This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

# Clustered Virtualized Network Functions Resource Allocation based on Context-Aware Grouping in 5G Edge Networks

Sooeun Song, Changsung Lee, Hyoungjun Cho, Goeun Lim,
and Jong-Moon Chung, *Senior Member, IEEE*

**Abstract**—With the wide spread of various smart devices and the proliferation of IoT sensors, the amount of traffic on mobile networks is rapidly increasing, and applications with extreme requirements are increasing. Network function virtualization (NFV) and mobile edge computing (MEC) are emerging as core technologies to satisfy users' real-time service demands. Adapting NFV technology to MECs allows the ability to assign cloud-computing capabilities near the base stations (BSs) of radio access networks (RANs), resulting in extremely fast service access to user equipment (UE). However, placement of virtualized network functions (VNF) within the edge network need to consider the location and the requirements of the user which change in real-time. There has been almost no consideration in the existing research on VNF resource allocation (VNF-RA) based on these aspects. Therefore, in this paper, a VNF resource allocation scheme based on context-aware grouping (VNF-RACAG) technology is proposed that enables groups (based on the geographic context of users, such as location and velocity) to compute the optimal number of clusters to minimize the end-to-end delay of network services. Then, a graph partitioning algorithm is used to minimize user movement between clusters, optimizing the data rate that users lose due to VNF migration.

**Index Terms**—Network functions virtualization (NFV), virtualized network function (VNF), mobile edge computing (MEC), 5G, VNF resource allocation, VNF migration, graph partitioning

## 1 Introduction

Due to the advent of fifth generation (5G) networks, the rapid expansion of smart devices and the explosive increase in demand for various applications, the amount of data on mobile networks is increasing and the need of a data center to analyze the large amounts of data is growing. In addition, mobile data consumption is exponentially growing, where several types of services that use augmented reality (AR), virtual reality (VR), high definition (HD) or ultra high definition (UHD) video streaming require seamless and always-on network connectivity as well as multimedia oriented connectivity. According to Cisco, mobile traffic is expected to increase by more than 50% every year from 2015 to 2020, up to 8 times [1].

However, in the conventional network structure, network services are provided through application-specific hardware equipment, which cannot smoothly cope with various types of demands from an exponentially increasing number of services. In order to cope with new and various services based on the existing network structure, it is necessary to install and manage new hardware equipment suitable to support each service, which causes a rapid increase in operational expenditure (OPEX) as well as capital expenditure (CAPEX). In addition, the load of the data center in the core network is rapidly increasing due to significant increases in mobile and internet of things (IoT) application service traffic.

The convergence of network function virtualization (NFV) and mobile edge computing (MEC) can be a solution to meet the extreme requirements of new and diverse applications, where NFV and MEC are the key technologies that enable 5G network services. Due to advances in software and hardware technologies, NFV has emerged as a technology that enables network operators to design a network that consolidates a wide range of networking equipment using commodity standard high volume servers, switches, and storage [2]. In NFV, network functions that have operated on specific-purpose hardware, are virtualized, managed by software, and executed on general purpose commodity hardware. This allows modularization and isolation of each virtualized network function (VNF) (e.g., firewalls, caches, proxies, and WAN accelerators), so they can be managed independently [3]. Therefore, the rapid changing needs of users can be accommodated by installation and replacement of VNFs in commodity servers without the need for installation of new equipment. Network resource sharing not only reduces costs for deployment of new hardware equipment, but also reduces operational costs by taking advantage of the higher uniformity of the physical network platform and its homogeneity to other support platforms [2].

The implementation of VNF as a software entity that runs over the NFV infrastructure (NFVI) includes the physical resources (e.g., computing, networking, and storage resources) and how these can be virtualized [4]. Then, this is managed through NFV management and orchestration (NFV MANO), which is composed of 3 functional blocks:

- *S. Song, C. Lee, and J.-M. Chung are with the School of Electrical and Electronic Engineering, Yonsei University, Seoul, 03722, Republic of Korea.*
  *E-mail: {synice1224, onlyvine, jmc}@yonsei.ac.kr*
- *H. Cho and G. Lim are with the 5G R&D Group, Samsung Electronics, Co., Ltd., Suwon 16677, Republic of Korea.*

*Manuscript received June 19, 2018;*

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

NFV orchestrator (NFVO), VNF manager (VNFM), and virtualized infrastructure manager (VIM). NFV MANO is responsible for orchestration and lifecycle management of network services, physical/virtual resources to support the virtualization of infrastructure, and VNF lifecycle management. That is, NFV MANO focuses on all the virtualization-related management tasks required by the NFV framework.

MECs provide fast service support and cloud-computing capabilities at the edge of the network near the mobile subscribers [5]. The edge of the network may refer to the base stations (BSs), and data centers close to the radio access network (RAN). The service environment of the MEC is characterized by low latency, proximity, high bandwidth, and real-time insight into RAN information and location awareness. Mobile applications and services can be allocated at a BS or a small cell close to the user's current position. In addition, the MEC not only reduces communication delay, but also reduces traffic congestion in networks because much of the cloud-computing core network traffic can be distributed to the RAN edge [6].

NFV and MECs are mutually complementary. By deploying VNFs on a BS, a BS can enable users to receive network services from a much closer location. This enhances management flexibility and network scalability, as well as reduces backhaul overloading. However, VNF resource allocation (VNF-RA) in the edge of the network requires more considerations compared to VNF-RA in the data center. Available resources (e.g., storage, networking, and computing) in edge networks have more limited resources than data centers connected to the core network [7]. It is expected that the VNFs of the MEC will be smaller so that they are more specialized for individual users, while traditional VNFs deal with aggregated traffic from many users [8].

Due to the fact that applications have different requirements that need to be supported by the edge network resources, 5G usage scenarios are divided into three types, which are enhanced mobile broadband (eMBB), ultra-reliable and low latency communications (URLLC), and massive machine type communications (mMTC) [9]. Within usage scenario types, grouping of user applications based on application characteristics can be conducted to provide an improved performance in a more efficient manner. For example, edge cloud supported AR services supporting an individual user at a fixed location (or slowly moving) would have different service support requirements compared to the AR services used on commercial trucks of a logistics company, although both would need 5G URLLC support.

In this paper, a clustered VNF resource allocation scheme based on context-aware grouping in edge networks (VNF-RACAG) is proposed where the optimal number of clusters of each user group is derived to minimize the average end-to-end delay in edge networks, where an user group is defined as a set of users with similar speed, and a cluster is defined as a set of edge network resources allocated for a user group. In edge networks, the traffic characteristics of individual user applications have a strong influence on the edge network resource allocation. Therefore, the VNF-RACAG algorithm reduces the delay based on user application characteristics by grouping users that have similar traffic patterns.

The sections of this paper are organized as follows. The motivation and the contributions of the proposed VNF-RACAG scheme is presented in section II. Then, section III presents a summary of related research, section IV describes the network model, section V explains the operational mechanism of the proposed VNF-RACAG scheme, section VI provides the performance analysis, and the paper's conclusion is presented in section VII.

## 2 Motivation and Contribution

Supporting the network operations for 5G communications, NFV and MECs are regarded as key technologies, and researches to allocate VNFs or applications through virtualization of edge network resources have been continuously proposed [8], [10], [11], [12], [13]. While existing VNFs are allocated for services by large users, VNFs in the MEC environment are expected to support services for single or small group users [8]. Each user requires specific VNFs to execute its applications in an edge cloud. More specifically, each user receives network services based on specific VNFs, where a VNF is only used by one user. When a user with mobility requests a service, it is necessary to consider the network service migration according to the movement of the user. However, VNF migration causes additional traffic to the network, which causes consumption of network edge resources [14]. On the other hand, when VNF migration is not used (i.e., keeping the VNFs in their first allocated server) the physical distance between the user increases causing a performance degradation due to an increased number of hops. Such issues lead the need for a new edge network resource allocation scheme capable of actively responding to mobile users and network services requested in real-time.

In this paper, the proposed VNF-RACAG scheme assigns users to different groups according to their speed. Then, clustering is conducted on the resources of the network edge (e.g., CPU cores of the MEC server) for each user group. When an user moves across the boundary to another cluster, VNF migration will be executed by applying the criteria for VNF migration to the boundary between the clusters. The optimal number of clusters of each user group is derived to minimize the average end-to-end delay in edge networks. Then, clustering is performed based on the graph partitioning algorithm according to the calculated optimal number of clusters. The original contributions of this paper can be summarized as follows.

- The self-similar stochastic model of the traffic is derived for VNF-RA in 5G edge networks, which simplifies the mixed integer linear programming (MILP) using the average-case for the network resource usage, thereby reducing the long computation time and the resulting scalability limitations of MILP.
- While most NFV researches [3], [15], [16] consider VNF-RA in the data center for network services of large-scale traffic, VNF-RA based on MEC networks is presented, which considers user's mobility.
- The VNF-RACAG scheme takes into account the different groups of users (according to their speed and the requested services, but other criteria are also possible), since different groups may change BS connectivity with different probabilities, thus requiring a different allocation to balance the cost of migrating

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

the VNFs and the closeness between the users and the VNFs.

