# End-to-End Slicing With Optimized Communication and Computing Resource Allocation in Multi-Tenant 5G Systems

Hsu-Tung Chien, Ying-Dar Lin, *Fellow, IEEE*, Chia-Lin Lai, and Chien-Ting Wang

***Abstract*—Slicing is a key technology in 5G networks to provide scalability and flexibility in allocating computing and communication resources among multiple tenants. Typically, 5G networks have a 2-tier architecture consisting of a central office and transport network in the upper tier and a multi-access edge and radio access network in the lower tier. The tenants which share the 2-tier architecture typically have different service-dependent resource requirements. This study proposes an algorithm, designated as Upper-tier First with Latency-bounded Over-provisioning Prevention (UFLOP), to adjust the capacity and traffic allocation in such a way as to minimize the "over-provisioning ratio" while still satisfying the latency constraints and Service Level Agreements (SLAs) of the tenants. The performance of UFLOP is evaluated experimentally with a real testbed on an end-to-end slicing framework using three typical 5G services, namely Enhanced Mobile Broadband (eMBB), Ultra-Reliable Low Latency (URLLC), and massive Machine Type Connection (mMTC). It is shown that UFLOP successfully determines the critical traffic allocation ratio between the central office and the edge which achieves an over-provisioning ratio close to zero while still meeting the latency requirements. The results suggest optimal resource allocation ratios of 10:0, 1.5:8.5 and 7.8:2.2 for the eMBB, URLLC and mMTC applications, respectively. Furthermore, it is shown that the computing resource behaves as a bottleneck for the eMBB and mMTC services, while the communication resource serves as a bottleneck for the URLLC service.**

***Index Terms*—Radio Access Network (RAN), Multi-access Edge Computing (MEC), slicing, computing resource, communication resource, virtualization, optimization.**

## I. Introduction

FOR infrastructure owners and service providers in multi-tenant 5G networks [1], slicing technology [2] provides an essential tool for meeting the diverse deployment requirements of the different tenants. Among the various service types deployed by 5G tenants, Enhanced Mobile Broadband (eMBB) [3], Ultra-Reliable Low Latency (URLLC) [4], and massive Machine Type Connection (mMTC) [5] are among the most common. eMBB services usually require high bandwidth and a large computational power, and may have either a tight or loose End-to-End (E2E) latency constraint, depending on the particular service involved, e.g., video streaming, Augmented Reality (AR), or Virtual Reality (VR). By contrast, URLLC services, such as Vehicle-to-everything (V2X), demand high reliability connections with E2E latency constraints as tight as 1 ms [4], [6]. Finally, mMTC services (e.g., Internet of Things (IoT) gateways) involve a sudden and massive burst of connections, and typically have a loose latency constraint (e.g., larger than 50 ms to a second). In multi-tenant 5G networks, it is frequently necessary to support these different service types at the same time. As a result, some form of resource isolation mechanism is required to ensure that a sufficient amount of resource is retained for each service. Furthermore, an efficient means of allocating the isolated resources in the 5G network in such a way as to guarantee the Quality of Service (QoS) requirements of the tenants is also required.

The resource isolation problem is generally solved by slicing; a technology which virtualizes the physical resources of the network and then partitions these resources into isolated instances, such as virtual machines and containers, for service deployment. In a previous study [7], the present group proposed a Joint Edge and Central Resource Slicer (JECRS) mechanism for slicing the network and computing resources in 2-tier 5G architectures consisting of a central office and a transport network in the upper tier and a Radio Access Network (RAN) [8] and multi-access mobile edge (MEC) [9], [10] in the lower tier. The experimental results showed that the slicing isolation effect ratio was equal to 1; indicating that the slices were independent of one another.

Since the resources at the edge are typically much rarer than those at the central office, the user traffic should ideally be routed to the central office for processing (providing that the latency constraint allows). In [7], the present group examined the latency requirement stuffocation ratio of three major 5G services types and their representative applications for various fixed configured ratios of the resource allocation distribution between the edge and central office. The results suggest a Central to Edge resource

Manuscript received April 11, 2019; revised September 29, 2019; accepted December 8, 2019. Date of publication December 11, 2019; date of current version February 12, 2020. This work was supported in part by the H2020 Collaborative Europe/Taiwan Research Project 5G-CORAL under Grant 761586 and in part by the Duan Jin Research Project in the Institute of Information and Communications from the Industrial Technology Research Institute, Taiwan. The review of this article was coordinated by Dr. F. Tang. *(Corresponding author: Hsu-Tung Chien.)*

H.-T. Chien and Y.-D. Lin are with the Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: hsutung@cs.nctu.edu.tw; ydlin@cs.nctu.edu.tw).

C.-L. Lai was with the Information and Communications Research Laboratories, Industrial Technology Research Institute, Hsinchu 300, Taiwan. She is now with MediaTek, Inc., Hsinchu 30078, Taiwan (e-mail: chia-linlai@itri.org.tw).

C.-T. Wang is with the Graduate Degree Program of Network and Information Systems, National Chiao Tung University, Hsinchu 300, Taiwan and Academia Sinica, Taipei 115, Taiwan (e-mail: ctwang@cs.nctu.edu.tw).

Digital Object Identifier 10.1109/TVT.2019.2959193

allocation ratio of 9:1, 3:7, and 1:9 for the eMBB, URLLC, mMTC applications respectively.

However, given the use of a fixed configuration ratio, the network resources may be over provisioned for some services. In other words, the measured E2E latency may be lower than that actually required by the service. In practice, 5G networks have only limited resources, and hence the allocation of the resources must be optimized in some way as to prevent this over-provisioning scenario. The authors in [11] presented an algorithm for minimizing the service delay through virtual machine (VM) migration and transmission power control. Meanwhile, the authors in [12] proposed a dynamic latency and reliability-aware policy for task computation, task offloading, and resource allocation between the user equipment (UE) and the MEC. The authors in [13] developed a network resource optimization mechanism for both the transport network and the RAN. However, the proposed mechanism did not consider the E2E delay and latency constraints. Moreover, the study in [12] focused only on the problem of UE side offloading, while that in [13] did not consider the problem of allocating the computing resources. The authors in [14] considered the full resource optimization problem from the RAN to the cloud. However, the key issue discussed in [14] differs from that considered in the present study. Moreover, the mechanisms in [11]–[14] were demonstrated only numerically. In other words, experimental emulations were not performed.

Accordingly, the present study paper proposes an algorithm, designated as Upper-tier First with Latency-bounded Over-provisioning Prevention (UFLOP), which aims to prevent the over-provisioning of the computing and network resources within a two 2-tier 5G network. In designing the algorithm, reference is made to the measured statistics of typical real-world 5G services, including the network propagation delay of the RAN and transport network, the computing power of the pure instances in the edge and central office, the basal delay of different services, and the processing delay tendency of 5G services as the volume of incoming traffic increases. Four different models are formulated, namely 1) ***central office only***: all of the traffic is handled by the central office layer; 2) ***edge only***: all of the traffic is handled by the edge layer; 3) ***central office and edge parallel***: the traffic is separated directly between the edge and the central office in accordance with the resource allocation ratio calculated by UFLOP; and 4) ***central office and edge sequential***: all of the traffic goes to the edge first and some of the traffic is then re-routed to the central office. The feasibility of UFLOP is demonstrated experimentally on the 2-tier slicing platform (JECRS) proposed by the present group in [7].

The remainder of this paper is organized as follows. Section II provides the background and related work. Sections III and IV formulate the system model and describe the detailed operation of the proposed framework. Section V presents and discusses the experimental results for three typical 5G service types (eMBB, URLLC and mMTC). Finally, Section VI provides some brief concluding remarks and indicates the potential direction of future research.

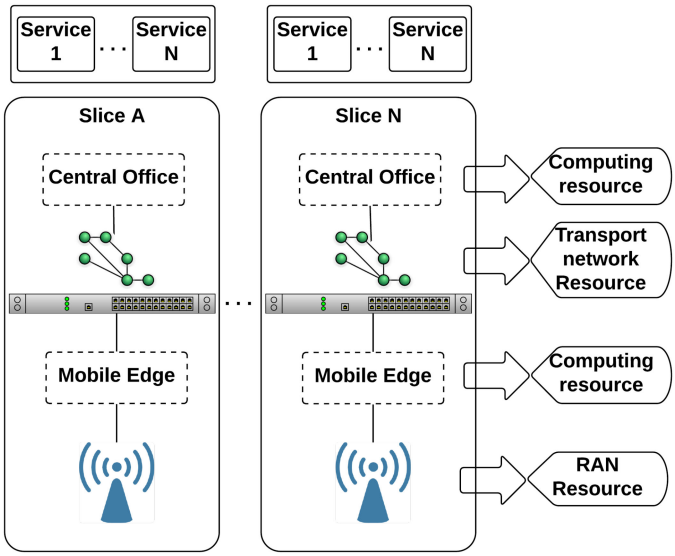



Fig. 1. E2E slicing concept [7].

## II. Background and Related Work

This section presents the background to the main issues involved in the 5G resource isolation and allocation problem considered in the present study, including MEC technology, E2E slicing, Network Functions Virtualization (NFV) [15], Management and Network Orchestration (MANO) [16], 2-tier slicing architecture implementation, and JECRS [7].

### A. MEC Technology and Concept of E2E Slice

Some 5G services (e.g., eMBB and URLLC) require low or ultra-low latency. To meet this requirement, ETSI [9] and 3GPP [10] have prescribed MEC as a key enabling technology for future 5G networks. In the MEC architecture, the network edge serves as a cloud computing platform to run services such that the services are physically closer to the users than in conventional cloud systems. As a result, both the service latency and the computational load on the central office are significantly reduced.

As described in Section I, 5G networks have a 2-tier architecture consisting of a central office, transport network, mobile edge and RAN. As shown in Fig. 1, such networks typically employ an E2E slicing mechanism to set up self-contained slices for each operator containing all the resources they require to run their particular services.

