Virtualization Enabled Multi-Point Cooperation with Convergence of Communication, Caching, and Computing

Shu Fu, Nan Cheng, Ning Zhang, Xin Jian, Feng Lyu, Hong Wen, and Xuemin (Sherman) Shen

Abstract

NFV can effectively improve the flexibility of resource allocation and save energy consumption in networks. With NFV, the servers in communication networks can be virtualized as different types of VMs, including SIVMs and CIVMs. To deal with the ever-increasing traffic, in this article, we study the cooperation of servers at different locations. This can efficiently and flexibly utilize the communication, caching, and computing resources to minimize network energy consumption with QoS guarantee. Specifically, we first investigate the impacts of different types of VMs on network performance. Then, we propose a joint allocation of different types of VMs in different servers, and discuss the tradeoffs between SIVMs and CIVMs. Case studies are also provided to verify the feasibility of VM based multi-point cooperation. Finally, potential research directions are presented to shed light on the future study of VM cooperation.

Introduction

According to a Cisco report, annual global traffic will reach 3.3 Zeta Bytes by 2021, which is a 127-fold increase from 2005 [1]. To accommodate the massive growth in data traffic and diverse services in a more energy efficient manner, future networks are envisioned to efficiently utilize network resources including communication, caching, and computing (3C) resources [2].

Network function virtualization (NFV) [3] can enable flexible and agile resource allocation, where the network servers with different types of resources are virtualized into virtual machines (VMs). To be specific, the resources of communication, computing, and caching in servers can be virtualized into storage-intense virtual machines (SIVM) and computation-intense virtual machines (CIVM) [4–5]. According to the European Telecommunication Standards Institute (ETSI), NFV standards have defined NFV structure and entities such as NFV orchestrator (NFVO) and NFV infrastructure (NFVI). NFVI describes the software and hardware components of the system on which virtual functions are built, while NFVO is mainly to increase interoperability of the system, and perform orchestration of resource and network services.

Allocation of different amounts of resources at various servers in the network can lead to low resource utilization and poor service provisioning. This calls for traditional wireless multi-point cooperation [6, 7] to be extended to VMs based multi-Point cooperation with the convergence of communication, caching, and computing. Specifically, SIVMs can cooperatively store the copies of the contents in remote service providers in a cache-as-a-service (CaaS) manner [8], which can significantly reduce the wired resource costs involved in data routing. CIVMs can cooperatively form a computing resource pool to enable fast network optimization [8]. Different allocation of VMs can lead to trade-offs among different network performance metrics, such as delay, throughput, and energy consumption, and the trade-offs in performance among wired and wireless segments of the network, which pose great challenges to the design of resource allocation.

In the literature, cooperative multi-point (CoMP) transmission technology in wireless networks is surveyed in [9]. Wired caching of services as CaaS to save energy consumption in traffic routing is investigated in [8]. However, the cooperation between caches is not considered in this work. Extensive research works investigate VM allocation at BSs to enhance the radio access network's performance [4-5]. However, these works only focus on cooperation based on a single type of VM, and lack the consideration of the heterogeneous resources [10].

In this article, we investigate multi-point VM cooperation with 3C convergence. We focus on the allocation of different types of VMs, as well as the cooperation of VMs to improve resource utilization. We first introduce a software defined networking (SDN) architecture with virtualization technology to enable flexible cooperation in allocating VMs at multiple servers. Then, we propose three types of VM cooperations:

- BS cooperation based on SIVMs
- Joint cooperative caching and data routing based on SIVMs
- Cooperative computation based on CIVMs.

Finally, we discuss the trade-off of VM allocations, including the trade-off of VM allocation between CIVMs and SIVMs, and between wireless access and wired networks. We also propose

Shu Fu and Xin Jian are with Chongqing University; Nan Cheng is with Xidian University; Ning Zhang (Corresponding author) is with Texas A&M University at Corpus Christi; Feng Lyu is with Shanghai Jiao Tong University; Hong Wen is with the University of Electronic Science and Technology of China; Xuemin (Sherman) Shen is with the University of Waterloo.

Digital Object Identifier: 10.1109/MNET.2019.1800140

0890-8044/19/$25.00 © 2019 IEEE

a VM based joint optimization framework to improve network performance. This work helps establish a cooperative framework for VM based virtualized resource allocation in terms of caching, computing, and communication. By integrating and flexibly allocating resources, the system performance can be significantly improved.

SDN Based Multi-Point Network Architecture

SDN architecture [11–13] can provide network agility and simplify network management, which will play a significant role in networks with dynamic conditions and heterogeneous resources of communication, caching, and computing. As shown in Fig. 1, through SDN, the control function is decoupled from the devices in the infrastructure plane which facilitates more flexible cooperation in the network. The decoupled SDN/NFV (network function virtualization) control function is placed at the control plane to provision NFV management and orchestration (NFVO). The network programmability capacity enabled by the control plane in SDN can help achieve virtual machine based cooperative networks (VCN). In the virtualization plane, servers will be virtualized as different categories of VMs. The virtual functions of computation and caching constitute of NFV infrastructure (NFVI) to provision more flexible resource utilization and cooperation in the network.