- The VNF-RACAG scheme is different from existing schemes based on the fact that the traffic characteristics of the end user is included into the optimization process to maximize the overall performance.

## 3 Related Works

Several NFV and MEC schemes have been proposed in recent articles. As virtualization is an efficient and cost-effective strategy to exploit and share physical network resources, it has received much attention.

In several papers that focus on NFV, the problem of finding an optimum embedding topology of virtual links and nodes onto a given physical substrate network is referred to as the VNF-RA problem [3], [10], [15], [16], where network services consist of a chain of network functions, called a service function chain (SFC). The VNF-RA problem is a problem of choosing the suitable location to allocate the SFC and its links to connect it to the network infrastructure. At the same time, resource optimization must be accompanied to meet specific objectives (e.g., maximizing remaining network resources, minimizing the number of used nodes, minimizing the latency, etc.) [3].

In [15], a model that formalizes the chaining of VNFs is presented using a context-free language, where the VNF-RA problem solution is derived using a mixed integer quadratically constrained program that finds the placement of the network functions. In order to maximize the utilization of the remaining data rate and minimize the number of used nodes and path latency in service chaining networks using NFV, Pareto optimization is applied. In [16], the problem of optimal VNF placement is formulated using binary integer programming placement in packet/optical data centers. The overall number of optical/electronic/optical conversions can be minimized and an efficient heuristic algorithm is proposed to solve the problem. Most of the researches dealing with the VNF-RA problem (including [15] and [16]) are focused on how to allocate pre-defined network functions supporting large-scale traffic of many users in the data center. To solve the VNF-RA problem, a common assumption applied is that network services are pre-defined and fixed, which is why they are not sufficient to be used as a VNF-RA solution for real-time mobile network services in edge networks.

In [10], the wireless VNF placement problem in RANs is proposed as an MILP problem. However, the authors point out that the ILP problem is impractical due to its limited scalability and long computation time, and suggests a heuristic algorithm called WiNE. WiNE is composed of three steps as computing the list of candidate nodes, sorting in decreasing order, and allocating the nodes and links. However, still the VNF-RA technique is used for pre-determined VNFs, and the mobility of the user is not considered.

Several papers have also focused on placement of appropriate virtual machines (VMs) with MECs according to user movement characteristics. In [17], an algorithm enabling flexible selection of communication paths with VM placement (PSwH) based on the prediction of user movement is proposed. The prediction is used for dynamic VM placement and to find the most suitable communication path according to expected user movement. The authors of paper [18] formulate the VMs allocation problem as a Markov decision process (MDP) to determine where user VMs should be allocated across the different BSs to optimize the performance.

Since the convergence of MECs and NFV can result in a viable solution to the VNF-RA problem in the edge network, several papers have focused on the convergence of MEC and NFV [11]. In [8], a container-based NFV platform is proposed which orchestrates lightweight container VNFs and identifies the virtualization at the network edge using three examples of the platform (i.e., IoT DDoS remediation, on-demand troubleshooting, and roaming of VNFs). One of the arguments of [8] is that VNFs on edge networks can effectively manage a small amount of traffic for a small group, while conventional VNFs are appropriate to manage aggregated traffic for many users. The convergence of NFV and MEC is focused on the network architecture and platform implementation, and the VNF-RA problem at the actual network edge is not considered.

## 4 Network Model

The network model is assumed to be composed of a set of $\mathbf{M}$ BSs, in which $m \in \mathbf{M}$, and $L$ bidirectional transmission links that connect between the BSs. Cloud function equipped BSs can act as MEC servers in which VNFs can be allocated. In 5G networks, a BS would be implemented as a gNB. It is assumed that RAN user set $\mathbb{U}$ consists of user nodes $u \in \mathbb{U}$ that request for specific network services for each user while in movement [8], and the service request process of the users follow a Poisson distribution with rate $\lambda$, and the total number of network services is $S$. Network services consist of consecutive VNFs called SFCs, and virtual links are used to connect the SFCs. VNFs need to be allocated as actual physical resources to provide services.

Users are connected to the RAN through the mobile device's serving BS. If all VNFs of the requested service are placed on the serving BS, the traffic does not consume any link resources in the RAN. However, when the requested VNFs are allocated in other BSs in the RAN, the traffic needs to be transmitted through the link to the BS where the requested VNFs are allocated. Unlike static users, it is assumed that mobile users change their position according to their movement, and the replacement of a serving BS is updated through a general handover process between BSs. Then, the VNFs of the service are migrated to a new serving BS or other appropriate MEC servers. In addition, the traffic path is updated to provide continuous services [17]. It is assumed that the migration of VNFs is based on hot migration, in which no additional delay occurs prior to migration of a VNF since it is exchanged through an additional separate link [14]. It is assumed that a single path with sufficient resources will be used for VNF migration, or multiple paths may be used to distribute the resources for VNF migration.

Fig. 1 represents the change of the serving BS and the VNF migration according to the movement of the user equipment (UE). As the UE moves, the network service

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

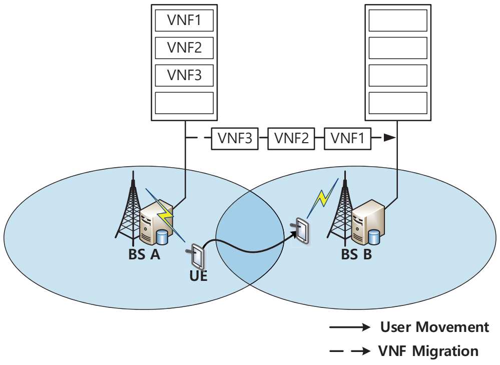


Fig. 1. Virtualized network function migration model.

functions for the user should be migrated to the optimal location. Each BS can serve as a server, and VNFs can be placed in this server. It is assumed that the UE is requesting the network service composed of VNF 1, VNF 2, and VNF 3. When the UE moves from BS A to BS B, the serving BS of the UE changes to BS B according to the general handover process, and the requested VNFs (VNF 1, 2, and 3) of the UE are also migrated to BS B if the resources of BS B are sufficient to allocate every VNF as shown in Fig. 1. Otherwise, if there are not enough resources in the new serving BS, the VNFs can be allocated in adjacent servers that have sufficient resources.

To allocate the VNF, the required resources may include computation (CPU cores), storage, networking, etc. Since CPU cores are more limited resources than other resources, it is assumed that the CPU cores of the MEC server and the link capacity of the RAN have limits in the VNF-RACAG [19]. However, other resource types, such as storage resources, can be added as additional constraints. In reality, it is not possible to provision all needed VNFs within one MEC server due to various resource constraints. The resources in an individual MEC server are limited and can support a constrained number of VNFs with moderate needs for such resources [7]. Therefore, in some cases, it is necessary to provide network services through multiple relaying nodes.

In order to provide network services in NFV, it is necessary to allocate physical resources (e.g., MEC servers, links, and paths) for deployment of VNFs. VNF embedding is comprised of CPU assignment and link assignment. When the CPU of the $i$th VNF for the service requested by user $u$ is $C_{u,i}$, and $x_{u,i,m}$ is a binary variable which is 1 if the VNF is deployed on MEC server $m$, the remaining CPU resources $C^m_{rem}$ of MEC server $m$ can be described as

$$C^m_{rem} = C^m_{total} - \sum_{u\in\mathbb{U}}\sum_{i} x_{u,i,m}C_{u,i} \tag{1}$$

where $C^m_{total}$ is the total CPU capacity of server $m$. That is, as the VNFs are allocated to multiple users through a single MEC server, or a VNF requiring a large number of CPU cores is allocated, the available CPU computing resources of the MEC server are reduced. In addition, the link capacity is assigned considering the links for service and the links for VNF migration. When the service link capacity and migration capacity of the $i$th VNF for user $u$ are $r_{u,i}$ and $r_{mig,u,i}$, respectively, the remaining link capacity $r^{m,m'}_{rem}$ between $m$ and $m'$ can be described as

$$\begin{aligned} r^{m,m'}_{rem} = r^{m,m'}_{total} &- \sum_{u\in\mathbb{U}}\sum_{i} y_{u,i,m,m'}r_{u,i} \\ &- \sum_{u\in\mathbb{U}}\sum_{i} z_{u,i,m,m'}r_{mig,u,i} \end{aligned} \tag{2}$$

where $r^{m,m'}_{total}$ is the total link capacity between $m$ and $m'$. In (2), $y_{u,i,m,m'}$ and $z_{u,i,m,m'}$ are binary variables that are 1 if the link between $m$ and $m'$ is used as a part of the path for service or migration, respectively (and are otherwise 0). Similar to the remaining CPU computational resources, the remaining link resources are reduced when one link is selected as a service or a migration path by multiple users, or as a path of services or migrations requiring high link capacity.

The proposed VNF-RACAG scheme was designed to minimize the end-to-end delay through optimization of VNF placement in the edge network. Actual multiplexed network traffic is commonly self-similarity, and therefore, control models based on the self-similarity of traffic can result in more efficient and effective results. In this paper, a stochastic self-similarity model was derived and used in the computation of the end-to-end delay. It is assumed that the inflow traffic of the MEC network has self-similar characteristics, and the traffic model is designed based on a fractional Brownian motion (fBm) for effective self-similar traffic modeling [20].