### B. NFV MANO Architecture

One of the most fundamental technologies for implementing E2E slicing is NFV MANO [16]. As shown in Fig. 2, NFV MANO consists of three functional blocks, namely the NFV Orchestrator (NFVO), the Virtual Network Function Manager (VNFM) [17], and the Virtualized Infrastructure Manager (VIM). The NFVO communicates with the VNFM to monitor the status of the VIM, which controls the shared physical infrastructure. The NFVO additionally issues commands for service deployments depending on the availability of the resources

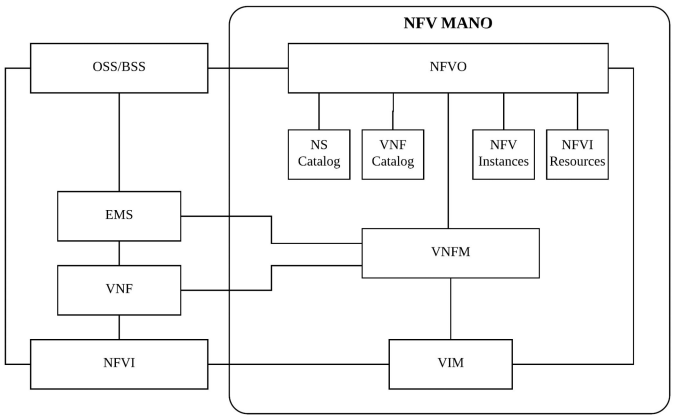



Fig. 2. NFV MANO architecture [15].

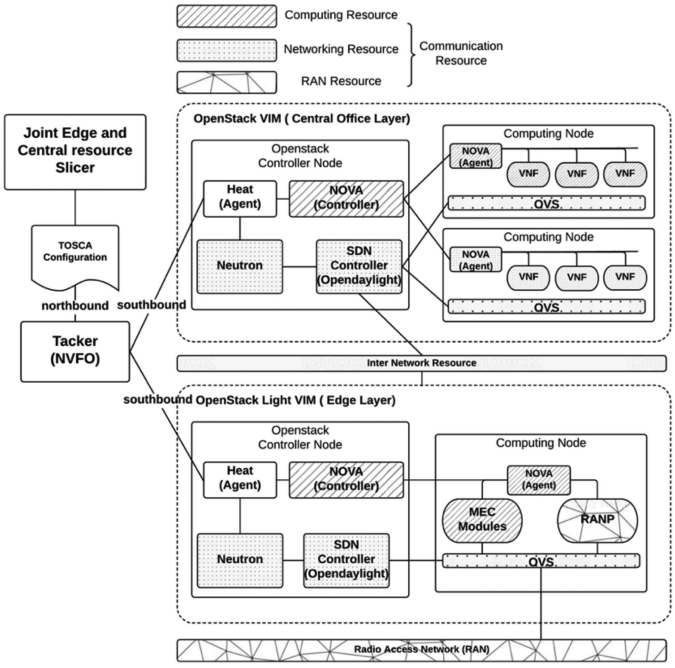



Fig. 3. 2-tier slicing architecture implementation with open source tools [7].

reported by the VNFM. Moreover, the VNFM records the status and maintains the lifecycle of the deployed Virtual Network Functions (VNFs). Finally, the VIM maintains the mapping of the shared logical resources of the NFV Infrastructure (NFVI) to the shared physical infrastructure. Overall, the NFV MANO framework provides network operators with the means to orchestrate the VNFs, arrange the functions in the network nodes into building blocks that can be connected, or chained together, to create services, and maintain the shared physical infrastructure.

### *C. 2-Tier Slicing Architecture Implementation and Joint Edge and Central Resource Slicer (JECRS)*

Fig. 3 shows the 2-tier slicing architecture implemented by the present group in [7]. In constructing the physical testbed for the MANO architecture described in the previous section, the NFVO was implemented using Tacker [18], while the VIMs in the upper and lower tiers of the architecture were implemented using

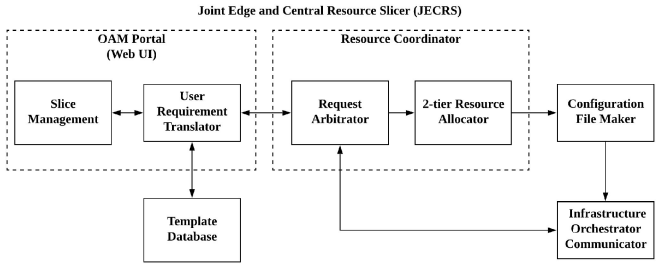



Fig. 4. JECRS framework [7].

OpenStack [19]. The south-bound communications between the NFVO and OpenStack were handled using heat agents. In addition, the computing and network resources were controlled using NOVA [20] and Neutron [21]. E2E slicing was performed using a Software-defined Networking (SDN) controller (OpenDaylight [22]) to manage the network resources with the assistance of Neutron. OpenDaylight was used to manage the inter-transport network between the two tiers of the 5G network and the intra-networks in the central office and mobile edge, respectively, using Open vSwitch (OVS) [23]. OpenDaylight additionally cooperated with a RAN proxy (RANP) [24] to control the RAN. Finally, the north-bound communications between Tacker and the JECRS framework were handled using TOSCA (Topology and Orchestration Specification for Cloud Applications) [25].

As shown in Fig. 4, the JECRS framework comprises five main elements, namely an Operations, Administration and Maintenance (OAM) portal, a resource coordinator, a template database, a configuration file maker, and an infrastructure orchestrator communicator. The OAM portal further comprises a slice management module to enable the tenants to monitor the status of their slices/services and perform create/delete/modify tasks as required, and a user requirement translator module to enable the tenants to specify the particular capability requirements of their services (e.g., latency constraint and number of users). Having specified these requirements, the user requirement translator searches for an appropriate baseline configuration in the template database. The database consists of a set of default mappings (templates), accumulated from translation history, which relate the capability requirements of common 5G services to the capacity of the 2-tier architecture. Having determined an appropriate mapping, the request arbitrator in the resource coordinator module evaluates whether or not the request can be accommodated based on an inspection of the free resource information provided by the infrastructure orchestrator communicator and the Service-level Agreement (SLA) of the tenant. If the request is accepted, the resource allocator in the resource coordinator module determines the resources to be provided by each tier of the 2-tier infrastructure. The resource allocation result is then passed to the configuration file maker, which creates a corresponding Network Service Descriptor (NSD) [26] with a format dynamically determined by the infrastructure orchestrator. The NSD is then transferred via the infrastructure orchestrator communicator to the orchestrator (e.g., Tacker, OpenMANO [27], or Cloudify [28]). On receipt of the NSD, the orchestrator sends a command to the VNFM to

TABLE I
COMPARISON OF PRESENT STUDY WITH RELATED WORKS

| Papers | Algorithm | Objectives | Network/ Server slicing (R/E/T/C) | Number of computing resource tiers | Emulation |
|---|---|---|---|---|---|
| Service Delay Minimizing in ECC [11] | - Integrated Transmission Power and VM Migration Control | - Minimum service delay | No slicing 0/0/0/0 | 1 | No |
| Latency and Reliability-Aware Task Offloading [12] | - dynamic latency and reliability-aware policy | - Maximum computation and transmit power | | 2 (UE and MEC) | |
| SDN Based Service Slicing [13] | - GA-[SS] | - Maximum utilization | Network R/0/T/0 | 0 | |
| An auction-based model [14] | - Two-step Auction Mechanism | - Choose a slice<br>- Maximum revenue | Both R/E/T/C | 2 | |
| Our proposed | - UFLOP | - Create a slice/service<br>- Over-provisioning preventing | | | Yes |

deploy the corresponding slice to the tenant such that they can build their service.

### D. Related Work

The performance of the JECRS framework was compared qualitatively with that of other existing methods in [7]. It was shown that JECRS was the only mechanism to consider both server slicing and network slicing with a full 2-tier (i.e., RAN to central office) sharing of the resources. The results presented in [7] further proved that full slicing of the two-tier architecture is needed to properly isolate the computing and network resources in each tier. Therefore, the present study focuses directly on the problem of designing a joint resource allocation algorithm to satisfy the E2E latency requirements of 5G services. It is noted that several studies relating to latency-aware resource allocation without slicing have been reported in the literature. For example, the authors in [11] attempted to minimize the service delay through VM migration and transmission power control. Similarly, the authors in [12] proposed a dynamic latency and reliability-aware policy for task computation, task offloading, and resource allocation between the user equipment (UE) and the MEC. However, although the method in [11] successfully reduced the service delay, the authors did not consider a 2-tier architecture or the joint slicing of the computing and communication resources. Furthermore, the study in [12] focused specifically on UE side offloading, which is significantly different in scope from the problem considered in the present study. Table I presents a qualitative comparison of the UFLOP mechanism proposed in the current study with other existing methods reported in the literature.

The problem of E2E slicing has attracted great interest in the literature [13], [14]. However, most previous studies focus on the architecture design and message flow management aspects

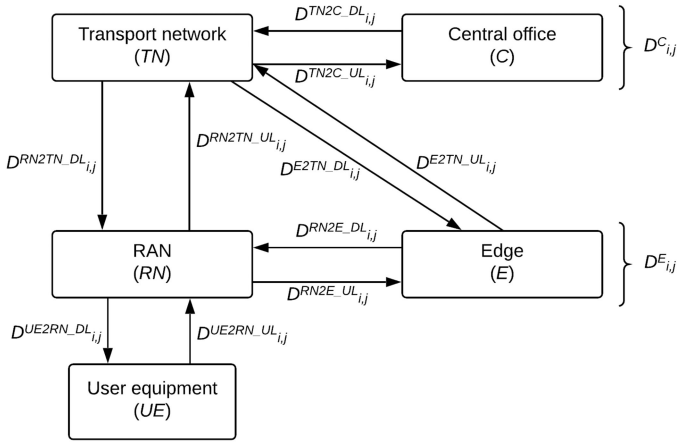



Fig. 5. 2-tier slicing architecture with notations.

of the slicing/service deployment. By contrast, very few studies consider the resource allocation problem. As shown in Table I, the algorithm in [13] provides a SDN-supported solution for optimizing the resources of the transport network and RAN while satisfying the latency constraints of the tenants. However, the computing resources are not considered. Therefore, the latency considered in [13] is not the full E2E latency, and there is thus a risk that the QoS requirements of the tenants may not be met. The scheme in [14] considers both the communication resources and the computing resources. However, the scope of the considered problem is different from that considered in the present study. In particular, the objective of [14] is to choose a slice among the existing slices already deployed by the operator in order to maximize the revenue of the selected slice and meet the QoS requirements of the deployed 5G services. By contrast, the present study optimizes the resource allocation, subject to the E2E latency constraint, in such a way as to prevent resource over-provisioning. Moreover, the study in [14] provides only a high-level treatment of the QoS requirements of different 5G services. Finally, both studies ([13], [14]) evaluate the performance of the proposed mechanisms using simulations only. By contrast, the present study demonstrates the feasibility of the proposed UFLOP mechanism experimentally using a physical testbed.