Servers possess two main hardware resources: computation resources and storage resources. Considering the software defined environment, it is desired to incorporate more functionalities into the network resources [4]. To improve resource utilization and facilitate the cooperation between resources, virtualization provides a more stable, flexible and energy efficient method. Specifically, servers deployed at different locations in the network are further virtualized as multiple SIVMs and CIVMs. The categories of VMs and their functions can be seen in Fig. 2. For SIVMs, they can be used as end-end SIVMs (EE-SIVMs) for data storage of wireless transmission, and data routing. SIVMs at different servers can also be utilized as content provider SIVMs (CP-SIVMs) to cache the data of the remote traffic providers. The larger amount of SIVMs can support more flexible resource allocation as well as lower the cost of users to offload network traffic. By CIVM, computing functions in network are virtualized to execute different computing tasks, where the capacity of each computing function can be flexibly allocated. For example, CIVMs can be used as cloud computing CIVM (CC-CIVM) for network optimization by forming a computing resource pool. CIVMs can also be used as edge computing CIVM (EC-CIVM) to achieve computing offloading [14]. The larger computing capacity can provide less network delay, and the cooperation of CIVMs can save network cost because computing resource is relatively expensive in network.

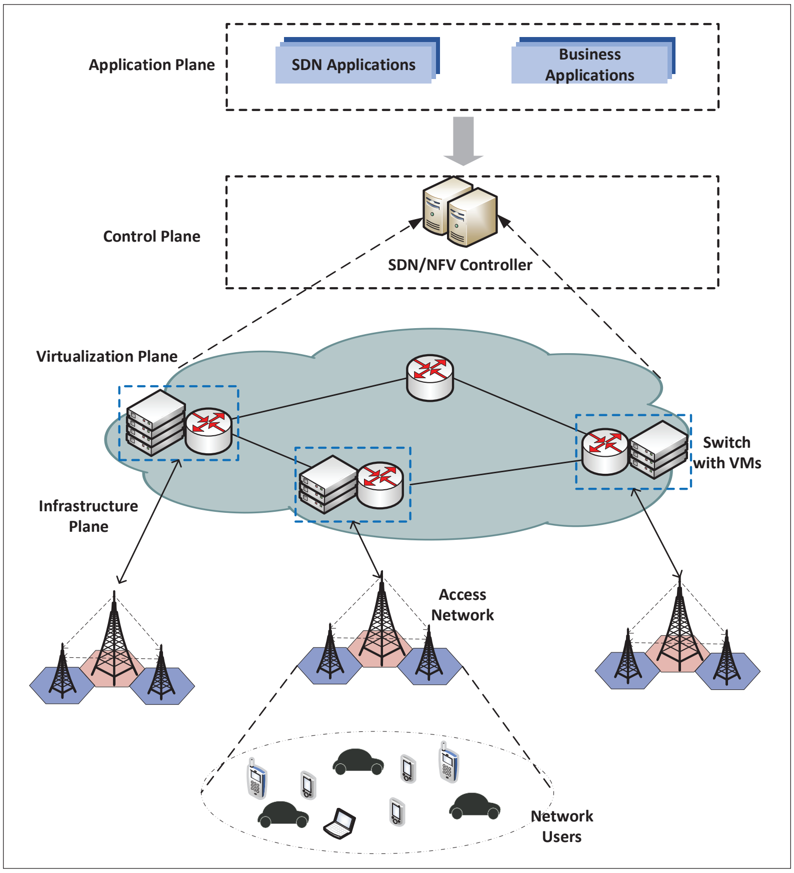


FIGURE 1. Software-defined network architecture with virtulization.

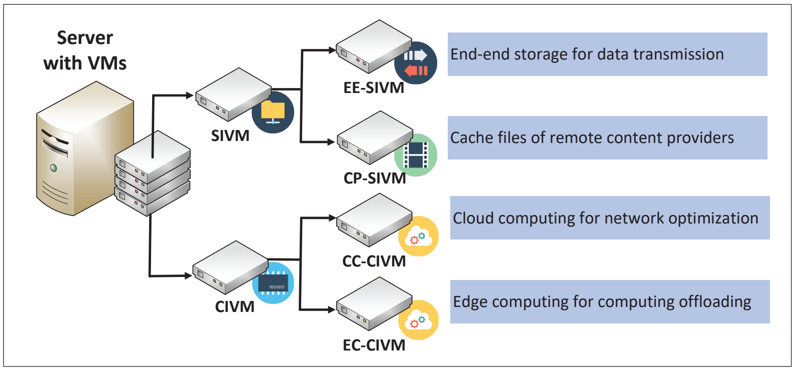

Server with VMs
SIVM
CIVM
EE-SIVM
CP-SIVM
CC-CIVM
EC-CIVM
End-end storage for data transmission
Cache files of remote content providers
Cloud computing for network optimization
Edge computing for computing offloading

FIGURE 2. Categories of VMs: SIVM and CIVM.