## 5 VNF-RACAG Scheme Description

In this section, details of the VNF-RACAG scheme are presented. In the model, it is assumed that there are $M$ BSs equipped with a MEC server and $U$ users. Each user is moving within a RAN supported cell at a constant velocity and requests for non-overlapping network services. It is assumed that the CPU computation capacity and link capacity in the RAN are sufficient such that every requested service can be processed in the RAN, and network service $S_u$ is requested by user $u$. $S_u$ is composed of a VNF chain $\{f_{u,1}, f_{u,2}, ...\}$ which forms the SFC [2]. That is, it is assumed that the load of the VNFs should not exceed the overall capacity of the MEC servers. However, in an actual network, there may be an overload condition that requires more VNFs than capacity of the MEC servers. In [21], a novel overload control framework is proposed to protect NFV services from short term overloads by filtering the traffic. However, the overload condition is a different issue that should be dealt with through a complementary approach, which is beyond the scope of this paper.

In this paper, users form a group based on similar characteristics (e.g., mobility), and the network resources are allocated according to the size of each group. The network resources allocated to each group are divided into the optimal number of clusters for each user group, which minimizes the end-to-end delay of each group. When allocating server resources to a user group, it is assumed that the group with more users has a higher priority in the resource distribution. In some cases, it may be preferable to allocate all CPU resources of a BS to a large number of users

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

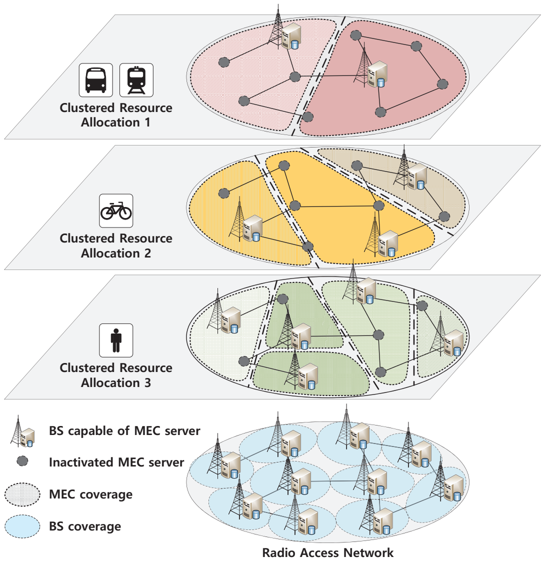

Clustered Resource Allocation 1
Clustered Resource Allocation 2
Clustered Resource Allocation 3
BS capable of MEC server
Inactivated MEC server
MEC coverage
BS coverage
Radio Access Network

Fig. 2. Resource allocation group models.

TABLE 1
Parameters and variables

| Parameters | Description |
|---|---|
| $U$ | Total number of users |
| $M$ | Total number of MEC servers in the RAN |
| $S$ | Total number of network services in network |
| $S_u$ | Network service required by $u$ and composed with SFC $\{f_{u,1},\ f_{u,2},\ \cdots\}$ |
| $C_{u,i}$ | Required CPU resources in $f_{u,i}$ |
| $r_{u,i}$ | Required link capacity to service $f_{u,i}$ |
| $r_{mig,u,i}$ | Required link capacity to migrate $f_{u,i}$ |
| $N$ | Number of user groups |
| $\mathbb{U}_n$ | The $n$th user group based on the user's velocity |
| $U_n$ | Number of users belonging to user group $\mathbb{U}_n$ |
| $M_n$ | Average number of MEC servers assigned to $\mathbb{U}_n$ |
| $\mathbf{k}$ | Set of the number of clusters and composed of $\{k_1,\cdots,k_N\}$ |
| $k_n$ | Number of clusters for $\mathbb{U}_n$ |
| $v_n$ | Maximum velocity of $\mathbb{U}_n$ |
| $S_n$ | Total number of network services of $\mathbb{U}_n$ |
| $S_{mig,n}$ | Number of migrated services of $\mathbb{U}_n$ |
| $\lambda$ | Average service request rate |
| $p_n$ | Probability that user $u$ will belong to $\mathbb{U}_n$ |
| $p_{mig,n}$ | Migration probability of $\mathbb{U}_n$ |
| $\mu_{M_n}$ | Server boundary crossing rate of $\mathbb{U}_n$ |
| $\mu_n$ | Cluster boundary crossing rate of $\mathbb{U}_n$ |

| Variables | Description |
|---|---|
| $x_{u,i,m}$ | 1, if $f_{u,i}$ is deployed in $m$<br>0, otherwise |
| $y_{u,i,m,m'}$ | 1, if link $(m,\ m')$ is by $f_{u,i}$<br>0, otherwise |
| $z_{u,i,m,m'}$ | 1, if the migration path of $f_{u,i}$ passes link $(m,\ m')$<br>0, otherwise |

in one group. After the initial resource assignments, only the remaining resources of a MEC server may be allocated. That is, most MEC servers may be assigned to a single large group, while a few MEC servers may distribute their resources across multiple groups. It is assumed that when a user moves within a cluster, VNF migration does not occur, but VNF migration will occur when an user enters another cluster beyond the boundaries of its cluster.

Fig. 2 illustrates an example of a cluster-based network resource allocation model. The network physical resources are deployed throughout the RAN, which is composed of physical nodes and links. There are 10 BSs equipped with MEC servers, which are randomly distributed through the RAN, and the MEC servers are connected through multiple physical links. The transmission coverage of each BS is represented by light blue circles, denoted as "BS coverage." After the clustering for each user group, the boundaries of the clusters are represented by dashed lines and the coverage of the MEC servers is represented by dotted lines denoted as "MEC coverage." In Fig. 2, it is also assumed that there are three user groups in the network, which are vehicle, bicycle, and static (including slowly nomad) users, respectively with 20%, 30%, and 50% of all users in each group. It is also assumed that network resources are allocated proportional to the size of the group, then the server resources in the network are assigned proportionally to the groups based on a 20%, 30%, and 50% ratio, respectively. The requested VNFs are allocated to the MEC servers assigned to the corresponding user group. The MEC servers assigned to another group will have the unused functions of the MEC server disabled and only serve as a BS transmitting traffic. In addition, when the optimal number of clusters in each group is 2, 3, and 4, clustering is performed for each user group to minimize the average end-to-end delay by solving a graph partitioning problem.

### 5.1 User Grouping Model Based on Context

In this subsection, the context-aware user grouping model is presented. Although there are many characteristics of the user (i.e., service preference, location, history, channel property, etc.), the geographic context is analyzed as the main context of this paper. In wireless communications, geographic context refers to the location of the mobile users and speed information through past and present position comparisons. However, other types of context, such as charging policy, preference service type, etc., can also be used to set the grouping model using the proposed scheme.

It is assumed that each users' location is periodically reported to the orchestrator in the RAN by monitoring the locations of the users in real-time. It is assumed that the speed of an individual user is constant and follows the Erlang distribution [22]. The orchestrator assigns each user to one of $N$ groups based on the reported user's velocity, where the set of the user group is $\{\mathbb{U}_1, \mathbb{U}_2, \cdots, \mathbb{U}_N\}$ [23].

A service provider can prioritize the quality of service according to the priority of each group. The service provider decides the priorities on the status of the network. The priority of each group can be determined by the size of the group, the handover rate, the service variability, the characteristics of the service, and the charging policy. In this paper, it is assumed that when a larger number of users belong to a group, the priority of the group is higher. Network resources are then allocated from the highest priority group in a descending priority order.

Each user can belong to only one group, and the sum of the number of users belonging to each group is equal to the total number of users in the RAN. That is, when the total number of users in the RAN is $U$ and the size of group $\mathbb{U}_n$ is $\mathrm{N}(\mathbb{U}_n) = U_n$, then $\sum_{n=1}^{N} U_n = U$. It is assumed that the representative velocity $v_n$ of $\mathbb{U}_n$ is set to

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

the maximum velocity of $\mathbb{U}_n$. It is assumed that the CPU resource allocation of the server in the RAN is proportional to the size of each user group. When the total number of servers in the RAN is $M$, and the CPU of each server is the same, the average number of MEC servers allocated to the user group $\mathbb{U}_n$ is $M_n = MU_n/U$.

It is assumed that $M$ servers are randomly distributed in a circular area where the size of the total network is $a$, and the coverage of each server also forms a circular area. Then, the average coverage area of each server for $\mathbb{U}_n$ can be described as $a_{M_n} = a/M_n$. If a user is out of this range, then the user will need to change its connection to a different server. Based on [24], the server boundary crossing rate of $\mathbb{U}_n$ can be described as (3).

$$\mu_{M_n} = \frac{2v_n}{\sqrt{\pi a_{M_n}}} = \frac{2v_n\sqrt{M_n}}{\sqrt{\pi a}} \tag{3}$$

According to (2), frequent VNF migration consumes additional link resources which degrade the performance of the service. On the other hand, relaying the traffic path continuously without migration deteriorates the performance due to the long physical distance between the required VNFs and the user. In addition, it is possible to weaken the advantage of the MEC by lowering the effect of the traffic distribution.