## III. SYSTEM MODEL AND PROBLEM FORMULATION

This section presents the system model considered in the present study and introduces the problem statement. Table II lists the related notations and their associated meanings.

### A. System Model

Fig. 5 illustrates the arrangement of the 2-tier slicing architecture and user equipment. Note that $D^{C}_{i,j}$ and $D^{E}_{i,j}$ denote the computing delays of the central office $(C)$ and edge $(E)$, respectively, in processing the *j-th* request of the *i-th* tenant. In addition, $D^{TN2C\_DL}_{i,j}$ and $D^{TN2C\_UL}_{i,j}$ are the network delays of the downlink and uplink between $C$ and the transport network $(TN)$. Similarly, $D^{E2TN\_DL}_{i,j}$ and $D^{E2TN\_UL}_{i,j}$ are the network delays of the downlink and uplink between $TN$ and $E$.

TABLE II
USED NOTATIONS

| Notations | Meaning |
|---|---|
| **Entity** | |
| $C$ | Central office |
| $TN$ | Transport network |
| $E$ | Mobile edge |
| $RN$ | RAN |
| $UE$ | User equipment |
| **Tenant** | |
| $T_i$ | *i-th* tenant |
| $Slice_{i,l}$. | Slices of $T_i$ |
| $SLA_i$ | SLA of $T_i$ |
| $SLA_i^C$ | Guaranteed resource at $C$ for $T_i$ |
| $SLA_i^{TN\_UL}$ and $SLA_i^{TN\_DL}$ | Guaranteed resource of uplink and downlink of $TN$ for $T_i$ |
| $SLA_i^E$ | Guaranteed resource at $E$ for $T_i$ |
| $SLA_i^{RN\_UL}$ and $SLA_i^{RN\_DL}$ | Guaranteed resource of uplink and downlink of $RN$ for $T_i$ |
| $F_i^C$ | Available resource at $C$ for $T_i$ |
| $F_i^{TN\_UL}$ and $F_i^{TN\_DL}$ | Available resource of uplink and downlink of $TN$ for $T_i$ |
| $F_i^E$ | Available resource at $E$ for $T_i$ |
| $F_i^{RN\_UL}$ and $F_i^{RN\_DL}$ | Available resource of uplink and downlink of $RN$ for $T_i$ |
| **Service Deployment Request** | |
| $r_{i,j}$ | *j-th* request of $T_i$ |
| $SST_l^{ri,j}$ | Slice/service type of $r_{i,j}$ |
| $VNF^{ri,j}$ | VNF to deploy of $r_{i,j}$ |
| $D^{ri,j}$ | Requirement of E2E delay of $r_{i,j}$ |
| $\lambda^{ri,j}$ | Requirement of traffic of $r_{i,j}$ |
| **Determined Number** | |
| $Req_{i,j}^C$ | Required resource at $C$ of $r_{i,j}$ |
| $Req_{i,j}^{TN\_UL}$ and $Req_{i,j}^{TN\_DL}$ | Required resource of uplink and downlink of $RN$ of $r_{i,j}$ |
| $Req_{i,j}^E$ | Required resource at $E$ of $r_{i,j}$ |
| $Req_{i,j}^{RN\_UL}$ and $Req_{i,j}^{RN\_DL}$ | Required resource of uplink and downlink of $RN$ of $r_{i,j}$ |
| $R^{C\_ri,j}$ | traffic separated ratio of traffic handling by $C$ of $r_{i,j}$ |
| $R^{E\_ri,j}$ | traffic separated ratio of traffic handling by $E$ of $r_{i,j}$ |
| **Delay** | |
| $D_{i,j}^{Cal}$ | Calculated E2E delay |
| $D_{i,j}^C$ | Computing delay of $C$ |
| $D_{i,j}^{TN2C\_UL}$ and $D_{i,j}^{TN2C\_DL}$ | Network delay of uplink and downlink between $TN$ and $C$ |
| $D_{i,j}^{E2TN\_UL}$ and $D_{i,j}^{E2TN\_DL}$ | Network delay of uplink and downlink between $E$ and $TN$ |
| $D_{i,j}^E$ | Computing delay of $E$ |
| $D_{i,j}^{RN2E\_UL}$ and $D_{i,j}^{RN2E\_DL}$ | Network delay of uplink and downlink between $RN$ and $E$ |
| $D_{i,j}^{RN2TN\_UL}$ and $D_{i,j}^{RN2TN\_DL}$ | Network delay of uplink and downlink between $RN$ and $TN$ |
| $D_{i,j}^{UE2RN\_UL}$ and $D_{i,j}^{UE2RN\_DL}$ | Network delay of uplink and downlink between $UE$ and $RN$ |
| **Objective** | |
| $R_{i,j}^P$ | Over-provisioning ratio of $r_{i,j}$ |

*Note:* R, E, T and C denote RAN, Edge, Transport network and Central office, respectively, and 0 indicates that slicing is not applied in the respective layer.

$D_{i,j}^{RN2E\_DL}$ and $D_{i,j}^{RN2E\_UL}$ denote the network delays of the downlink and uplink between $E$ and $RN$. $D_{i,j}^{RN2TN\_DL}$ and $D_{i,j}^{RN2TN\_UL}$ are the network delays of the downlink and uplink between $RN$ and $TN$. Finally, $D_{i,j}^{UE2RN\_DL}$ and $D_{i,j}^{UE2RN\_UL}$ are the network delays of the downlink and uplink between $RN$ and the user equipment ($UE$).

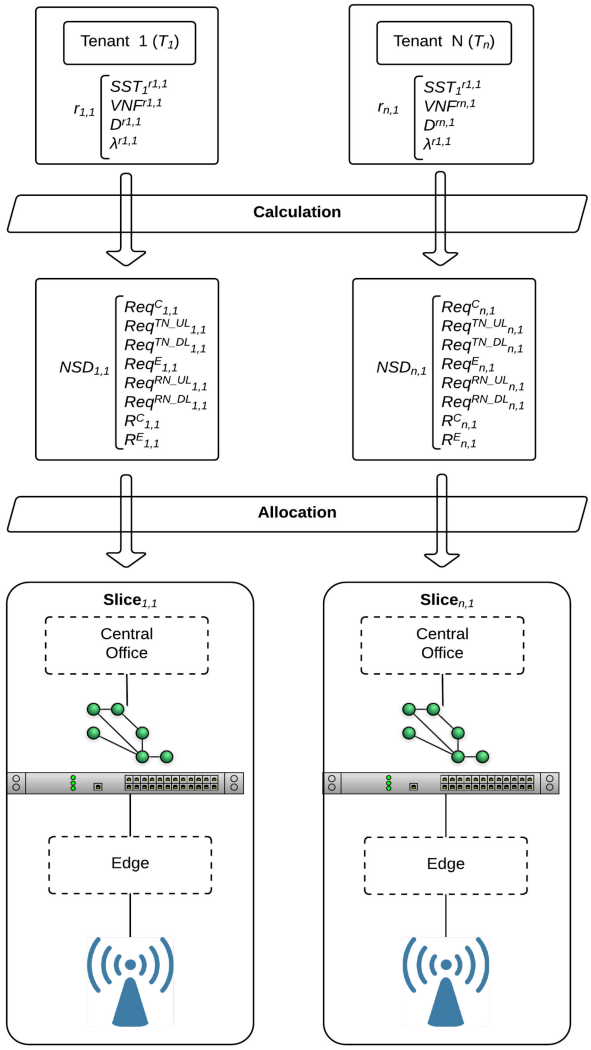



Fig. 6. Overview of slicing problem and associated notations.

Fig. 6 illustrates the slicing problem considered in the present study. Each tenant ($T_i$) has a particular SLA ($SLA_i$) with the operator which owns the infrastructure. In particular, $SLA_i$ describes the agreement of the operator to guarantee tenant $T_i$sufficient resources in each tier to meet its deployment requirements (i.e., $SLA_i^C$, $SLA_i^{TN\_UL}$, $SLA_i^{TN\_DL}$, $SLA_i^E$, $SLA_i^{RN\_UL}$, and $SLA_i^{RN\_DL}$). Let the service deployment requests from the *i-th* tenant be denoted as $r_{i,j}$, where *j* is an index of the individual requests. For request $r_{i,j}$, $SST_l^{ri,j}$ denotes a particular slice/service type, where *l* has a value of 1, 2 or 3 for the eMBB, URLLC and mMTC service types, respectively. In addition, $VNF^{ri,j}$, $D^{ri,j}$ and $\lambda^{ri,j}$ denote the deployed VNF, E2E delay requirement and traffic arrival rate associated with request $r_{i,j}$. In implementing the slicing process, the specific resources required of each tier and the traffic separation ratio between the upper and lower tiers are determined by JECRS in accordance with the $SLA_i$ and $D^{ri,j}$ constraints, and are converted into a corresponding Network Service Descriptor

$NSD_{i,j}$ by the configuration file maker. Referring to Fig. 6, the specific resources required of each tier are denoted as $Req_{i,j}^{C}$, $Req_{i,j}^{TN\_UL}$, $Req_{i,j}^{TN\_DL}$, $Req_{i,j}^{E}$, $Req_{i,j}^{RN\_UL}$, and $Req_{i,j}^{RN\_DL}$, respectively. Moreover, the traffic separation ratio is defined as $R^{C\_ri,j}$: $R^{E\_ri,j}$, where $R^{C_ri,j}$denotes the traffic handled by $C$ and $R^{E\_ri,j}$ is the traffic handled by $E$. If the available free resources of each tier, denoted as $F_i^{C}$, $F_i^{TN\_UL}$, $F_i^{TN\_DL}$, $F_i^{E}$, $F_i^{RN\_UL}$, and $F_i^{RN\_DL}$ respectively, are sufficient to meet the corresponding resource requirements, $VNF^{ri,j}$ is deployed to a slice (denoted as $\mathrm{Slice}_{i,l}$). Otherwise, $r_{i,j}$ is rejected.

### B. Problem Statement

The objective of the UFLOP mechanism proposed in the present study is to allocate the computing and traffic resources of the 2-tier 5G network in such a way as to minimize the over-provisioning ratio of the tenant services while still satisfying the latency constraint and SLA. The problem statement is thus defined as follows.