SIVMs (EE-SIVMs and CP-SIVMs) in the network infrastructure such as switches, BSs, and so on, constitute the user plane and CIVMs constitute the control plane. Controlled by the SDN controller, the requested user data can be fetched through two means. A part of the data can be routed between placements of servers and cached in EE-SIVMs for wireless transmissions, while the other part can be downloaded from remote service providers and cached in CP-SIVMs through CaaS to reduce the wired transmission costs. The data cached in CP-SIVMs can be routed to the intended destinations with less wired bandwidth and reduced power consumption. In terms of CIVMs, CC-CIVMs at multiple servers connect to a SDN controller and provide the capability of network computing. By EC-CIVM, a part of the computing tasks at the user end can be offloaded to the BS to cut down network delay and system energy consumption. Controllers can be further interlinked to facilitate large-scale distributed computing. Such an architecture is scalable since more SIVMs/CIVMs can be allocated to enhance the network performance without the need to deploy more servers physically.

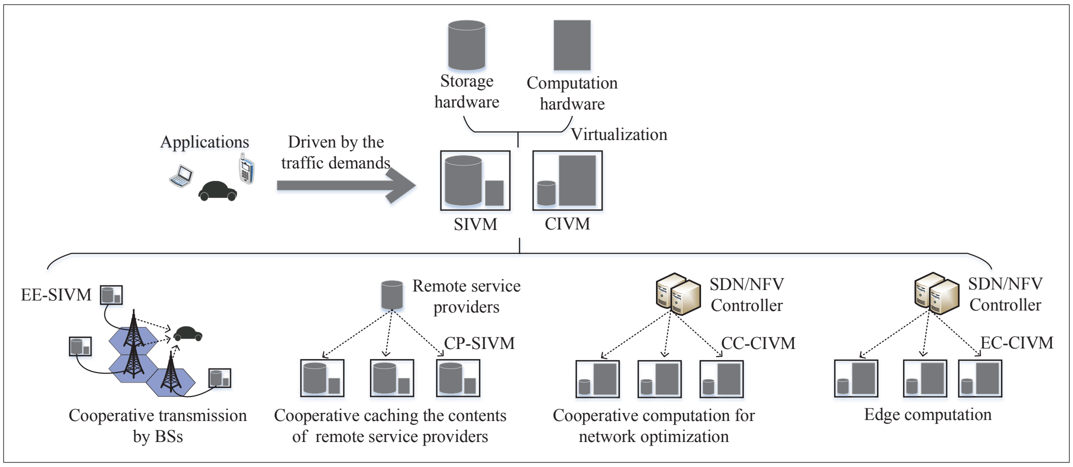


FIGURE 3. Trade-off and cooperations in VCN.

Multi-Point Cooperation with 3C

In this section, we review multi-point cooperation in the context of the network with 3C convergence, including cooperative transmission, cooperative caching, and cooperative computing.

BS Cooperation Mechanism

In VCN, the increased cell density can largely enhance the system throughput by improving the spatial frequency reuse factor. However, it also leads to higher energy consumption. One effective method to address this issue is BS cooperative transmission, such as joint transmission (JT) mode and muting mode in cooperative multipoint (CoMP). For the JT mode, a set of neighboring BSs and users in the coverage areas of the BSs will be clustered as one cooperative BS cluster (CBS). Since BSs in the same CBS may connect to different switches or servers, EE-SIVMs will cooperate to cache the user data. The data can be transmitted to the users from multiple BSs simultaneously to achieve an enhanced data rate. For the muting mode, BSs in the same CBS will use the frequency sub-channel in a pre-determined manner to avoid inter-cell interference in the same CBS. In wireless communication, CoMP is generally employed to improve the performance of edge users [7].

By employing an appropriate pre-coding scheme in CBS, the available space resource can be enlarged as the size of CBS increases. This provides more flexibility and performance gain, and thus the energy efficiency of wireless transmissions can be improved.

Cooperative Caching Mechanism

When the data requested by users is cached in the nearby CP-SIVMs, the data can be directly fetched from the CP-SIVMs in a CaaS manner. This can largely cut down the overall data routing distance, wired delay, and the costs. For example, in wavelength division multiplexing (WDM) based data routing, multiple wavelengths are contained in each optical link, each of which is carried on one optical path. The ends of one optical path are configured as a pair of lasers for the transmitting and receiving of the optical signal. Frequent data routings on the optical links will lead to large energy consumption and costs [15] when the remote traffic is demanded frequently. By CP-SIVM based CaaS, a large distance of routing can be avoided, and thus energy consumption and routing costs are reduced.