The clustered VNF allocation model (based on user grouping) is described in the following. If the user moves within the cluster, it does not carry out VNF migration and prevents the network from consuming additional link resources for migration. However, if the user moves between clusters, VNF migration is performed immediately, and the distance between the user and the service is maintained to keep the physical distance short. It is assumed that the shortest path between the serving server and the target server is used when service connection is provided. If the set of number of appropriate clusters is $\mathbf{k} = \{k_1, k_2, \cdots, k_N\}$ and $k_n$ is the number of clusters of $\mathbb{U}_n$, then the cluster boundary crossing rate (i.e., rate that an user goes out of the cluster) is expressed by (4) based on (3),

$$\mu_n = \frac{2v_n}{\sqrt{\pi a_n}} = \frac{2v_n}{\sqrt{\pi \frac{a}{k_n}}} = \frac{2v_n\sqrt{k_n}}{\sqrt{\pi a}} \tag{4}$$

where $a_n$ is the average cluster size of $\mathbb{U}_n$. It is assumed that the residence time in coverage of a server and cluster follow an exponential distribution with boundary crossing rates $\mu_{M_n}$ and $\mu_n$, respectively, while the service request follows a Poisson distribution with rate $\lambda$. The cluster boundary crossing probability can be obtained as (5).

$$p_{mig,n} = \frac{\mu_n}{\mu_n + \lambda} \tag{5}$$

If the total number of network services requested by $\mathbb{U}_n$ is $S_n = SU_n/U$, the number of network services which need to migrate can be described as

$$\begin{aligned} S_{mig,n} &= S_n p_{mig,n} \qquad (6)\\ &= \frac{2Sp_n v_n\sqrt{k_n}}{2v_n\sqrt{k_n} + \lambda\sqrt{\pi a}} \end{aligned}$$

where $p_n = U_n/U$, which is the probability that user $u$ will belong to group $\mathbb{U}_n$. In (6), $S_{mig,n}$ has a tendency to increase when the user's speed $v_n$ is fast and the number of clusters $k_n$ increases, and vice versa.

## 5.2 Optimal Number of Clusters

The optimization problem statement for end-to-end delay minimization is presented in (7), which is decided by the set of the optimal number of clusters $\mathbf{k} = \{k_1, k_2, \cdots, k_N\}$, in which $k_n$ is the number of clusters of $\mathbb{U}_n$. $T(\mathbf{k})$ is the end-to-end delay of the flow.

$$\text{minimize} \quad T(\mathbf{k}) \tag{7}$$

subject to:

$$\sum_{n=1}^{N} U_n = U \tag{8a}$$

$$1 \le k_n \le M_n \tag{8b}$$

$$C_{total}^{m} \ge \sum_{u\in\mathbb{U}}\sum_{i} x_{u,i,m}C_{u,i} \tag{8c}$$

$$r_{total}^{m,m'} \ge \sum_{u\in\mathbb{U}}\sum_{i} y_{u,i,m,m'}r_{u,i} + \sum_{u\in\mathbb{U}}\sum_{i} z_{u,i,m,m'}r_{mig,u,i} \tag{8d}$$

$$x_{u,i,m}, y_{u,i,m,m'}, z_{u,i,m,m'} \in \{0,1\} \tag{8e}$$

$$k_n \in \mathbb{N} \tag{8f}$$

The meaning of the constraints are as follows. Every user should belong to exactly one group (8a). For a given user group, the number of clusters cannot be set larger than the average number of servers assigned to that group (8b), and thus, one server can not belong to more than one cluster. Based on (1) and (2), the remaining network resources must be larger than zero, resulting in constraints (8c), (8d), and (8e). In addition, the number of clusters should be a natural number (8f).

Assuming the average inflow traffic rate for network services is $\mathrm{E}[r_{u,i}] = r$ bps, the maximum link delay of self-similar traffic between adjacent nodes $m$ and $m'$ can be modeled as (9), based on [12],

$$d_{m,m'}(k_n) = \omega^{-1}\left(\frac{r_{rem}^{m,m'} - r}{H}\right)^{\frac{2H}{2H-2}} \tag{9}$$

where $\omega = (-2\sigma^2\log\varepsilon)^{1/(2H-2)}/(1-H)$. In (9), $H$ is the Hurst parameter which indicates the degree of self-similarity of a time series, $\sigma$ is the average standard deviation of the flow, and $\varepsilon$ is the overflow probability of the self-similar fBm traffic model [20]. Then, the delay $T(\mathbf{k})$ is obtained by averaging the end-to-end delay in each group $T_n(k_n) = \frac{1}{U_n}\sum_{u\in\mathbb{U}_n}\sum_{(m,m')\in\text{Path}_u} d_{m,m'}(k_n)$ where $\text{Path}_u$ is the path for $S_u$.

$$\begin{aligned} T(\mathbf{k}) &= \sum_{n=1}^{N} p_n T_n(k_n) \qquad (10)\\ &= \frac{1}{U}\sum_{n=1}^{N}\sum_{u\in\mathbb{U}_n}\sum_{(m,m')\in\text{Path}_u} d_{m,m'}(k_n) \end{aligned}$$

It is assumed that the average number of VNFs to provide service is $f$, and average link capacity to migrate a VNF is

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

$r_{mig}$. The total number of network services which need to migrate can be expressed as $S_{mig} = \sum_{n=1}^{N} S_{mig,n}$. Based on (2), (4), and (6), the average available link capacity $\tilde{r}_{rem}$ can be described as (11).

$$\begin{aligned}\tilde{r}_{rem} &= \mathrm{E}\left[r_{total}^{m,m'} - \sum_{n=1}^{N}\sum_{u\in\mathbb{U}_n}\sum_{i} z_{u,i,m,m'} r_{mig,u,i}\right] \qquad (11)\\ &= r_{total} - \frac{hS_{mig}f}{M} r_{mig}\\ &= r_{total} - \frac{hr_{mig}f}{M}\sum_{n=1}^{N} S_n p_{mig,n}\\ &= r_{total} - \frac{hr_{mig}Sf}{M}\sum_{n=1}^{N} p_n p_{mig,n}\end{aligned}$$

To complete the VNF migration, the average path length is the same as the average hop count in cluster $h_n$, and all links are available to all groups in the network and can be bidirectional.

It is assumed that the average hop count is proportional to the linear distance between the two servers. Thus, for an average number of hop counts $h$ between two servers in the network, the average hop count between two nodes in the cluster can be expressed as $h_n = h/\sqrt{k_n}$. For an average remaining link capacity $\tilde{r}_{rem}$, the end-to-end delay $T(\mathbf{k})$ can be approximated by the average end-to-end delay $\tilde{T}(\mathbf{k})$ as in (12).

$$\begin{aligned}\tilde{T}(\mathbf{k}) &= \sum_{n=1}^{N} \frac{hp_n}{\sqrt{k_n}} \tilde{d}(k_n) \qquad (12)\\ &= \frac{h_f h}{\omega}\left(\frac{\tilde{r}_{rem} - r}{H}\right)^{\frac{H}{H-1}} \sum_{n=1}^{N} \frac{p_n}{\sqrt{k_n}}\end{aligned}$$

The simplified model $\tilde{T}(\mathbf{k})$ enables formulation of a convex optimization problem, where the degree of margin between $T(\mathbf{k})$ and $\tilde{T}(\mathbf{k})$ is analyzed in Lemma 1 [25].

**Lemma 1.** *The difference between $T(\boldsymbol{k})$ and $\tilde{T}(\boldsymbol{k})$ is bounded by a finite upper bound.*

*Proof.* Based on $\mathbf{k} = [k_1,\ k_2,\ \cdots,\ k_N]$ and $\tilde{r}_{rem}$, as $H/(H-1) < 0$, and $T(\mathbf{k}) \leq \frac{n_f h}{\omega}\left(\frac{r_{min}-r}{H}\right)^{\frac{H}{H-1}} \sum_{n=1}^{N} \frac{p_n}{\sqrt{k_n}}$, where $r_{min}$ and $r_{max}$ are the minimum and the maximum available link capacity of the network, respectively. Thus, $T(\mathbf{k}) - \tilde{T}(\mathbf{k})$ is bounded by $\frac{n_f h}{\omega}\left[\left(\frac{r_{min}-r}{H}\right)^{H/(H-1)} - \left(\frac{r_{max}-r}{H}\right)^{H/(H-1)}\right] \sum_{n=1}^{N} \frac{p_n}{\sqrt{k_n}}$. ■

As the end-to-end delay can be approximated by $\tilde{T}(\mathbf{k})$, the optimal number of clusters that minimizes the end-to-end delays can be obtained by solving a minimization problem of function $\tilde{T}(\mathbf{k})$ if $\tilde{T}(\mathbf{k})$ is convex. Then, the function $\tilde{T}(\mathbf{k})$ is convex as shown in Lemma 2.