**Inputs:** (1) The 2-tier slice architecture (consisting of a central office, a transport network, an edge, and a RAN); (2) a user equipment; (3) multiple requests from the tenants for slices of a particular service type $(SST_l^{ri,j})$on which to deploy services $(r_{i,j})$; (4) a VNF to deploy $(VNF^{ri,j})$; (5) the required E2E latency constraint $(D^{ri,j})$; (6) the traffic arrival rate $(\lambda^{ri,j})$ to be processed by each service depending on the expected number of served users in a timeslot; and (7) the SLA with each tenant $(SLA_i)$.

**Output:** (1) The required specific resources of each tier $[\mathrm{Req}_{i,j}^{C}, \mathrm{Req}_{i,j}^{TN\_UL}, \mathrm{Req}_{i,j}^{TN\_DL}, \mathrm{Req}_{i,j}^{E}, \mathrm{Req}_{i,j}^{RN\_UL}, \mathrm{Req}_{i,j}^{RN\_DL}]$; and (2) the traffic separation ratio $[R^{C_ri,j}, R^{E_ri,j}]$between the upper and lower tiers of the 5G network.

**Objective:** Minimize $R_{i,j}^{P} = \frac{D^{ri,j}}{D_{i,j}^{\mathrm{Cal}}} - 1$, where $\frac{D^{ri,j}}{D_{i,j}^{\mathrm{Cal}}} \geq 1$. Note that $R_{i,j}^{P} > 0$ indicates that excessive resources are allocated to the service, while $R_{i,j}^{P} = 0$ indicates that the resources are just sufficient $r_{i,j}$ perfectly.

**Constraints:** The required specific resources of each tier for each tenant must not exceed the corresponding SLA. In other words, $\sum Req_{i,j}^{C} \leq SLA_i^{C}$, $\sum T_{i,j}^{TN\_UL} \leq SLA_i^{TN\_UL}$, $\sum T_{i,j}^{TN\_DL} \leq SLA_i^{TN\_DL}$, $\sum Req_{i,j}^{E} \leq SLA_i^{E}$, $\sum T_{i,j}^{RN\_UL} \leq SLA_i^{RN\_UL}$, and $\sum T_{i,j}^{RN\_DL} \leq SLA_i^{RN\_DL}$. Furthermore, the calculated E2E latency for each tenant must satisfy the respective latency constraint, that is $\frac{D^{ri,j}}{D_{i,j}^{Cal}} \geq 1$.

## IV. Upper-Tier First With Latency-Bounded Over-Provisioning Prevention (UFLOP)

As shown in Fig. 7, UFLOP employs a two-step process to (1) determine the total resource requirements of the tenant request, and (2) allocate the resources in such a way as to meet the objective and constraints defined above. In order to achieve this two-step process, UFLOP must probe the required resource of the services to be deployed based on the served traffic and evaluate the latency such that it can properly estimate the traffic and perform resource allocation. This section introduces

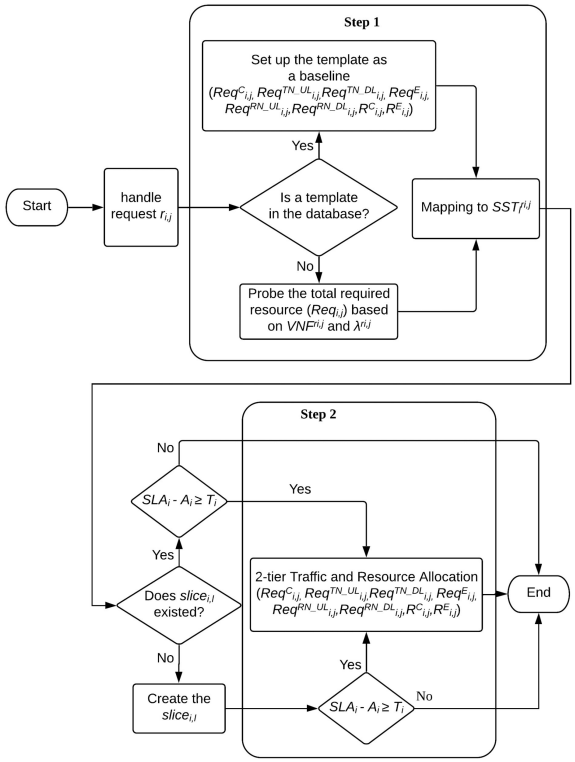



Fig. 7. Overview of UFLOP.

the detailed mechanisms and formulae employed by UFLOP to conduct the resource probing process, evaluate the latency, and allocate the resources. The implementation of UFLOP on the JECRS slicing framework is then briefly described.

### A. Required Resource Probing

The aim of the first step in UFLOP is to establish the required computing and network resources of $VNF^{ri,j}$ which satisfy request $r_{i,j}$ (i.e., request *j* of tenant *i* for slices, as described in Section III-B) with and without $\lambda^{ri,j}$, respectively. The required resources without $\lambda^{ri,j}$ can be regarded as the lower-bound requirement, i.e., the minimum resources which must be deployed on $VNF^{ri,j}$. By contrast, the required resources with $\lambda^{ri,j}$ can be viewed as the upper-bound requirement of $VNF^{ri,j}$. Having identified the lower- and upper-bound resource requirements, the mapping between the required computing and network resources and the tenant request is determined using the following procedure.

To facilitate the probing process for the computing resources, several break points are placed on $VNF^{ri,j}$ when running, and a count is made of the number of instructions processed in each case. The probing mechanism is deployed on a testing instance. As shown in Fig. 7, on receipt of a new tenant request $r_{i,j}$, UFLOP searches for an appropriate baseline configuration in a database consisting of a set of default templates ($Req_{i,j}^{C}$, $Req_{i,j}^{TN\_UL}$, $Req_{i,j}^{TN\_DL}$, $Req_{i,j}^{E}$, $Req_{i,j}^{RN\_UL}$, $Req_{i,j}^{RN\_DL}$, $R^{C\_ri,j}$, and $R^{E\_ri,j}$) accumulated from previous probing rounds. If no suitable template can be found, UFLOP sets up $VNF^{ri,j}$ in the testing instance and conducts a further

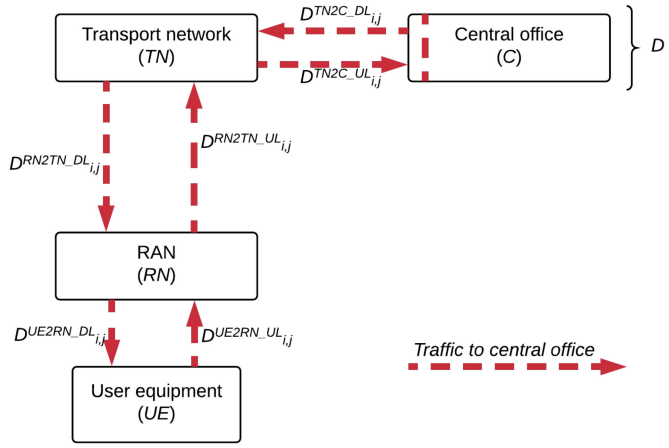



Fig. 8. Central office only model: all traffic is handled by *C*.

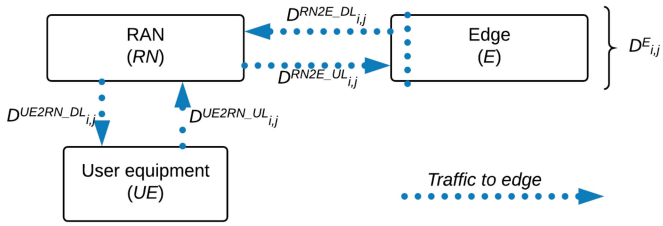



Fig. 9. Edge only model: all traffic is handled by *E*.

probing process. In particular, UFLOP evaluates $VNF^{ri,j}$ for 0% of $\lambda^{ri,j}$, 25% of $\lambda^{ri,j}$, 50% of $\lambda^{ri,j}$, 75% of $\lambda^{ri,j}$, and 100% of $\lambda^{ri,j}$, respectively, in order to accurately estimate the relationship between the number of instructions needed and the actual value of $\lambda^{ri,j}$. Regarding the network resource probing process, UFLOP monitors the usage of the bandwidth at *TN* and *RN* during running and evaluates $VNF^{ri,j}$ using the same break points as those considered when probing the computing resource in order to determine the relationship between the bandwidth required and the actual value of $\lambda^{ri,j}$. The probing results (both computing and network) are then passed to the second step of UFLOP to determine the optimal allocation of the required resources which meets $D^{ri,j}$.

### B. Latency Modeling

In the present study, the delays of the various links and nodes in the 2-tier slicing architecture (see Fig. 5) are modeled using an M/M/1 queuing model, in which each node (or link) is treated as a server and the packet arrivals are assumed to follow a Poisson process with rate $\lambda$. Furthermore, the packet service rate follows an exponential distribution with rate $\mu$. The time spent by each packet in the system is therefore given by $\frac{1}{\mu-\lambda}$. In practice, the traffic in the 2-tier slicing architecture may be served by either the central office or the edge (or a combination of the two), and hence different types of traffic may be served by different servers (nodes or links). Thus, as described in the following, in evaluating the latency, the UFLOP framework formulates four different M/M/1 queuing models.