Moreover, to release the caching pressure of CP-SIVMs, the amount of data copy stored in the nearby CP-SIVMs should be determined by the traffic demand and the amount of the available CP-SIVMs. The remaining traffic data will still be stored in the caches of the remote service providers.

Cooperative Computing Mechanism

In terms of the cooperative computing mechanism, CC-CIVMs at multiple servers can jointly form one computing resource pool to optimize network operations. The overall optimization task is divided into multiple sub-tasks which are allocated to different CC-CIVMs and executed in a distributed manner. When the sub-tasks are accomplished, all the results are fed back to the SDN controller, in which the results are integrated and analyzed. Therefore, optimization can be achieved by distributive and parallel computing in CC-CIVMs. The cooperation of CIVMs can largely improve the computing resource utility and save network cost because computing resource is generally the most expensive in 3C resources.

Trade-Off on VM Allocation

Due to the limited amount of network resources and the maximal power of servers at each placement point P_0, there might be trade-offs among different allocations of the resources. In this section, we first analyze the impacts of VM allocation on the performance of multi-point cooperation, and discuss the trade-offs resulting from the VM allocation.

In Fig. 3, we describe the architecture of cooperations in VCN. According to the demands of caching and computation in the network, the allocation between SIVMs and CIVMs will be appropriately determined to achieve the tradeoff between the wireless capacity of communication, caching, and computation. The wireless capacity is involved with the capacity of EE-SIVM (bit); the capacity of caching the contents of remote service providers is involved with the capacity of CP-SIVM (bit); and the capacity of computation is involved with the capacity of CC-CIVM (cycles per second (cps)). To improve resource utilization, resources of the same category can cooperate across multiple placement points. The available amount of resource for each category of VM is determined by the traffic demands and optimized to minimize the system cost under the constraints of maximal tolerant delay, amount of available hardware, the maximal power of servers at each placement point, and so on. Resources in VCN can be integrated by such cooperations between the same category of VM and trade-off between the allocation of different categories of VMs.

The Impact of VM Allocation on Cooperation Performance

We consider a VCN with N deployed placements of servers. In terms of multi-point cooperation on communication, the capacity of EE-SIVMs (bit) regarding wireless transmitting is denoted by $\boldsymbol{W} = [W_1, W_2, ..., W_N]$, and the maximum wireless tolerable delay is denoted by $\boldsymbol{d}_W = [d^1_{\mathrm{W}}, d^2_{\mathrm{W}}, ..., d^N_{\mathrm{W}}]$. Then, the energy consumption function can be represented by $\varepsilon_W(\boldsymbol{W}, \boldsymbol{d}_W)$, as described in [7].

In terms of caching, we denote the capacity of CP-SIVMs (bit) by $\boldsymbol{Q} = [Q_1, Q_2, ... , Q_N]$, and the maximal wired tolerant delay by $\boldsymbol{d}_S = [d^1_S, d^2_S, ..., d^N_S]$. The capacity of EE-SIVMs (bit) regarding wired data routing is denoted by $\boldsymbol{S} = [S_1, S_2, ... , S_N]$. Then, the overall wired energy consumption can be represented as $\varepsilon_S = \varepsilon_S(\boldsymbol{S}, \boldsymbol{Q}, \boldsymbol{d}_S)$. Intuitively, larger $\boldsymbol{Q}$ leads to smaller ε_S and decreased $\boldsymbol{d}_S$ because larger amount of traffic data can be stored in the neighboring CP-SIVMs to reduce the energy consumption in data routing. However, as $\boldsymbol{Q}$ further increases, the decreased amount of ε_S and $\boldsymbol{d}_S$ will be lessened. This is because the overall demand of traffic data from the remote service providers is limited.

In terms of $\boldsymbol{S}$, we have proved that ε_S can be decreased by increasing $\boldsymbol{S}$ in the synchronous routing scenario [15]. However, determining an appropriate $\boldsymbol{S}$ is more complex in the asynchronous routing scenario, where the capacity of EE-SIVMs differs from each other and the related research problems remain open.