**Lemma 2.** *The function $\tilde{T}(\boldsymbol{k})$, defined on a convex set $\Omega \subset \mathbb{N}^N$ is convex, and satisfies the convex condition $\tilde{T}(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) \leq \alpha\tilde{T}(\boldsymbol{x}) + (1-\alpha)\tilde{T}(\boldsymbol{y})$ if for $\forall\ \boldsymbol{x}, \boldsymbol{y} \in \Omega$ and $\alpha \in (0, 1)$.*

*Proof.* The left term in the inequality above can be described as in (13).

$$\begin{aligned}\tilde{T}(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) =& \frac{n_f h}{\omega}\left(\frac{\tilde{r}_{rem} - r}{H}\right)^{\frac{H}{H-1}} \qquad (13)\\ &\cdot \sum_{n=1}^{N} \frac{p_n}{\sqrt{\alpha x_n + (1-\alpha)y_n}}\end{aligned}$$

The right term of (13) can be expressed as in (14).

$$\begin{aligned}\alpha\tilde{T}(\mathbf{x}) + (1-\alpha)\tilde{T}(\mathbf{y}) =& \alpha\left[\frac{n_f h}{\omega}\left(\frac{\tilde{r}_{rem} - r}{H}\right)^{\frac{H}{H-1}} \sum_{n=1}^{N} \frac{p_n}{\sqrt{x_n}}\right] \qquad (14)\\ &+ (1-\alpha)\left[\frac{n_f h}{\omega}\left(\frac{\tilde{r}_{rem} - r}{H}\right)^{\frac{H}{H-1}} \sum_{n=1}^{N} \frac{p_n}{\sqrt{y_n}}\right]\\ =& \frac{n_f h}{\omega}\left(\frac{\tilde{r}_{rem} - r}{H}\right)^{\frac{H}{H-1}} \sum_{n=1}^{N} p_n\left(\frac{\alpha}{\sqrt{x_n}} + \frac{1-\alpha}{\sqrt{y_n}}\right)\end{aligned}$$

Based on $n_f > 0$, $h > 0$, $\omega > 0$, $H > 0$, and $\tilde{r}_{rem} > r$, then $\frac{n_f h}{\omega}\left(\frac{\tilde{r}_{rem}-r}{H}\right)^{\frac{H}{H-1}} > 0$. Since the function $\frac{1}{\sqrt{x}}$ is convex in the positive real number range, $\frac{1}{\sqrt{\alpha x_n + (1-\alpha)y_n}} < \left(\frac{\alpha}{\sqrt{x_n}} + \frac{1-\alpha}{\sqrt{y_n}}\right)$ for every $x$, $y$, and $n$. Therefore, the function $\tilde{T}(\mathbf{k})$ satisfies the convex condition. ■

From (12), the set of optimal number of clusters $\mathbf{k}_{opt}$ can be calculated using the gradient descent method, where the starting point can be selected as the optimum solution for the previous iteration, based on

$$\mathbf{k}^{+} = \mathbf{k} - \ \mathbf{w} \circ \frac{\partial\tilde{T}(\mathbf{k})}{\partial\mathbf{k}} \qquad (15)$$

where $\mathbf{w} = [w_1,\ w_2,\ \cdots\ ,\ w_N]$ is a vector of weights for each element of the gradient, and $\circ$ is the Hadamard product which multiplies each component of two matrices of the same size. $\tilde{T}(\mathbf{k})$ is expressed as (16), where its elements are described as in (17),

$$\frac{\partial\tilde{T}(\mathbf{k})}{\partial\mathbf{k}} = \left[\frac{\partial\tilde{T}(\mathbf{k})}{\partial k_1},\ \frac{\partial\tilde{T}(\mathbf{k})}{\partial k_2},\ \cdots\ ,\ \frac{\partial\tilde{T}(\mathbf{k})}{\partial k_N}\right] \qquad (16)$$

$$\begin{aligned}\frac{\partial\tilde{T}(\mathbf{k})}{\partial k_n} =& \frac{h}{\omega}\frac{bH}{1-H}\frac{p_n c}{2\sqrt{k_n}(\sqrt{k_n}+c)^2} \qquad (17)\\ &\cdot\left(a - b\sum_{m=1}^{N}\frac{p_m\sqrt{k_m}}{\sqrt{k_m}+c}\right)^{\frac{1}{H-1}} \sum_{m=1}^{N}\frac{p_m}{\sqrt{k_m}}\\ &- \frac{h}{\omega}\left(a - b\sum_{m=1}^{N}\frac{p_m\sqrt{k_m}}{\sqrt{k_m}+c}\right)^{\frac{H}{H-1}}\left(\frac{p_n}{2k_n\sqrt{k_n}}\right)\end{aligned}$$

where $a = \frac{r_{total}-r}{H}$, $b = \frac{hr_{mig}Sf}{MH}$, and $c = \frac{\lambda\sqrt{\pi a}}{2v_n}$. Before performing the gradient descent, it is necessary to verify that the function converges. The area of the variable where the gradient descent converges is identified in Lemma 3.

**Lemma 3.** *When the domain of $k_n$ is a positive integer in $[1, M_n]$ and the partial derivative $\frac{\partial\tilde{T}(k)}{\partial k_n}$ is a function of $k_n$, $f(k_n)$, if $r \leq \tilde{r}_{rem} - \frac{hr_{mig}S_n f}{M}(1 - p_{mig,n})$ then $f(k_n)$ is Lipschitz continuous*

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

*with parameter $L_n > 0$ and the slope of $\tilde{T}(\mathbf{k})$ does not change rapidly.*

*Proof.*

$$f(k_n) = \frac{A}{\sqrt{k_n}(\sqrt{k_n}+c)^2}\left(a - b\sum_{m\neq n}\frac{p_m\sqrt{k_m}}{\sqrt{k_m}+c} - \frac{bp_n\sqrt{k_n}}{\sqrt{k_n}+c}\right)^F \tag{18}$$

$$\begin{aligned} &\cdot\left(\sum_{m\neq n}\frac{p_m}{\sqrt{k_m}} + \frac{p_n}{\sqrt{k_n}}\right) \\ &- \frac{B}{k_n\sqrt{k_n}}\left(a - b\sum_{m\neq n}\frac{p_m\sqrt{k_m}}{\sqrt{k_m}+c} - \frac{bp_n\sqrt{k_n}}{\sqrt{k_n}+c}\right)^{1-F} \\ =&\frac{AI}{\sqrt{k_n}(\sqrt{k_n}+c)^{2+F}}\Big((D-E)\sqrt{k_n}+cD\Big)^F \\ &+ \frac{Ap_n}{k_n(\sqrt{k_n}+c)^{2+F}}\Big((D-E)\sqrt{k_n}+cD\Big)^F \\ &- \frac{B}{k_n\sqrt{k_n}(\sqrt{k_n}+c)^{1-F}}\Big((D-E)\sqrt{k_n}+cD\Big)^{1-F} \end{aligned}$$

where $A = hbH/\omega(1-H)$, $B = hp_n/\omega$, $D = a - b\sum_{m\neq n}\frac{p_m\sqrt{k_m}}{\sqrt{k_m}+c}$, $E = bp_n$, $F = 1/(H-1)$, and $I = \sum_{m\neq n}\frac{p_m}{\sqrt{k_m}}$. Due to $F < 0$, there is a need to check if $((D-E)\sqrt{k_n}+cD)$ is zero in the domain. Using (5) and (11), $D$ and $(D-E)$ can be represented as (19) and (20), respectively.

$$D = \frac{r_{total} - r}{H} - \frac{hr_{mig}Sf}{MH}\sum_{\forall m}\frac{p_m\sqrt{k_m}}{\sqrt{k_m}+c} + \frac{hr_{mig}Sf}{MH}\frac{p_n\sqrt{k_n}}{\sqrt{k_n}+c} \tag{19}$$

$$= \frac{1}{H}\left(\tilde{r}_{rem} - r + \frac{hr_{mig}Sf}{M}\frac{p_n\sqrt{k_n}}{\sqrt{k_n}+c}\right)$$

$$D - E = \frac{1}{H}\left(\tilde{r}_{rem} - r + \frac{hr_{mig}Sf}{M}\left(\frac{p_n\sqrt{k_n}}{\sqrt{k_n}+c} - p_n\right)\right) \tag{20}$$

$$= \frac{1}{H}\left(\tilde{r}_{rem} - r - \frac{hr_{mig}Sfp_n}{M}(1-p_{mig,n})\right)$$

Based on the assumption of Lemma 3 and $r \leq \tilde{r}_{rem} - \frac{hr_{mig}S_nf}{M}(1-p_{mig,n})$, the term $((D-E)\sqrt{k_n}+cD)$ is always larger than zero in $[1, M_n]$. Then, every term of $f(k_n)$ is continuous and differentiable in $[1, M_n]$, and $f(k_n)$ is also continuous and differentiable. Based on the mean value theorem,

$$||f(x) - f(y)|| \leq L_n||x-y|| \tag{21}$$

and therefore, $\frac{\partial\tilde{T}(\mathbf{k})}{\partial k_n}$ satisfies Lipschitz continuity and the change range of $\frac{\partial\tilde{T}(\mathbf{k})}{\partial k_n}$ is bounded by $L_n$ when $r \leq \tilde{r}_{rem} - \frac{hr_{mig}S_nf}{M}(1-p_{mig,n})$. ■

In this case, since $\mathbf{k}$ is a subset of positive integers $\mathbb{N}$ based on (8f), equation (15) needs to be projected onto the domain of $\mathbf{k}$ as (22)

$$\mathbf{k}^+ = P\left[\mathbf{k} - \mathbf{w}\circ\frac{\partial\tilde{T}(\mathbf{k})}{\partial\mathbf{k}}\right] \tag{22}$$

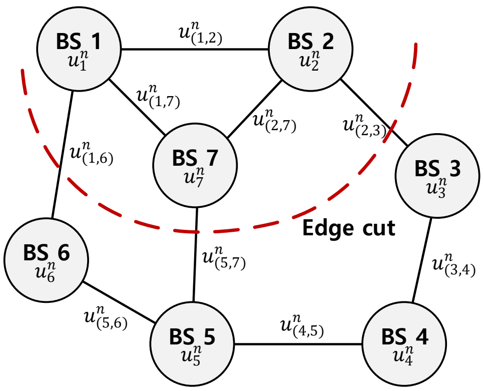


Fig. 3. Clustered network as a graph partitioning problem

where $P[\cdot]$ is the projection operator which projects onto the feasible set, where $\mathbf{w}$ is composed of the fixed step size $w_n \leq 1/L_n$ based on Lemma 3. In addition, $\mathbf{k}$ converges with rate $O(1/t)$, where $t$ is the number of iterations [26].