*1 ) Central Office (C) Only:* An assumption is made that all of the traffic is handled by the central office. In other words, the traffic is handled by a total of seven servers, namely *C*, the up- and down-links between *TN* and *C*, the up- and down-links between *TN* and *RN*, and the up- and down-links between *RN* and *UE* (see Fig. 8). In general, the network delay comprises the propagation delay, the transmission time, and the queueing delay. The one-way propagation delay $(D_{i,j}^{P_C})$is given by

$$D_{i,j}^{P\_C} = \frac{\text{Dist}^{UE2RN} + \text{Dist}^{RN2TN} + \text{Dist}^{TN2C}}{\text{lightspeed}}, \quad (1)$$

where $\text{Dist}^{\text{entity2entity}}$ denotes the distance (in meters) between the two entities and lightspeed is the speed of light ($3 \times 10^8$ m/s). The combined one-way transmission $(D_{i,j}^{T_C})$time and queueing delay $(D_{i,j}^{Q\_C})$ is given as

$$D_{i,j}^{T\_C} + D_{i,j}^{Q\_C} = \frac{1}{\text{Req}_{i,j}^{RN\_UL} - \lambda^{ri,j}} + \frac{1}{\text{Req}_{i,j}^{RN\_DL} - \lambda^{ri,j}} + 2\left(\frac{1}{\text{Req}_{i,j}^{TN\_UL} - \lambda^{ri,j}} + \frac{1}{\text{Req}_{i,j}^{TN\_DL} - \lambda^{ri,j}}\right). \quad (2)$$

The E2E network delay of the "*C* only" model $(D_{i,j}^{\text{Net\_}C})$ is thus equal to

$$\begin{aligned} D_{i,j}^{Net\_C} &= D_{i,j}^{UE2RN\_UL} + D_{i,j}^{RN2TN\_UL} + D_{i,j}^{TN2C\_UL} \\ &\quad + D_{i,j}^{TN2C\_DL} + D_{i,j}^{RN2TN\_DL} + D_{i,j}^{UENRN\_DL} \\ &= 2\left(D_{i,j}^{P\_C} + D_{i,j}^{T\_C} + D_{i,j}^{Q\_C}\right). \end{aligned} \quad (3)$$

In addition, the computing delay of the "*C* only" model $(D_{i,j}^{\text{Com\_}C})$ is equal to the computing delay of $VNF^{ri,j}$ in *C*, i.e.,

$$D_{i,j}^{\text{Com\_}C} = D_{i,j}^{C} = \frac{1}{\text{Req}_{i,j}^{C} - \lambda^{ri,j}}. \quad (4)$$

The overall E2E delay of the "*C* only" model is thus obtained as

$$D_{i,j}^{E2E\_C} = D_{i,j}^{\text{Net\_}C} + D_{i,j}^{\text{Com\_}C}. \quad (5)$$

*2 ) Edge Only:* All of the traffic is handled by the edge. In other words, the traffic is handled by a total of five servers, namely *E*, the up- and down-links between *RN* and *E*, and the up- and down-links between *RN* and *UE* (see Fig. 9). In determining the network delay, the one-way propagation delay $(D_{i,j}^{P_E})$is evaluated as

$$D_{i,j}^{P\_E} = \frac{\text{Dist}^{UE2RN} + \text{Dist}^{RN2E}}{\text{lightspeed}}. \quad (6)$$

Meanwhile, the combined one-way transmission time $(D_{i,j}^{T_E})$and queueing delay $(D_{i,j}^{Q\_E})$ is computed as

$$D_{i,j}^{T\_E} + D_{i,j}^{Q\_E} = \frac{1}{\text{Req}_{i,j}^{RN\_UL} - \lambda^{ri,j}} + \frac{1}{\text{Req}_{i,j}^{RN\_DL} - \lambda^{ri,j}} + \frac{1}{\text{Req}_{i,j}^{TN\_UL} - \lambda^{ri,j}} + \frac{1}{\text{Req}_{i,j}^{TN\_DL} - \lambda^{ri,j}}. \quad (7)$$

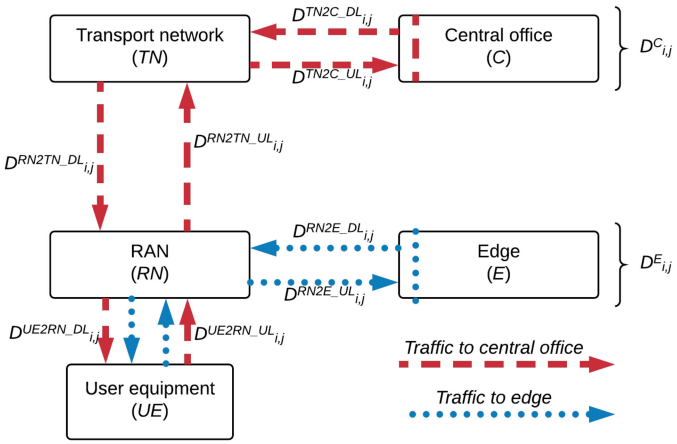



Fig. 10. Central office and edge parallel model: traffic is separated directly between the edge and central office.

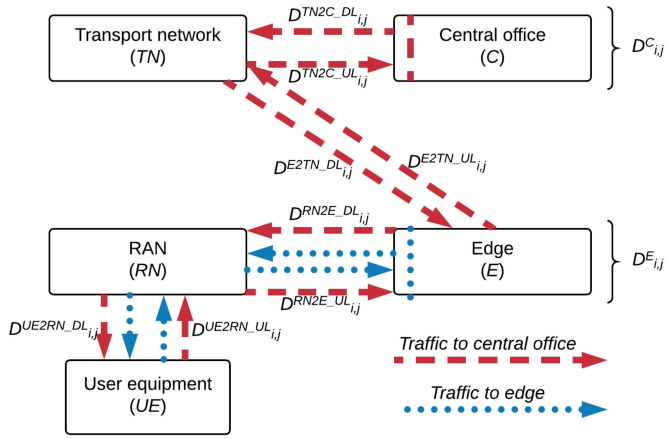



Fig. 11. Central office and edge sequential model: all traffic goes to the edge first and some of the traffic is then routed to the central office.

In other words, the E2E network delay of the "*E* only" model $(D_{i,j}^{\text{Net\_}E})$ is similar to $D_{i,j}^{Net\_C}$ and is given as

$$D_{i,j}^{Net\_E} = D_{i,j}^{UE2RN\_UL} + D_{i,j}^{RN2E\_UL} + D_{i,j}^{RN2R\_DL} + D_{i,j}^{UENRN\_DL} = 2\left(D_{i,\bar{j}}^{P\_E} + D_{i,\bar{j}}^{T\_E} + D_{i,\bar{j}}^{Q\_E}\right). \tag{8}$$

The computing delay of the "*E* only" model $(D_{i,j}^{\text{Com\_}E})$ is equal to that of $VNF^{ri,j}$ in *E*, i.e.,

$$D_{i,j}^{\text{Com\_}E} = D_{i,j}^{E} = \frac{1}{\text{Req}_{i,j}^{E} - \lambda^{ri,j}}. \tag{9}$$

Thus, the E2E delay of the "*E* only" model is obtained as

$$D_{i,j}^{E2E\_E} = D_{i,j}^{\text{Net\_}E} + D_{i,j}^{\text{Com\_}E}. \tag{10}$$

*3 ) Central Office and Edge Parallel:* An assumption is made that the traffic is separated directly between the central office and the edge in accordance with $R^{C\_ri,j}$ and $R^{E\_ri,j}$, respectively. The traffic processed by *C* is handled by the same seven servers as those in the "*C* only" model, while that processed by *E* is handled by the same five servers as in the "*E* only" model (see Fig. 10). In computing the network delay, the one-way propagation delay of the traffic processed by *C* is given by (1), while that of the traffic handled by *E* is given by (6). Meanwhile, the combined one-way transmission and queueing delay of the traffic handled by *C* is given by $\frac{1}{R^{C\_ri,j}} \times (3)$, while that of the traffic handled by *E* is equal to $\frac{1}{R^{E\_ri,j}} \times (9)$. For the computing delay, the traffic handled by $VNF^{ri,j}$ in *C* incurs a delay of $\frac{1}{R^{C\_ri,j}} \times (4)$, while the traffic handled by $VNF^{ri,j}$ in *E* has a delay of $\frac{1}{R^{E\_ri,j}} \times (9)$. Thus, the E2E delay of the "*CE* parallel" model $(D_{i,j}^{E2E\_CEP})$ is obtained as

$$D_{i,j}^{E2E\_CEP} = 2\left[D_{i,\bar{j}}^{P\_C} + \frac{1}{R^{C_{ri},j}}\left(D_{i,\bar{j}}^{T\_C} + D_{i,\bar{j}}^{Q\_C}\right)\right] + \left[\frac{1}{R^{C_{ri},j}} \times (4)\right], \tag{11}$$

for the traffic handled by *C*, and

$$D_{i,j}^{E2E\_CEP} = 2\left[D_{i,\bar{j}}^{P\_E} + \frac{1}{R^{E_{ri},j}}\left(D_{i,\bar{j}}^{T\_E} + D_{i,\bar{j}}^{Q\_E}\right)\right] + \left[\frac{1}{R^{E_{ri},j}} \times (9)\right], \tag{12}$$

for the traffic handled by *E*.

*4) Central Office and Edge Sequential:* All of the traffic generated by the *UE* is passed initially to the edge, and some of this traffic is then further routed to the central office. In other words, the traffic is handled by a total of twelve servers, as shown in Fig. 11. Let the proportion of traffic routed from the edge to the central office be denoted as $R^{S2C}$. In evaluating the network delay, the one-way propagation delay $(D_{i,\bar{j}}^{P_CES})$is given by

$$D_{i,\bar{j}}^{P\_CES} = \frac{\text{Dist}^{UE2RN} + \text{Dist}^{RN2E}}{\text{lightspeed}} + \frac{\text{Dist}^{UE2RN} + \text{Dist}^{RN2E} + \text{Dist}^{E2TN} + \text{Dist}^{TN2C}}{\text{lightspeed}}. \tag{13}$$

In addition, the combined transmission and queuing delay of the uplinks is computed as

$$D_{i,\bar{j}}^{T\_CES\_UL} + D_{i,\bar{j}}^{Q\_CES\_UL} = \frac{1}{\text{Req}_{i,j}^{RN\_UL} - \lambda^{ri,j}} + \frac{1}{\text{Req}_{i,j}^{TN\_UL} - \lambda^{ri,j}} + \frac{1}{R^{S2C}} \times 2 \times \frac{1}{\text{Req}_{i,j}^{TN\_UL} - \lambda^{ri,j}}, \tag{14}$$

Similarly, the transmission and queuing delay of the downlinks is given by

$$D_{i,\bar{j}}^{T\_CES\_DL} + D_{i,\bar{j}}^{Q\_CES\_DL} = \frac{1}{R^{S2C}} 2\left(\frac{1}{\text{Req}_{i,j}^{TN_{UL}} - \lambda^{ri,j}} + \frac{1}{\text{Req}_{i,j}^{RN\_UL} - \lambda^{ri,j}}\right.$$