In terms of the cooperation in computing tasks, controllers first estimate the quantity of calculation, which can be measured by the number of CPU cycles. CC-CIVMs at different servers can constitute a computing resource pool and cooperate to optimize network operations. We denote the capacity of CC-CIVMs (cps) by $\boldsymbol{C} = [C_1, C_2, ... , C_N]$. Then, CC-CIVMs at servers of the N placement points constitute a computing pool with capacity $C = \Sigma_{i=1}^{N} C_i$. We consider that the number of CPU cycles involved in network optimization is $\aleph$. Then, the computing delay of the network optimization by CC-CIVM is $d_o = \aleph / C$. Therefore, more CC-CIVMs can be clustered to support a faster network optimization. Besides, for the edge computing via EC-CIVM, we denote the capacity of EC-CIVMs (cps) by $\boldsymbol{C}' = [C'_1, C'_2, ..., C'_N]$. For the placement point i, users possess the number of $\aleph_u$ CPU cycles for computing, and $\beta\aleph_u$ CPU cycles of computing task will be sent to the BS for edge computing via EC-CIVM. Then, the computing delay, d_{e}, by EC-CIVM at the placement point i is $(\beta\aleph_u)/C'_i$.

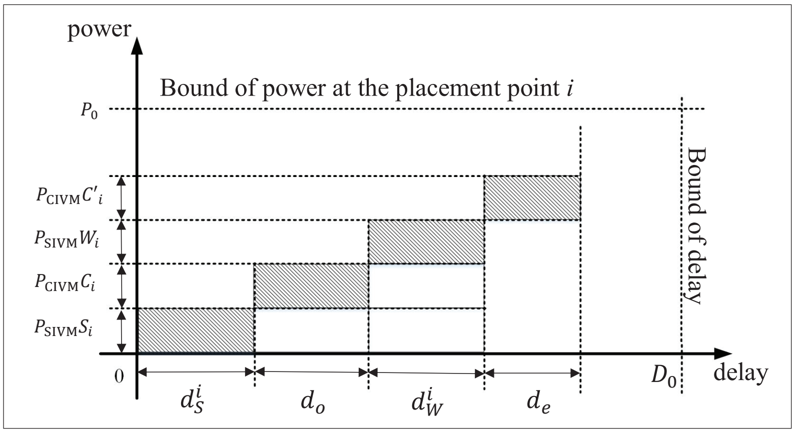


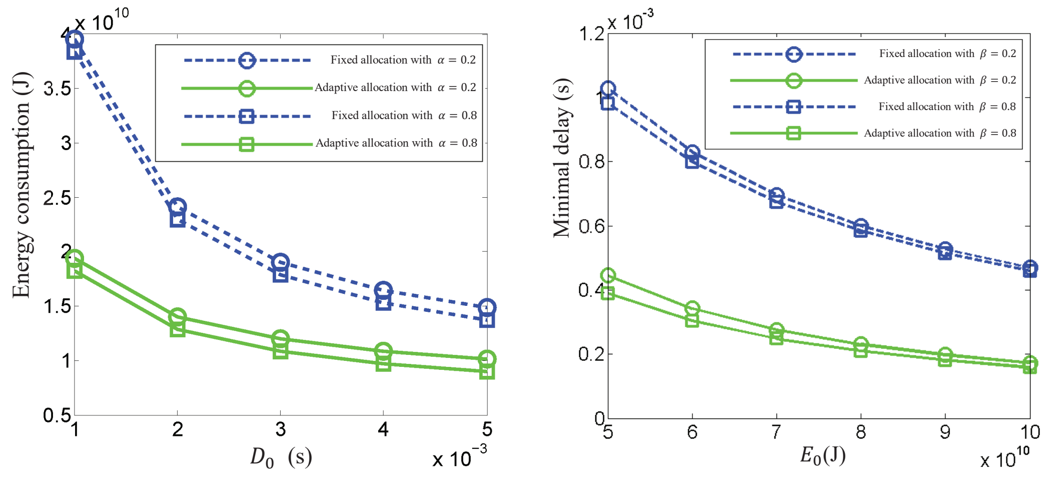
FIGURE 4. The trade-offs and bounds in VCN.

The Trade-Offs between VM Cooperations

In Fig. 4, we show an example of the trade-offs in VCN with N placements of servers. The system is divided into three parts, that is, data routing, network optimization, and wireless transmission. Each part corresponds to one category of the VMs in Fig. 3. The horizontal axis in Fig. 4 represents the system delay and the vertical axis represents the power consumption of VMs. The maximum tolerant system delay is denoted by D_0, the maximal power of the servers of each placement point is denoted by P_0, the power consumed by SIVM is denoted by P_s, and the power consumed by CIVM is denoted by P_c. The overall delay of the four parts should not exceed D_0, that is, $d^i_S + d_o + d_W + d_{\mathrm{e}} \leq D_0$. Moreover, the maximal power consumed at the placement point i is constrained by P_0. Taking the server placement optimization at the point i as an example, it corresponds to the second stripe piece in Fig. 4 which contains two orthogonal domains, that is, delay, d_o, and power consumption of CC-CIVMs at servers of placement point i. As discussed earlier, when d_o increases, the required capacity of the computing pool C can be reduced, which provides larger power to support VMs for other functions, and vice versa. On the other hand, the increased d_o also compresses the delay tolerance in other parts under the constraint of D_0. Similarly, for an arbitrary part of the network, loosing the delay requirements in this part can decrease the required capacity of the involved VMs, yet the decreased limitation of delay in the other parts may require larger capacity of VMs to meet the traffic demand. In practice, the allocation of VMs among the four parts interact with each other under the constraints of P_0 and D_0, which strongly impacts the system performance. Therefore, to maximize the system performance, the optimal trade-off between the VMs allocation should be determined by optimization techniques such as geometric programming.