### 5.3 Physical Resource Allocation using Graph Partitioning Algorithm

In this section, the actual physical resource allocation scheme is proposed to allocate usable physical resources (e.g., CPU cores, link capacity) to users by utilizing the number of clusters obtained above. In (11) and (12), the end-to-end delay has a tendency to increase as the cluster transfer rate increases. In addition, the process of finding a new MEC server according to the movement between clusters can be an obstacle to service provisioning.

The proposed resource allocation scheme minimizes transfers between the clusters using the user transfer rate between BSs and the number of users connecting to the current BS. Clustering is conducted based on a graph partitioning algorithm, which derives a solution to the problem of dividing the network into several clusters in which the number of transfer users running between clusters is as small as possible. As shown in Fig. 3, the network assigned to a user group can be represented in the form of a graph $G_n = (V_n, E_n)$ with $V_n$ nodes and $E_n$ edges. When composing a graph $G_n$ for user group $\mathbb{U}_n$, a higher group user will have a preemption right (over a lower priority group user) to use the server resources. In this paper, it is assumed that the higher the number of users belonging to the group, the higher the priority. The $M_n$ BSs are set as the nodes $V_n$ in the group order that has the largest number of users, and the links between the nodes are set as the edges of the graph partitioning problem. The weight of vertex $u_m^n$ is the number of users assigned to the user group $n$ connected to BS $m$, and the weight of the edge $(m, m')$, $u_{m,m'}^n$ is the number of users in $n$ moving between BS $m$ and $m'$. In this section, a $k_n$-way partition is performed which divides the BS set into $k_n$ clusters, $V_1, V_2, \cdots, V_{k_n}$. A set of edges joining BSs in different clusters is presented as an edge cut, $\delta(V_1, \cdots, V_{k_n})$ [27], [28], [29].

It is assumed that the user's speed and the request for the service maintain the similar tendency for a certain period of time, and it repeats every period [29]. In [29], the period in which the user's tendency changes is periodic and

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

can be computed. The user's tendency period is monitored, and the graph partitioning algorithm is executed when the phase of the period changes. Then, the minimum end-to-end latency based on the number of optimal clusters is investigated. In (12), as migration between clusters is minimized, the actual end-to-end delay decreases. The graph partitioning problem to minimize the number of users migrating between clusters is represented in (24) [27], [28].

$$\text{minimize} \sum_{(m,m')\in\delta(V_1,\cdots,V_{k_n})} u^n_{m,m'} \tag{23}$$

which is subject to

$$V_1 \cup \cdots \cup V_{k_n} = V_n \tag{24a}$$

$$V_m \cap V'_m = \phi \qquad \forall m \neq m' \tag{24b}$$

$$\max_i |V_i| \leq (1+\epsilon)\lceil\frac{|V_n|}{k_n}\rceil \qquad \forall i \in \{1,\cdots,k_n\} \tag{24c}$$

where $|V_i| = \sum_{m\in V_i} u^n_m$ (24c), and $\epsilon$ is the imbalance parameter in which $\epsilon \in \mathbb{R}_{\geq 0}$. In case of $\epsilon = 0$, the sum of the weight of each cluster $|\tilde{V}_i|$ is perfectly the same, which is referred to as perfectly balanced [27]. Therefore, (24c) allows the number of users in each clusters to be within a certain range.

The graph partitioning problem is considered to be a challenging problem among combinatorial optimization problems based on being NP-complete. It is difficult to obtain the optimal solution of the graph partitioning problem. However, several methods and tools have been developed to obtain approximate solutions, such as METIS, where the complexity of METIS is $O(|V_n| + |E_n| + k_n log(k_n))$ [28].

### 5.4 Clustered VNF Resource Allocation Scheme

In this section, the clustered VNF allocation scheme (based on the context-aware grouping algorithm) that assigns VNFs to clusters in the edge network is presented. The pseudo code of the proposed VNF-RACAG scheme is presented in Algorithm 1.

**Algorithm 1** VNF-RACAG Scheme

```
1: monitor network status and the network resources (CPU cores, link) in the MEC network using NFV MANO
2: monitor users' tendency and determine period of the users' tendency change
3: when the phase changes, select number of groups, and assign users to group based on the mobility (pedestrian / vehicle)
4: starting from the high priority group, allocate MEC servers favorable to groups
5: compute set of optimal number of clusters $k_{opt}$ using Algorithm 2
6: execute graph partitioning algorithm using METIS [28] to minimize the number of users which move between clusters
7: if user crosses the border of a cluster, search MEC server with sufficient resources to allocate VNFs
8: migrate VNFs to the selected MEC server
```

The network status and network resources are monitored in the MEC network using NFV MANO (step 1). The users' mobility is monitored through the access and mobility function (AMF), which monitors and reports the mobility events individually for each user's mobility [30]. In this paper, more frequent monitoring is performed for users with higher speeds such that all changes in user clusters (and group) are detected immediately. Then, the phase change of the user's tendency is determined based on the monitoring (step 2). When the phase of tendency changes, VNF-RACAG is executed. The number of user groups is selected, and the users are assigned to the group based on their mobility (step 3). The higher priority group has the preemption right for the MEC server, and groups prefer a MEC server that can support as many users as possible that belong to its group (step 4). This is because a higher priority is assigned to groups with more users. Next, the set of optimal number of clusters $\mathbf{k}_{opt}$ is calculated using Algorithm 2 (step 5), and the graph partitioning algorithm is executed to divide the clusters (step 6). If the user moves out of the cluster and into another cluster, the VIM searches for a server with a sufficient amount of free resources that is close to the user in the new cluster (step 7). Then, the VNFM migrates the user's VNFs to the chosen server (step 8).

**Algorithm 2** Optimum Clusters Computation

```
Require: $\mathbf{k} = [k_1, \cdots, k_N]$, $k_n \in \mathbb{N}$, $1 \leq k_n \leq M$ for $\forall n$, and network parameters
1: initialize the components of k as $k_n = 1$ for $\forall n$
2: confirm the convergence of k based on Lemma 3
3: set w as $1/L_n$
4: while k converges do
5:     compute $\partial\tilde{T}(\mathbf{k})/\partial\mathbf{k}$ using (16) and (17)
6:     update k based on (22)
7: end while
8: set k as the optimal number of clusters
```

Algorithm 2 is executed to calculate the optimal number of sets of **k**, which is described as follows. First, the necessary network parameters are requested using NFV MANO and AMF. The initial components of **k** are initialized to 1 (step 1). Lemma 3 is used to confirm whether **k** converges (step 2). The vectors of weight **w** are set to $1/L_n$ so that the Lipschitz continuity of Lemma 3 is satisfied [25] (step 3) and (22) is repeated until **k** converges to a specific value (step 4-7). The convergent **k** is set as the optimal **k** and returned to Algorithm 1 (step 8).

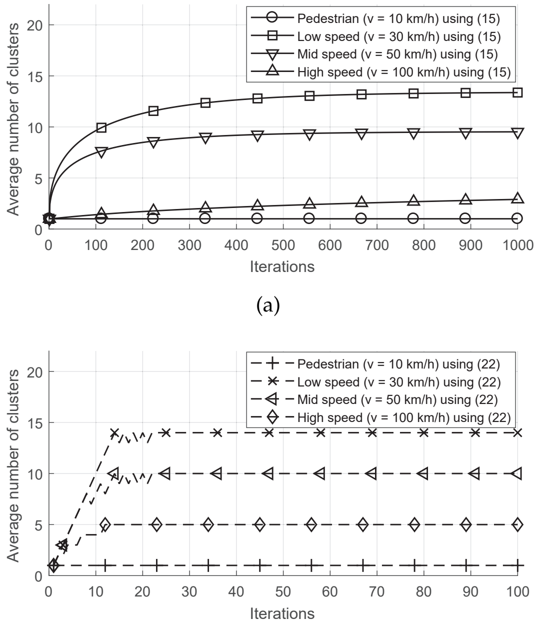
## 6 Performance Analysis

In this section, the performance of the clustered VNF allocation scheme based on context-aware grouping with other techniques for VNF placement in edge networks is compared. Simulation was carried out based on a discrete event simulator implemented in MATLAB.