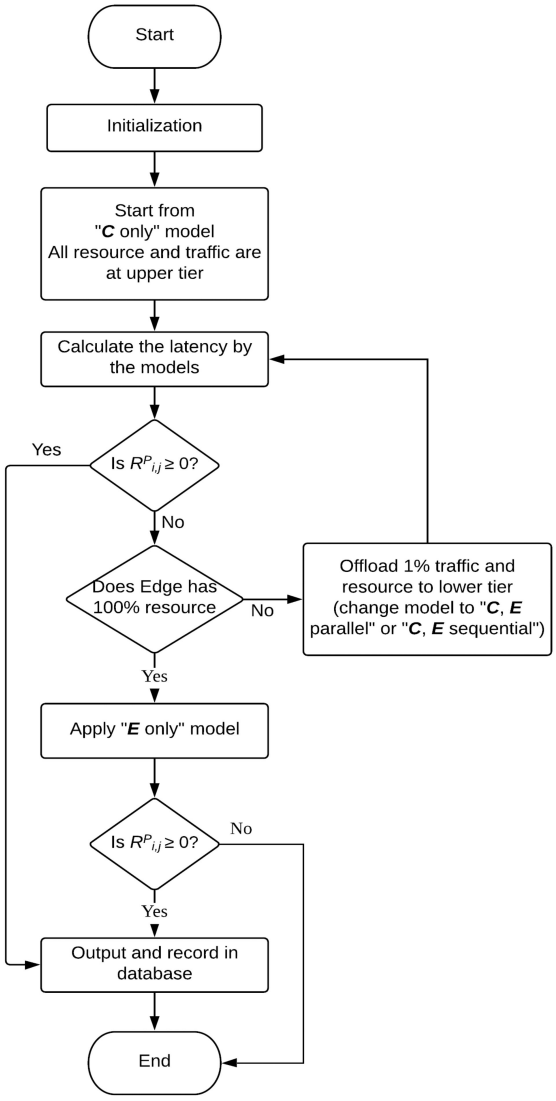



Fig. 12. Second step: 2-tier traffic and resource allocation process.

$$+\frac{1}{\mathrm{Req}_{i,j}^{TN\_UL}-\lambda^{ri,j}}\Bigg) + \frac{1}{(1-R^{S2C})}\left(\frac{1}{\mathrm{Req}_{i,j}^{RN_{UL}}-\lambda^{ri,j}}+\frac{1}{\mathrm{Req}_{i,j}^{TN_{UL}}-\lambda^{ri,j}}\right). \quad (15)$$

Regarding the computing delay, the traffic handled by $VNF^{ri,j}$ in *C* has a delay equal to $\frac{1}{R^{S2C}} \times (4)$, while the traffic handled by $VNF^{ri,j}$ in *E* has a delay equal to $(9)$.

The total E2E delay of the "*CE* sequential" model is thus obtained as

$$D_{i,j}^{E2E\_CES} = 2D_{i,j}^{P\_CES} + D_{i,j}^{T\_CES\_UL} + D_{i,j}^{Q\_CES\_UL} + D_{i,j}^{T\_CES\_DL} + D_{i,j}^{Q\_CES\_DL} + \frac{1}{R^{S2C}} \times (4) + (9). \quad (16)$$

TABLE III
CONSIDERED METRICS

| End-to-End latency (ms) | Data rate (per device) | Number of devices/connections |
|---|---|---|
| eMBB | | |
| 5000-10000 | 2 Mbps | 10 |
| URLLC | | |
| 16-24 | > 25 kbps | 1 |
| mMTC | | |
| 55-75 | 800 kbps | 500 |

### C. Implementation of UFLOP

In the present study, the UFLOP mechanism was implemented as the resource coordinator in the JECRS slicing framework (see Fig. 4) using Python (Ver. 3.6). Referring to Fig. 7, when a tenant request is received, UFLOP (Step 1, implemented by Python with Perf [29], a benchmark testing application) determines the corresponding computing resource requirement in mips and probes the network resource requirement using iftop [30]. UFLOP then checks whether or not the required slice currently exists. If the slice exists, it is selected directly. Otherwise, a new one is created. Finally, UFLOP checks whether the tenant still has sufficient resources available to it under the respective SLA. If enough resources are available, the tenant request is passed to Step 2 to calculate the optimal resource allocation setting. Otherwise, the request is rejected.

Three pseudo tenant services were implemented, namely VLC (v2.2.2) [31] as a streaming service to simulate the behavior of eMBB, Pyvit (v0.2.1) [32] to simulate the behavior of V2X traffic for URLLC, and Mosquitto (v3.1) [33] to simulate the behavior of mMTC based on the MQTT protocol. NextEPC (v0.3.8) [34] was used as the core network for all three services and a MEC traffic shunt [35] was used to identify and forward the traffic to the edge. Regarding the emulation scenarios considered for each service, the eMBB scenario was based on [36], in which the authors studied the metrics of streaming services by examining a dataset of 23 million views compiled from 6.7 million individual users; URLLC referred to the V2X usage cases described in [6] for 5G V2X; and mMTC referred to [37], in which the authors investigated the requirements of various latency-critical IoT cases. Table III summarizes the metrics considered in the various emulation scenarios.

## V. INVESTIGATED ISSUES AND RESULTS

This section commences by describing the physical testbed and emulation scenarios used to investigate the performance of the UFLOP framework. A detailed investigation is then performed into the optimal resource allocation ratio for each of the 5G service types, the bottleneck factors for each service, and the latency sensitivity of each service. Finally, a brief discussion is presented on the applicability of the proposed framework to LTE RAN and 5G New Radio (5G-NR) [38].

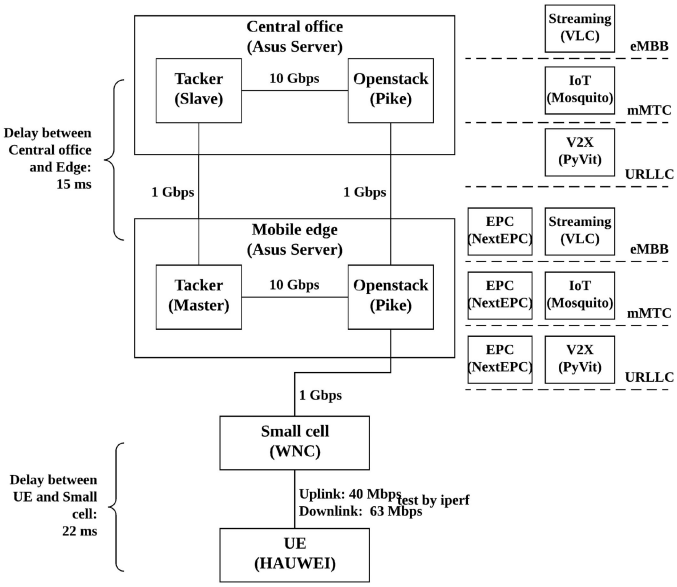



Fig. 13. Emulation platform.

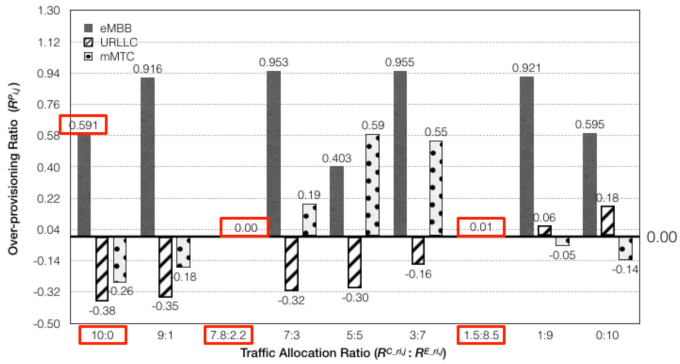



Fig. 14. Over-provisioning ratios of eMBB, URLLC, and mMTC with configured and optimized (red squares) traffic allocation ratios.

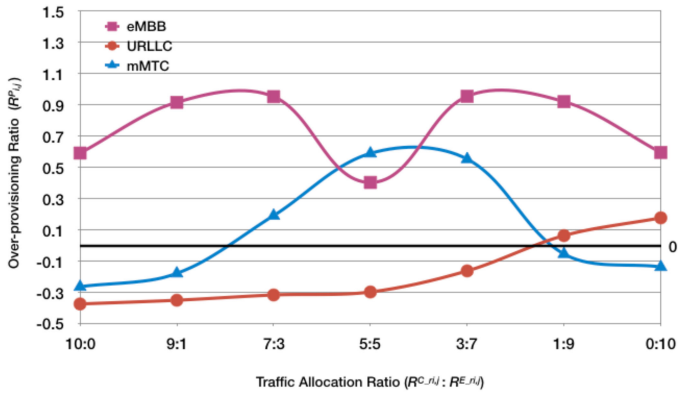



Fig. 15. Over-provisioning ratios of eMBB, URLLC and mMTC given different traffic allocation ratios.

### A. Testbed Description

Fig. 13 shows the emulation platform constructed in the present study. The main elements of the platform include a CAT.4 dongle with a laptop attached to a commercial Frequency-Division Duplexing LTE small cell provided by Wistron NeWeb Corporation (WNC) and supporting Multiple-Input Multiple-Output access. The air interface between the UE (a laptop computer) and the LTE small cell had an uplink rate of 40 Mbps and a downlink rate of 63 Mbps (as determined by iPerf [39]). Two servers equipped with Intel Xeon CPUs and 32G RAM were used as the central office and edge, respectively. The central office ran Openstack Pike and Tacker (v0.10.0) as the master NFVO, while the edge ran Openstack Pike within light-weight containers and Tacker (v0.10.0) as the slave NFVO. The delay between the UE and the small cell was measured to be 22 ms, while that between the central office and the edge was 15 ms. All of the network equipment supported gigabit Ethernet.

### B. Results

In performing the experiments, the latency requirements were set as follows: 10 s for eMBB streaming [36]; 20 ms for URLLC V2X [6]; and 75 ms for mMTC IoT [35]. The experiments emulated ten clients, each of which requested a video (eMBB), created 500 connections (where each connection sent 100 messages (mMTC IoT)) and updated a video (total 10000 frames) to the server (URLLC). In practical networks, the long LTE RAN delay (typically longer than 15 ms before 3GPP Rel. 13, one way) makes it difficult to satisfy the low latency requirement. Consequently, in the emulation experiments, the LTE RAN delay was replaced by the ideal delay of 5G-NR, which is around 1 ms [38].

*1) Over-Provisioning vs. Under-Provisioning:* Fig. 14 shows the experimental results obtained for the over-provisioning ratio $(R^P_{i,j})$of the three services given different traffic allocation ratios between the central office and mobile edge. Note that $R^P_{i,j} > 0$ indicates than excessive resources are allocated to the service, while $R^P_{i,j} < 0$ indicates that the resources are insufficient to meet $r_{i,j}$. As described earlier, the aim of UFLOP is to find the optimal resource allocation ratio for each service. eMBB has a fairly loose latency constraint (10 s). As a result, the latency requirement can be satisfied using all considered values of the traffic allocation ratio. In practice, therefore, a resource allocation ratio of 10:0 represents the optimal solution since it conserves the relatively rarer resources of the edge. Regarding the URLLC and mMTC service types, the ideal case of $R^P_{i,j} = 0$ is obtained for allocation ratios in the range of 3:7 to 1:9 for URLLC and 9:1 to 7:3 for mMTC. In accordance with the calculation process described in Section IV, the UFLOP results suggest an optimal traffic allocation ratio of 1.5:8.5 for the URLLC application (latency 20 ms), and 7.8:2.2 for the mMTC application (latency 75 ms).