Case Study

In this section, we explore the relationship between energy consumption and the maximal system tolerant delay D_0, as well as the relationship between the minimal system delay and the maximal enabled system energy consumption E_0. We will study the performance of adaptive VMs allocation mechanism via geometric programming



(a) The impact of D_0 on the system energy consumption.

(b) The impact of E_0 on the minimal system delay.

FIGURE 5. System performance under different scenarios in VCN: a) The impact of D0 on the system energy consumption; b) The impact of E0 on the minimal system delay.

compared to that of the fixed VMs allocation mechanism. The simulation is conducted on Matlab 2013b.

This section provides an example of VM allocation in a VCN with N = 19 placements points of servers. The demand of the wireless traffic at each of the N placements points of servers is set to $L_0 = 10^9$ bits, where αL_0 bits can be stored in the local CP-SIVMs. The remaining $(1 - \alpha)L_0$ bits of traffic data will be downloaded from the remote service providers. The QoS of the traffic is measured by the maximal delay tolerance D_0 where the wireless transmission of the L_0 bits of traffic will be finished at each of the N placements points of servers. In terms of CC-CIVM, we assume that the number of CPU cycles involved in the computation for networks optimization is $\aleph = 10^{12}$ cycles. In terms of EC-CIVM, we consider that at each placement point, users possess $\aleph_u = 10^{10}$ cycles for computation, where $\beta\aleph_u$ cycles of CPU will be executed in EC-CIVM, and the remaining $(1 - \beta)\aleph_u$ cycles of CPU will be executed at the user end. For simplification, optical connections are considered between the placements points of servers, and thus energy consumption involved in routing among the N placement points is ignored, but the power of traffic routing from the remote service providers with distance X = 100 km is considered as $P_r = 10^{-3}$ W/bps/km. Suppose that the transmitting power of BS is $P_{BS} = 10^{-12}$ W/Hz. The wireless bandwidth is denoted by B Hz. The power of CIVM is $P_C = 10^{-3}$ W/cps. At each placement point, the power of user computation at the user end is $P_{cu} = 5 \times 10^{-3}$ W/cps, and the static power consumed at the user end is $P_u = 5 \times 10^{-3}$ W/cps. The power of each laser is $P_l = 10^{-3}$ W, and the capacity of each laser is $C_l = 10^9$ bps. For the static circuit power, 1 cps of CIVM consumes $P_v^c = 10^{-3}$ W static power, 1 bit of SIVM consumes static power $P_v^s = 10^{-3}$ W. BS consumes static power $P_b = 2 \times 10^{-10}$ W/bps/Hz. The power of white Gauss noise is –174 dBm/Hz.

System delay can be divided into delay of network optimization (d_o), edge computing (d_e), data routing from the remote service providers (d_s), and the delay of wireless transmission (d_w). Obviously, different delay tolerance in the four segments impacts the number of VMs demanded in terms of communication, caching, and computation, which will further affect the system energy consumption. It is assumed that the amount of hardware resource is enough to support the allocation of VMs, and the aim is to minimize the system energy consumption and system delay, respectively.

In this article, we minimize the system energy consumption under the delay constraint of $d_s + d_w + d_o + d_e \le D_0$. By geometric programming, the optimization model can be easily solved. Likewise, we can formulate the optimization model of minimizing system delay with the constraint of maximal system energy consumption E and solve it by geometric programming.

In this case study, we compare the performance of two schemes of VM allocation: fixed VM allocation and adaptive VM allocation. Figure 5a shows the energy consumption for different VM allocation schemes. For the fixed VM allocation mechanism, the tolerant delay is equivalently divided into four parts for network optimization, edge computation, data routing, and wireless transmission. For the adaptive VM allocation, the allocation of VMs for minimizing system energy consumption is optimized by geometric programming [9] which provides a global optimal solution. It can be seen that the system energy consumption decreases as the delay tolerance D_0 increases. This is because a larger D_0 allows moderate data routing and wireless transmission, which supports more adaptive and appropriate tolerant delay for network optimization. Moreover, the fixed VM allocation scheme consumes more energy, compared with the adaptive allocation scheme. Another observation in Fig. 5a is that as α increases, the energy consumption decreases for both schemes, which confirms the availability of VMs for improving the performance of the networks.

