS and the Art of Network Function Virtualization. *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14)*, Seattle, WA, 2-4 April 2014, 459-473.
<https://www.usenix.org/conference/nsdi14/technical-sessions/presentation/martins>