*2) Bottleneck: Communication vs. Computing:* The results presented in Fig. 15 show that the over-provisioning ratio for the eMBB service type has an M-shaped tendency as the relative amount of traffic allocated to the central office reduces. This tendency implies that if most of the traffic is handled by the same tier, the processing time (i.e., the latency) increases. A significant reduction in the over-provisioning ratio is observed

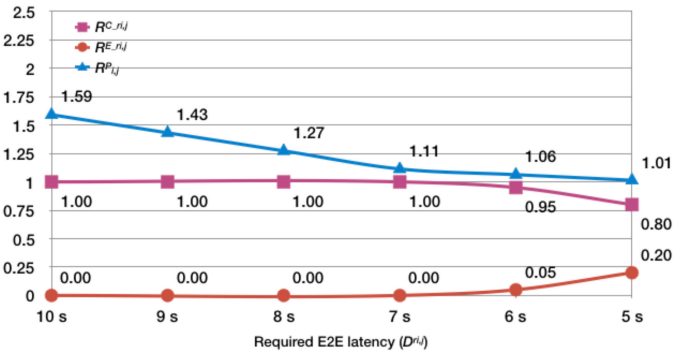



Fig. 16. Traffic allocation ratio for different latency constraints on eMBB service.

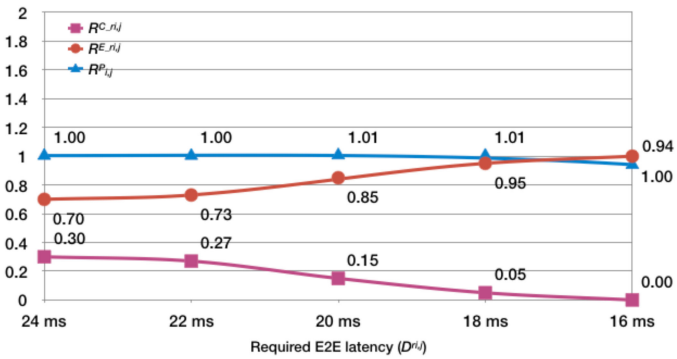



Fig. 17. Traffic allocation ratio for different latency constraints on URLLC service.

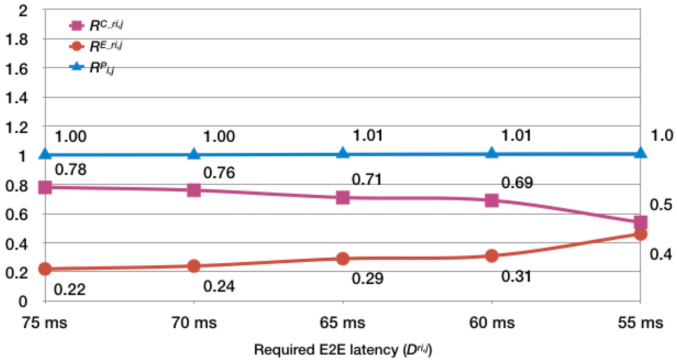



Fig. 18. Traffic allocation ratio for different latency constraints on mMTC service.

for traffic allocation ratios of 10:0, 5:5 and 0:10 as a result of the allocated computing capacity constraint. In particular, since the units of computing capacity in Openstack are limited to the core, JECRS can only allocate computing capacity in the same units. However, for the eMBB emulation scenario considered in the present study, the core capacity is insufficient to handle more than 50% of the eMBB traffic. Consequently, other configuration ratios obtain more computing capacity (total 3 cores) than the three ratios mentioned above (total 2 cores), and hence an over-provisioning of the computing resource occurs for some configuration ratios. For the mMTC service type, the over-provisioning ratio has an inverted U-shaped tendency. As for the eMBB service, this tendency implies that the processing time increases as a greater amount of traffic is handled by the same tier. In other words, for both the eMBB service and the mMTC service, the computing capacity imposes a bottleneck on the performance of UFLOP in minimizing the over-provisioning ratio. However, for the URLLC service, the over-provisioning ratio increases approximately linearly as the relative amount of traffic handled by the edge increases. Hence, it is inferred that the latency of URLLC depends mainly on the network capacity. Overall, therefore, the computing resource serves as the bottleneck for the eMBB and mMTC services, while the communication resource serves as the bottleneck for the URLLC service.

*3) Latency Sensitivity of eMBB, URLLC and mMTC:* Figs. 16–18 show the optimal traffic allocation results obtained from UFLOP for the eMBB, URLLC and mMTC services, respectively, under different latency constraints in order to observe the latency sensitivity of the three services, i.e., the increase in traffic at the edge when changing from a loose latency constraint to a tight latency constraint. As shown in Fig. 16, the eMBB service is assumed to have a fairly loose latency constraint of 5∼10 s. From inspection, the amount of traffic handled by the edge increases by only 5% as the latency is reduced from 10 s to 6 s. However, the traffic increases by 15% as the latency is further reduced from 6 s to 5 s. In other words, the results show that the threshold of the latency constraint for eMBB is 6 s since for lower latencies, a large increase in the amount of resources required at the edge occurs. For the URLLC service (Fig. 17), the results show a threshold value of 22 ms for the latency constraint. In particular, for latency constraints greater than this value, the amount of traffic at the edge increases by only 2% for each 2-ms reduction in the required latency. By contrast, for latency constraints lower than this value, the amount of traffic handled at the edge increases by more than 10% for each reduction of 2 ms in the latency. An inspection of Fig. 18 shows that for the mMTC service, the threshold of the latency constraint is 60 ms. For a given latency constraint threshold (75 ms to 60 ms), the traffic at the edge increases by only 2%–5% for each reduction of 5 ms in the latency. However, the traffic at the edge increases by 15% when the latency constraint is tightened from 60 ms to 55 ms.

Overall, the results presented in Figs. 16∼18 show that, before their respective thresholds, all three services have a low latency sensitivity. In particular, the amount of traffic handled at the edge does not increase by more than 5% as the latency is reduced. In other words, the operators have the opportunity to offload capacity and traffic to the edge in order to improve the E2E latency before reaching the latency constraint threshold.

*4) LTE RAN vs. 5G-NR:* LTE RAN usually has a large round trip time (RTT) of more than 30 ms. For the current testbed, the average RTT was found to be 44 ms. Both values are far higher than the URLLC latency constraint (20 ms). Moreover, the RTT is just one component of the total network delay. If both the

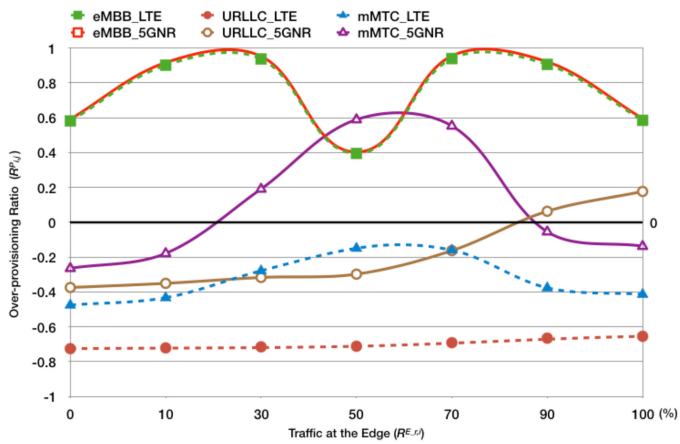



Fig. 19. Over-provisioning ratio under LTE RAN and 5G-NR.

network delay and the computing delay are considered, the total delay would exceed even the relatively relaxed latency constraint of 75 ms for the mMTC service. Fig. 19 shows that under LTE RAN, only eMBB has an over-provisioning ratio greater than 0 (green line with solid squares) because of its fairly loose latency constraint. However, 5G-NR, which has been proposed as the next generation wireless access technology, specifies an ideal RTT of just 1–2 ms [31]. If the ideal RTT of 5G-NR can be realized, the latency constraints of all 5G scenarios can be satisfied by allocating the capacity and traffic appropriately using the UFLOP mechanism.

## VI. Conclusion

This paper has proposed an Upper-tier First with Latency-bounded Over-provisioning Prevention (UFLOP) algorithm for optimizing the capacity and traffic allocation in 2-tier 5G slicing architectures in such a way as to minimize the over-provisioning of the network resources while simultaneously satisfying the latency constraint requirements of the tenants. UFLOP commences by investigating the relationship between the incoming traffic of the tenant services and the required computing resources. It then determines the corresponding lower- and upper-bounds on the computing resources. On receipt of a new tenant request, UFLOP determines whether or not the request can be satisfied using the resources guaranteed to the tenant under the SLA, and then (assuming that sufficient resources are available), iteratively adjusts the capacity and traffic allocation ratio until the optimal allocation ratio is found. In particular, for each considered traffic allocation ratio, the relationship between the incoming traffic of the tenant services and the resource requirements found in the first step are used to calculate the over-provisioning ratio based on M/M/1 queuing theory in order to determine the optimal allocation ratio which satisfies the prescribed service latency.

The feasibility of the UFLOP mechanism has been demonstrated by means of emulation experiments conducted on a physical testbed for three representative 5G services, namely eMBB, URLLC and mMTC. The results have shown that the optimal allocation ratios are equal to 10:0 for the eMBB application, 1.5:8.5 for the URLLC application (latency 20 ms), and 7.8:2.2 for the mMTC application (latency 75 ms). In other words, the results confirm that as the latency constraint becomes tighter, a relatively greater amount of resources are required at the edge. The bottleneck analysis results have shown that for services constrained by the computing resource (i.e., eMBB and mMTC), the traffic should be separated between different tiers of the 5G network architecture. By contrast, for services constrained by the network resource (i.e., URLCC), the traffic should be offloaded to the edge in order to reduce the E2E latency. The latency sensitivity experiments have shown that each service has a particular threshold of the latency constraint, i.e., 6 s for eMBB, 22 ms for URLLC and 60 ms for mMTC. For each threshold value, the latency can be reduced by offloading a small percentage (2∼5%) of the total traffic from the central office to the edge. Finally, a comparison of the over-provisioning ratios of eMBB, URLLC and mMTC has been made between LTE RAN and 5G-NR. The results have confirmed the need for 5G-NR in order to satisfy the low and ultra-low latency constraints of many 5G scenarios.