In Fig. 5b, we study the performance between the minimal system delay and the maximal system energy consumption E_0. For the fixed VM allocation mechanism, E_0 is equivalently divided into four parts for network optimization, edge computation, data routing, and wireless transmission. For the adaptive VM allocation, the allocation of VMs for minimizing system delay is optimized by geometric programming [9] where E_0 can be allocated to the four parts in an adaptive manner. As shown in Fig. 5b, as E_0 increases, the minimal system delay can be cut down which indicates that a larger amount of VMs can speed up the data transmission. On the other hand, as β increases, the system delay can be decreased, since a larger amount of EC-CIVMs can alleviate the computation intensity at the user end.

As shown in the observations and discussions above, VMs based adaptive resource allocation can indeed improve the system performance regarding system energy consumption and system delay.

Research Directions

The framework of joint cooperations in VCN can effectively integrate the resources of 3C functions in the network. With the unified resources, that is, VMs, trade-offs among 3C functions can be achieved to maximize the system performance. In this section, we outline some key research issues.

First, the key issues of establishing the integrated framework of cooperations in VCN are to determine the specific expressions of functions $\varepsilon_W(\boldsymbol{W}, \boldsymbol{d}_W)$ and $\varepsilon_S(\boldsymbol{S}, \boldsymbol{Q}, \boldsymbol{d}_S)$. The two functions illustrate how the capacity of VMs, limitation of delay, the data routing, and the corresponding energy consumption interact with each other. In practice, the joint optimization of cooperations in VCN cannot be directly solved due to the complexity of functions $\varepsilon_W(\boldsymbol{W}, \boldsymbol{d}_W)$ and $\varepsilon_S(\boldsymbol{S}, \boldsymbol{Q}, \boldsymbol{d}_S)$. Therefore, the approximation methods of the functions, advanced optimization techniques, and the corresponding bounds of the system performance should be further studied.

Second, the optimal placement of the CP-SIVM based cached contents is another important issue to explore. Specifically, each server location has a demand for cooperative caching of its most needed data. The servers involved in its request of cooperative caching constitute a cooperative caching cluster. From the viewpoint of the network performance, only a part of the caching demands can be met under the limitation of CP-SIVMs capacity. How to design a dynamic caching mechanism according to the traffic demand for maximizing system performance will be an interesting problem.

Finally, in terms of the heterogeneous routing of the EE-SIVMs based wired connections, how to asynchronously choose the placement points of servers as transmitting-receiving pairs to achieve delay controllable non-blocking routing remains an open problem. Since the capacity of EE-SIVMs generally differs from each other across the placement points of servers, the maximal amount of data cached at the placement points are different. Accordingly, an asynchronous traffic scheduling and routing mechanism is required.

Conclusions

In this article, we have investigated multi-point VM cooperation in networks with communications, caching, and computing resources. We have discussed the trade-off among allocations of different resources, and proposed a joint cooperation framework based on virtual machine allocation. Essential research topics have also been provided to achieve a flexible and effective VCN. In the future, we will develop virtual machine cooperation algorithms from the viewpoint of optimization.

From the viewpoint of the network performance, only a part of the caching demands can be met under the limitation of CP-SIVMs capacity. How to design a dynamic caching mechanism according to the traffic demand for maximizing system performance will be an interesting problem.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (NSFC Project No. 61701054 and 61501065); the Fundamental Research Funds for the Central University (Project No. 106112016CDJRC000119 and 106112017CDJXY160002); and the Natural Sciences and Engineering Research Council (NSERC), Canada.

References

[1] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021 White Paper," 2017; available: https://www.cisco.com/c/en/us/solutions/collateral/serviceprovider/ visual-networking-index-vni/mobile-white-paper-c11- 520862.html.

[2] N. Cheng *et al.*, "Big Data Driven Vehicular Networks," *IEEE Network*, vol. 32, no. 6, 2018, pp. 160–67.

[3] R. Mijumbi *et al.*, "Network Function Virtualization: State-of-the-Art and Research Challenges," *IEEE Commun. Surveys & Tutorials*, vol. 18, no. 1, 2016, pp. 236–62.

[4] K. Guo, M. Sheng, and J. Tang, "Exploiting Hybrid Clustering and Computation Provisioning for Green C-RAN," *IEEE JSAC*, vol. 34, no. 12, 2016, pp. 4063–76.

[5] J. Tang, W. Tay, and T. Quek, "Cross-Layer Resource Allocation with Elastic Service Scaling in Cloud Radio Access Network," *IEEE Trans. Wireless Commun.*, vol. 14, no. 9, 2015, pp. 5068–81.

[6] S. Bassoy *et al.*, "Coordinated Multipoint Clustering Schemes: A Survey," *IEEE Commun. Surveys & Tutorials*, vol. 19, no. 2, 2017, pp. 743–64.