The velocity distribution of the user is assumed to follow the Erlang distribution [22], and the average speed range is from 1 $\sim$ 80 km/h. In addition, it is assumed that the groups consist of (near) static user groups (pedestrians, 0 $\sim$ 10 km/h), a low-speed user group (10 $\sim$ 30 km/h), a mid-speed user group (30 $\sim$ 50 km/h), and a high-speed

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing



(a)

(b)

Fig. 4. Optimal number of clusters computation based on two gradient descent methods.

user group (above 50 km/h). The performance is examined over a random network formed by 100 MEC nodes that are positioned following a uniform random distribution over a normalized circular area with $n_f$ flows generated, where each flow has a random source and destination node, and the number of required VNFs is uniformly random within the range of 5 to 15. It is assumed that each MEC server has 16 CPU cores, and each VNF randomly requests usage of 1 to 4 CPU cores [19]. The transmission radius is assumed to be 1/3 of the radius of the entire region, and the link capacity is uniformly random within the range from 5 $\sim$ 10 Gbps. The required link capacity for migration is uniformly random within the range from 300 $\sim$ 500 Mbps. The traffic inflow rate is uniformly random within the range of 3 $\sim$ 5 Gbps, and the Hurst parameter $H$ is 0.8.

In Fig. 4, the number of clusters suitable for each group are computed using two gradient descent methods. The average velocity in the RAN area was assumed to be 30 km/h. In Fig. 4, it can be confirmed that the optimum cluster calculation based on the gradient projection method of (22) (Fig. 4.(b)) converges within 100 iterations, while the calculation method based on (15) (Fig. 4.(a)) converges after approximately 1000 iterations. This is because the projection method can take advantage of the constraint that the number of clusters is limited to a positive integer range (i.e., $k_n \in \mathbb{N}$), instead of needing to consider all real numbers.

In Fig. 5, the number of clusters suitable for each group are compared with the average speed of users in the RAN. A group with a lower velocity has a relatively larger number of clusters (i.e., the size of each cluster is smaller), and a group with a higher velocity is composed of a relatively smaller number of clusters. This is because the faster the speed, the more frequent migration occurs, resulting in a smaller number of clusters.

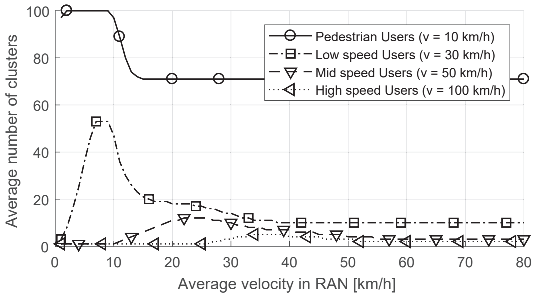


Fig. 5. Optimal number of clusters computation based on the average velocity in the RAN.

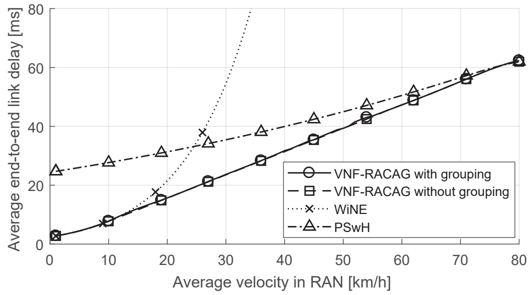


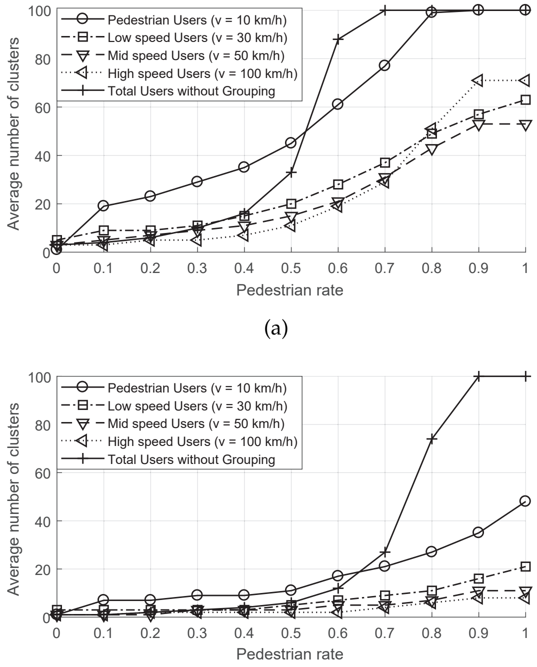
Fig. 6. End-to-end delay performance comparison.

To analyze the performance of VNF-RACAG, the WiNE [10] and PSwH [17] algorithms are compared. The WiNE algorithm is composed of three steps using a heuristic algorithm. In the first step, the list of candidate nodes which have enough capability to allocate VNFs is prepared. In the second step, sorting into decreasing order is conducted. The final step allocates VNFs to the nodes and links to minimize the embedding cost. The PSwH algorithm is an optimal path selection algorithm used in delivery of the offloaded data back to the UEs, while the network service is computed by the initial VNF allocated servers (i.e., without VNF migration). Fig. 6 shows the average end-to-end delay for each scheme. The performance of WiNE is high for near-static users, but the performance deteriorates sharply as the user speed increases due to a degradation in data rate based on an increase in unnecessary VNF migrations. This is because the users' mobility is not considered in WiNE. In contrast to WiNE, the performance of PSwH at high speeds is relatively better. This is because PSwH maintains the initial VNF placement, so there is no data rate consumption due to migrations. PSwH and WiNE do not consider clusters and user context. Fig. 6 shows that the proposed VNF-RACAG results in an average of 94.0% and 22.7% reduced end-to-end delay compared to WiNE and PSwH, respectively.

Fig. 7 and Fig. 8 present the average number of optimal clusters and average end-to-end delay, respectively, according to the ratio of pedestrians to the number of vehicle passengers. In this case, grouping and non-grouping clustering techniques are applied based on the average pedestrian speed of 3 km/h and the average vehicle speed of 50 km/h. In VNF-RACAG, when the ratio of the pedestrians increase (i.e., the number of vehicle passengers decrease), the average

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

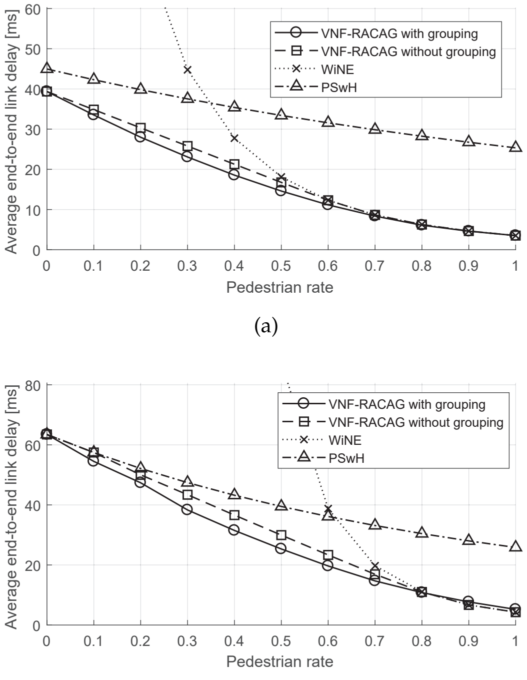
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing



(a)

(b)

Fig. 7. The optimal number of clusters via pedestrian rate. (a) $f = 10$. (b) $f = 15$.



(a)

(b)

Fig. 8. Average end-to-end delay via pedestrian rate. (a) $f = 10$. (b) $f = 15$.

number of clusters will increase and the average delay will decrease as the migration rate decreases. However, the larger the speed difference of each user is, the smaller the delay is when the grouping technique is applied. The results show that VNF-RACAG has a maximum delay that is 12.8% and 15.8% lower than the non-grouping schemes, WiNE and PSwH, respectively. This is because the grouping scheme that derives the optimal number of clusters suitable for each individual group is better in matching the resource needs of each individual user than the non-grouping scheme that assigns the same number of clusters to all users.

## 7 Conclusion

The proposed VNF-RACAG scheme derives the optimal number of clusters that can minimize the end-to-end delay in edge networks and allocates physical resources to users to minimize the number of transfers between clusters. The VNF-RACAG scheme uses a NFV stochastic model as well as geographic contexts and transfer history of the users in the chaining time optimization process. In addition, VNF-RA uses the location and characteristics of individual users to group them into clusters for more efficient functional support. In comparison to the WiNE and PSwH schemes, the simulation results show that a significant gain in end-to-end delay can be obtained by using the proposed VNF-RACAG algorithm.