## References

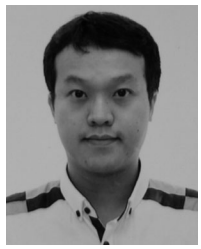
[1] ETSI, "5G." Accessed: Mar. 2019. [Online]. Available: https://www.etsi.org/technologies-clusters/technologies/5g
[2] NGMN Alliance, NGMN Network Slicing, "Description of network slicing concept." Accessed: Mar. 2019. [Online]. Available: https://www.ngmn.org/uploads/media/160113_Network_Slicing_v1_0.pdf
[3] 3GPP, "5G-NR workplan for eMBB." Accessed: Mar. 2019. [Online]. Available: http://www.3gpp.org/news-events/3gpp-news/1836-5g_nr_workplan
[4] "Study on enhancement of ultra-reliable low-latency communication (URLLC) support in the 5G core network (5GC)," 3GPP Specification, 3GPP, Sophia Antipolis, France, 3GPP TR 23.725 V0.3.0, Jul. 2018.
[5] 3GPP, "The path to 5G: As much evolution as revolution." Accessed: Mar. 2019. [Online]. Available: http://www.3gpp.org/news-events/3gpp-news/1774-5g_wiseharbour
[6] M. Boban, A. Kousaridas, K. Manolakis, J. Eichinger, and W. Xu, "Use cases, requirements, and design considerations for 5G V2X," *IEEE Veh. Technol. Mag.*, Dec. 2017.
[7] H. T. Chien, Y. D. Lin, C. L. Lai, and C. T. Wang, "E2E slicing as a service with computing and communication resource allocation for multi-tenant 5G systems," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 104–112, Oct. 2019.
[8] 3GPP, "Radio access network." Accessed: Mar. 2019. [Online]. Available: http://www.3gpp.org/specifications-groups/ran-plenary
[9] "Mobile edge computing; A key technology towards 5G," ETSI, Sophia Antipolis, France, White Paper 11, 2015.
[10] *Technical Specification Group Services and System Aspects; System Architecture for the 5G Systems: Stage 2 (Release 15)*, 3GPP Standard TS 23.501 V0.4.0, 2017.
[11] T. G. Rodrigues, K. Suto, H. Nishiyama, and N. Kato, "Hybrid method for minimizing service delay in edge cloud computing through VM migration and transmission power control," *IEEE Trans. Comput.*, vol. 66, no. 5, pp. 810–819, May 2017.
[12] C. F. Liu, M. Bennis, and H. V. Poor, "Latency and reliability-aware task offloading and resource allocation for mobile edge computing," in *Proc. IEEE Globecom Workshops*, 2017, pp. 1–7.
[13] X. Xu *et al.*, "SDN based next generation mobile network with service slicing and trials," *China Commun.*, vol. 11, no. 2, pp. 65–77, Feb. 2014.
[14] M. Jiang, M. Condoluci, and T. Mahmoodi, "Network slicing in 5G: An auction-based model," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–6.
[15] "Network functions virtualisation (NFV); Virtualisation technologies; Hypervisor domain requirements specification," ESTI Specification, ETSI, Sophia Antipolis, France, ETSI GS NFV-EVE 001 V3.1.1, Jul. 2017.
[16] "Network functions virtualisation (NFV); Management and orchestration," ETSI Specification, ETSI, Sophia Antipolis, France, ETSI GS NFV-MAN 001 V1.1.1, Dec. 2014.

[17] A. Morton, "Considerations for benchmarking virtual network functions and their infrastructure," Internet Engineering Task Force, Fremont, CA, USA, RFC 8172, Jul. 2017.
[18] Openstack, "Tacker." Accessed: Mar. 2019. [Online]. Available: https://wiki.openstack.org/wiki/Tacker
[19] Openstack, "Openstack." Accessed: Mar. 2019. [Online]. Available: https://www.openstack.org/
[20] Openstack, "OpenStack compute (nova)." Accessed: Mar. 2019. [Online]. Available: https://docs.openstack.org/nova/latest/
[21] Openstack, "Neutron." Accessed: Mar. 2019. [Online]. Available: https://wiki.openstack.org/wiki/Neutron
[22] OpenDaylight, "OpenDaylight." Accessed: Mar. 2019. [Online]. Available: https://www.opendaylight.org/
[23] Open vSwitch, "Production quality, multilayer open virtual switch." Accessed: Mar. 2019. [Online]. Available: https://www.openvswitch.org/
[24] Y. D. Lin, H. T. Chien, H. W. Chang, C. Lai, and K. Lin, "Transparent RAN sharing of 5G small and macro cells," *IEEE Wireless Commun. Mag.*, vol. 24, no. 6, pp. 104–111, Dec. 2017.
[25] OASIS, "OASIS topology and orchestration specification for cloud applications (TOSCA) TC." Accessed: Mar. 2019. [Online]. Available: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tosca
[26] Openstack, "Orchestrating VNFs using network services descriptor (NSD)." Accessed: Mar. 2019. [Online]. Available: https://docs.openstack.org/tacker/ocata/devref/nsd_usage_guide.html
[27] OpenMANO. Accessed: Mar. 2019. [Online]. Available: http://www.tid.es/long-term-innovation/network-innovation/telefonica-nfv-reference-lab/openmano
[28] Cloudify, "One platform that simplifies orchestration from core to edge." Accessed: Mar. 2019. [Online]. Available: https://cloudify.co
[29] Perf, "Linux kernel profiling with perf." Accessed: Mar. 2019. [Online]. Available: https://perf.wiki.kernel.org/index.php/Tutorial
[30] iftop, "iftop: Display bandwidth usage on an interface." Accessed: Mar. 2019. [Online]. Available: http://www.ex-parrot.com/~pdw/iftop/
[31] VLC, "A free and open source cross-platform multimedia player and framework that plays most multimedia files and various streaming protocols." Accessed: Mar. 2019. [Online]. Available: https://www.videolan.org
[32] pyvit, "pyvit: Python vehicle interface toolkit." Accessed: Mar. 2019. [Online]. Available: https://github.com/linklayer/pyvit
[33] Eclipse, "Eclipse Mosquitto—An open source MQTT broker." Accessed: Mar. 2019. [Online]. Available: https://mosquitto.org
[34] NextEPC, "Build your own LTE network easy." Accessed: Mar. 2019. [Online]. Available: https://nextepc.org
[35] C.Y. Li *et al.*, "Mobile edge computing platform deployment in 4G {LTE} networks: A middlebox approach," in *Proc. USENIX Workshop Hot Topics Edge Comput.*, Boston, MA, USA, Jul. 2018.
[36] S. S. Krishnan and R. K. Sitaraman, "Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs," *IEEE/ACM Trans. Netw.*, vol. 21, no. 6, pp. 2001–2014, Dec. 2013.
[37] P. Schulz *et al.*, "Latency critical IoT applications in 5G: Perspective on the design of radio interface and network architecture," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 70–78, Feb. 2017.
[38] "NR; User equipment (UE) radio transmission and reception," 3GPP Specification, 3GPP, Sophia Antipolis, France, 3GPP TS 38.101, Jun. 2017.
[39] iPerf, "iPerf—The ultimate speed test tool for TCP, UDP and SCTP." Accessed: Mar. 2019. [Online]. Available: https://iperf.fr

**Hsu-Tung Chien** received the M.S. and Ph.D. degrees in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 2017 and 2019, respectively. He was part of H2020 projects in 5GPPP, 5G-CORAL, Crosshaul, and Transformer, to design and develop an integrated edge and fog system, front-/back-hauls, and a vertical slicer for 5G networks. His research interests include wireless networks, mobile networks, and protocol design.

**Ying-Dar Lin** (F'13) received the Ph.D. degree in computer science from the University of California, Los Angeles, Los Angeles, CA, USA, in 1993. He is currently a Distinguished Professor of computer science with the National Chiao Tung University, Hsinchu, Taiwan. He was a Visiting Scholar with Cisco Systems in San Jose during 2007–2008, the CEO with Telecom Technology Center, Taiwan, during 2010–2011, and a Vice President of the National Applied Research Laboratories, Taiwan, during 2017–2018. Since 2002, he has been the Founder and Director of Network Benchmarking Laboratory, which reviews network products with real traffic and automated tools, and has been an approved test laboratory of the Open Networking Foundation (ONF) since July 2014. He also Co-Founded L7 Networks, Inc., in 2002, later acquired by D-Link Corporation, and O'Prueba, Inc., in 2018. He has authored/coauthored a textbook: *Computer Networks: An Open Source Approach* (McGraw-Hill, 2011), with R.-H. Hwang and F. Baker. His research interests include network security, wireless communications, and network softwarization. His work on multi-hop cellular was the first along this line, and has been cited more than 900 times and standardized into IEEE 802.11s, IEEE 802.15.5, IEEE 802.16j, and 3GPP LTE-Advanced. He is an IEEE Distinguished Lecturer from 2014 to 2017, ONF Research Associate, and recipient of 2017 Research Excellence Award and K. T. Li Breakthrough Award. He has served or is serving on the Editorial Boards of several IEEE journals and magazines, and is the Editor-in-Chief for the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS.

**Chia-Lin Lai** received the M.S. and Ph.D. degrees from the National Cheng-Kung University, Tainan, Taiwan, in 2008 and 2014, respectively. She is currently a Senior Engineer with MediaTek, Inc., Hsinchu, Taiwan. She is a delegate to 3GPP SA Working Group 2 and focuses on related SIDs and WIDs of 5G network topics currently running in 3GPP. She was also part of the H2020 project in 5GPPP; 5G-Crosshaul to develop the transport network architecture and algorithms for 5G networks, and holds several patents in the United States, Mainland China, and Taiwan. Her research interests include optical networks, integrated fiber-wireless networks, next generation network architecture, and protocol design.



**Chien-Ting Wang** received the M.S. degree in communications engineering from the National Chung Cheng University, Chiayi, Taiwan, in 2013. He is currently working toward the Ph.D. degree in computer science with the National Chiao Tung University (NCTU), Hsinchu, Taiwan. He is currently with the Graduate Degree Program of Network and Information Systems, NCTU and Academia Sinica, Taipei, Taiwan. His research interests include software-defined networking, network function virtualization, service chain placement, and resource slicing.