[7] S. Fu *et al.*, "Cross-Networks Energy Efficiency Tradeoff: From Wired Networks to Wireless Networks," *IEEE Access*, vol. 5, 2017, pp. 15–26.

[8] J. Tang, T. Quek, and W. Tay, "Joint Resource Segmentation and Transmission Rate Adaptation in Cloud RAN with Caching as a Service," *Proc. IEEE SPAWC*, 2016, pp. 1–6.

[9] E. Pateromichelakis *et al.*, "On the Evolution of Multi-Cell Scheduling in 3GPP LTE/LTE-A," *IEEE Commun. Surveys & Tutorials*, vol. 15, no. 2, 2013, pp. 701–17.

[10] Q. Chen *et al.*, "Joint Resource Allocation for Software-Defined Networking, Caching, and Computing," *IEEE/ACM Trans. Networking*, vol. 26, no. 1, 2018, pp. 274–87.

[11] N. Zhang *et al.*, "Software Defined Networking Enabled Wireless Network Virtualization: Challenges and Solutions," *IEEE Network*, vol. 31, no. 5, 2017, pp. 42–49.

[12] H. Zhang *et al.*, "Energy Efficient Subchannel and Power Allocation for the Software Defined Heterogeneous VLC and RF Networks," *IEEE JSAC*, vol. 36, no. 3, 2018, pp. 658–70.

[13] M. Chen *et al.*, "Green and Mobility-Aware Caching in 5G Networks, *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, 2017, pp. 8347–61.

[14] M. Chen and Y. Hao, "Task Offloading for Mobile Edge Computing in Software Defined Ultra-Dense Network," *IEEE JSAC*, vol. 36, no. 3, 2018, pp. 587–97.

[15] B. Wu *et al.*, "Joint Scheduling and Routing for QoS Guaranteed Packet Transmission in Energy Efficient Reconfigurable WDM Mesh Networks," *IEEE JSAC*, vol. 32, no. 8, 2014, pp. 1533–41.

Biographies

Shu Fu (shufu@cqu.edu.cn) is an associate professor in the College of Communication Engineering, Chongqing University, Chongqing, P. R. China. His research interests include the next generation of wireless networks and network virtualization, and so on.

NAN CHENG [M] (dr.nan.cheng@ieee.org) received the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Waterloo in 2016, and the B.E. and M.S. degrees from the Department of Electronics and Information Engineering, Tongji University, Shanghai, China, in 2009 and 2012, respectively. He is currently a professor with the School of Telecommunication Engineering, Xidian University, Shaanxi, China. His current research focuses on space-air-ground integrated systems, big data in vehicular networks, and self-driving systems. His research interests also include performance analysis, MAC, opportunistic communication, and the application of AI for vehicular networks.

NING ZHANG [SM] (ning.zhang@tamucc.edu) received the Ph.D. degree from the University of Waterloo, Canada, in 2015. After that, he was a postdoc research fellow at the University of Waterloo and the University of Toronto, Canada. He is now an assistant professor at Texas A&M University-Corpus Christi, USA. He serves/served as an associate editor of *IEEE Transactions on Cognitive Communications and Networking*, *IEEE Internet of Things Journal*, *IEEE Access* and *IET Communications*. His research interests include next generation mobile networks, physical layer security, machine learning, and mobile edge computing.

XIN JIAN (jianxin@cqu.edu.cn) is an associate professor at the College of Microelectronics and Communication Engineering, Chongqing University, Chongqing, P. R. China. His research interests include the next generation of wireless networks and the Internet of Things, among others.

FENG LYU (fenglv@sjtu.edu.cn) received his B.S. degree in software from Central South University in 2013. He is pursuing a Ph.D. degree in the Department of Computer Science and Engineering at Shanghai Jiao Tong University. From October 2016 to October 2017, he was a visiting Ph.D. student at the Broadband Communications Research (BBCR) group in the Department of Electrical and Computer Engineering, University of Waterloo. His research interests include vehicular ad hoc networks and cloud/edge computing.

HONG WEN [SM] (sunlike@uestc.edu.cn) is a full professor with the National Key Laboratory of Science and Technology on Communications, University of Electronic & Science Technology of China, Chengdu, P. R. China. Her research interests include communication system performance and security.

XUEMIN (SHERMAN) SHEN [F] (sshen@uwaterloo.ca) is a professor and University Research Chair, Department of Electrical and Computer Engineering, University of Waterloo. He was the associate chair for Graduate Studies from 2004 to 2008. His research focuses on resource management in interconnected wireless/wired networks, wireless network security, social networks, smart grid, and vehicular ad hoc and sensor networks. He is a Fellow of the Royal Society of Canada, the Engineering Institute of Canada, and the Canadian Academy of Engineering. He is a Distinguished Lecturer of the IEEE Vehicular Technology Society and IEEE Communications Society, and a registered professional engineer of Ontario, Canada.