In the proposed VNF-RACAG scheme, the iterative method of gradient descent is used to calculate the number of clusters to minimize the end-to-end delay. Then, a graph partitioning algorithm is performed to minimize the movement between the clusters, which is executed when the phase of user's tendency changes. Future work will be conducted to extend beyond the parameters of speed, type of service, and predicted moving route used in the VNF-RA scheme by also considering the various wireless resources of the RAN. Based on these changes, in future research, new methods to enhance the VNF-RA process to individually match more end user QoS requirements will be conducted. In addition, improved control techniques to protect NFV services from short term overloads also need to be investigated.

## References

[1] Cisco Visual Networking Index, "Cisco visual networking index: Global mobile data traffic forecast update 2016-2021," Cisco White Paper, Feb. 2017.

[2] C. Cui *et al.*, "Network Function Virtualisation: An Indrodution, Benefits, Enablers, Challenges & Call for Action," ETSI White Paper, Oct. 2012.

[3] J. G. Herrera and J. F. Botero, "Resource Allocation in NFV: A Comprehensive Survey," *IEEE Trans. Netw. Serv. Manag.*, vol. 13, no. 3, pp. 518-532, Sep. 2016.

[4] ETSI GS NFV 002, "Network Function Virtualisation (NFV); Architectural Framework," V.1.2.1, Dec. 2014.

[5] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile Edge Computing: A Key technology towards 5G," ETSI White Paper, Sep. 2015.

[6] M. Patel *et al.*, "Mobile-Edge Computing Introductory Technical White Paper," 2014.

[7] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collarborative Mobile Edge Computing in 5G Netwokrs: New Paradigms, Scenarios, and Challenges," *IEEE Commun. Mag.*, vol. 55, no. 4, pp. 54-61, Apr. 2017.

[8] R. Cziva and D. P. Pezaros, "Container Network Functions: Bringing NFV to the Network Edge," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 24-31, June 2017.

[9] M. Series, "IMT vision: Framework and overall objectives of the future development of IMT for 2020 and beyond," Recommendation ITU-R, Rep. M. 2083-0, 2015.

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2019.2907593, IEEE Transactions on Mobile Computing

[10] R. Riggio, A. Bradai, D. Harutyunyan, T. Rasheed, and T. Ahmed, "Scheduling Wireless Virtual Networks Functions," *IEEE Trans. Netw. Serv. Manag.*, vol. 13, no. 2, pp. 240-252, June 2016.

[11] F. v. Lingen, M. Yannuzzi, A. Jain, R. Irons-Mdean, O. Lluch, D. Carrera, J. L. Perez, A. Gutierrez, D. Montero, J. Marti, R. Maso, and J. P. Rodriguez, "The Unavoidable Convergence of NFV, 5G, and Fog: A Model-Driven Approach to Bridge Cloud and Edge," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 28-35, Aug. 2017.

[12] Y. Nam, S. Song, and J.-M. Chung, "Clustered NFV Service Chaining Optimization in Mobile Edge Clouds," *IEEE Commun. Lett.*, vol. 21, no. 2, pp. 350-353, Feb. 2017.

[13] K. Han, S. Li, S. Tang, H. Huang, S. Zhao, G. Fu, and Z. Zhu, "Application-Driven End-to-End Slicing: When Wireless Network Virtualization Orchestrates With NFV-Based Mobile Edge Computing," *IEEE Access*, vol. 6, pp. 26567-26577, May 2018.

[14] H. Liu, H. Jin, C.-Z. Xu, and X. Liao, "Performance and Energy Modeling for Live Migration of Virtual Machines," *Cluster Comput.*, vol. 16, no. 2, pp. 249-264, June 2013.

[15] S. Mehraghdam, M. Keller, and H. Karl, "Specifying and Placing Chains of Virtual Network Functions," in *Proc. IEEE Int. Conf. Cloud Netw.* (*CloudNet*), Oct. 2014, pp. 7-13.

[16] M. Xia, M. Shirazipour, Y. Zhang, H. Green, and A. Takacs, "Network Function Placement for NFV Chaining in Packet/Optical Datacenters," *J. Lightw. Technol.*, vol. 33, no. 8, pp. 1565-1570, Apr. 15, 2015.

[17] J. Plachy, Z. Becvar, and P. Mach, "Path Selection Enabling User Mobility and Efficient Distribution of Data for Computation at the Edge of Mobile Network," *Comput. Netw.*, vol. 108, pp. 357-370, Oct. 2016.

[18] V. D. Valerio and F. L. Presti, "Optimal Virtual Machines Allocation in Mobile Femto-cloud Computing: an MDP approach," in *Proc. IEEE Wirel. Commun. and Netw. Conf. Workshops* (*WCNCW*), 2014, pp. 7-11.

[19] X. Song, X. Zhang, S. Yu, S. Jiao, and Z. Xu, "Resource-efficient Virtual Network Function Placement in Operator Networks," in *Proc. GLOBECOM*, Dec. 2017, pp. 1-7.

[20] A. Rizk and M. Fidler, "Non-Asymptotic End-to-End Performance Bounds for Networks with Long Range Dependent fBm Cross Traffic," *Comput. Netw.*, vol. 56, no. 1, pp. 127-141, Jan. 2012.

[21] D. Cotroneo, R. Natella, and S. Rosiello, "A Fault Correlation Approach to Detect Performance Anomalies in Virtual Network Function Chains," in *Proc. IEEE Int. Symp. Software Rel. Eng.* (*ISSRE*), Nov. 2017, pp. 90-100.

[22] E. Natalizio, A. Molinaro, and S. Marano, "The Effect of a Realistic Urban Scenario on the Performance of Algorithms for Handover and Call Management in Hierarchical Cellular Systems," in *Proc. Int'l Conf. Telecomm.* (*ICT*), Aug. 2004, pp. 1143-1150.

[23] F. B. Tesema, A. Awada, I. Viering, M. Simsek, and G. Fettweis, "Evaluation of Context-Aware Mobility Robustness Optimization and Multi-Connectivity in Intra-Frequency 5G Ultra Dense Networks," *IEEE Wirel. Commun. Lett.*, vol. 5, no. 6, pp. 608-611, Dec. 2016.

[24] C. Makaya and S. Pierre, "An Analytical Framework for Performance Evaluation of IPv6-Based Mobility Management Protocols," *IEEE Trans. Wirel. Commun.*, vol. 7, no. 3, pp. 972-983, Mar. 2008.

[25] D. Han and J.-M. Chung, "Self-Similar Traffic End-to-End Delay Minimization Multipath Routing Algorithm," *IEEE Commun. Lett.*, vol. 18, no. 12, pp. 2121-2124, Dec. 2014.

[26] A. S. Nemirovski and D. B. Yudin, *Problem Complexity and Method Efficiency in Optimization*. New York: Wiley, 1983.

[27] A. Buluç, H. Meyerhenke, I. Safro, P. Sanders, and C. Schulz, "Recent Advances in Graph Partitioning," *Algorithm Eng.: Sel. Results and Surveys*, vol. 9220, L. Klimann and P. Sanders, Ed. Cham: Springer, 2016, pp. 117-158.

[28] G. Karypis and V. Kumar, "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 359-392, Aug. 1998.

[29] M. Toril, S. Luna-Ramírez, and V. Wille, "Automatic Replanning of Tracking Areas in Cellular Networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 5, pp. 2005-2013, May 2013.

[30] ETSI TS 123 501, "5G; System Architecture for the 5G System" V.15.2.0, June 2018.

**Sooeun Song** received a B.S. degree from the School of Electrical and Electronic Engineering, Yonsei University, Seoul, Republic of Korea. He is currently a Ph.D. candidate in the School of Electrical and Electronic Engineering and a research member of the Communications and Networking Laboratory (CNL) at Yonsei University. His research focuses on NFV/SDN, MEC, 5G NR systems, and URLLC.

**Changsung Lee** received a B.S. degree from the School of Electrical and Electronic Engineering, Yonsei University, Seoul, Republic of Korea, in 2016. He is currently pursuing a combined M.S. and Ph.D. degree in electrical and electronic engineering at Yonsei University, where he is a researcher of CNL. His current research interests include 5G systems, handover, and machine learning.

**Hyoungjun Cho** received B.S., M.S., and Ph.D. degrees from the School of Electrical and Electronic Engineering, Yonsei University. He is currently a software engineer at Samsung Electronics, Republic of Korea. His research focuses on LTE and 5G NR systems.

**Goeun Lim** received B.S. and M.S. degrees from the School of Electrical and Electronic Engineering, Yonsei University. She is currently a software engineer at Samsung Electronics, Republic of Korea. Her research focuses on LTE, LTE-A, 5G NR systems, and CA.

**Jong-Moon Chung** received B.S. and M.S. degrees in electronic engineering from Yonsei University and a Ph.D. in electrical engineering from Pennsylvania State University. Since 2005, he has been a tenured professor in the School of Electrical and Electronic Engineering, Yonsei University. From 1997 to 1999, he was an assistant professor and instructor at the Pennsylvania State University. From 2000 to 2005, he was with the Oklahoma State University as a tenured associate professor. He is an Editor of *IEEE Transactions on Vehicular Technology*, Section Editor of the *Wiley ETRI Journal*, and Co-Editor-in-Chief of the *KSII Transactions on Internet and Information Systems*.

1536-1233 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.