QoS Satisfaction Aware and Network Reconfiguration Enabled Resource Allocation for Virtual Network Embedding in Fiber-Wireless Access Network☆

Pengchao Han[a], Yejun Liu[a,*], Lei Guo[a]

[a]*Smart Systems Lab, School of Computer Science and Engineering, Northeastern University, Shenyang, P. R. China.*

Abstract

Network virtualization emerges as a revolutionary transformation for network operation pattern, which potentially benefits Fiber-Wireless (FiWi) access network by overcoming the bottleneck of joint wireless and optical resource allocation. On one hand, the heterogeneity between optical and wireless subnetworks that has posed severe challenges on the global optimization of FiWi can be tackled using network virtualization by shielding their physical differences. On the other hand, the flexible nature of resource scheduling in FiWi provides an opportunity for Infrastructure Provider (InP) to obtain high profit in the process of Virtual Network Embedding (VNE). In this paper, we highlight the VNE problem in FiWi access network. The wireless channel allocation algorithm and dynamic bandwidth allocation algorithm in FiWi are put forward, based on which the Integer Liner Programming (ILP) model of VNE problem in FiWi access network is formulated mathematically where a practical model of Virtual Network (VN) is focused by endowing each VN with a unique QoS satisfaction requirement. Moreover, aiming at maximizing InP profit, a QoS satisfaction aware VNE algorithm is designed and then improved by network

☆This work was partly supported by Natural Science Foundation of China (NSFC) (61471109, 61775033 and 61501104), and Fundamental Research Funds for the Central Universities (N161608001).

*Corresponding author

Email addresses: hanpengchao199@163.com (Pengchao Han), liuyejun09@gmail.com (Yejun Liu), haveball@163.com (Lei Guo)

reconfiguration mechanisms that enable the reconfiguration of not only virtual networks but also substrate network. Simulation results demonstrate that our proposed algorithms are effective in achieving higher VN acceptance ratio as well as InP profit. Furthermore, the mobility in network virtualization is analyzed in company with suggested solutions for virtualized FiWi access network. Comprehensive designs will be addressed in future work.

Keywords: Fiber-Wireless access network, network virtualization, virtual network embedding, QoS satisfaction, network reconfiguration.

1. Introduction

The ever-increasing data rate in transport network and varieties of emerging applications make access network the bottleneck for high-speed and flexible service providing to end users. Fiber-Wireless (FiWi) access network [1, 2, 3] has been acknowledged as one of the promising solutions against this challenge. However, the heterogeneity between optical subnetwork (Passive Optical Network, PON) and wireless subnetwork (Wireless Mesh Network, WMN) ossifies FiWi in terms of global network management and resource optimization. Network virtualization [4, 5, 6] that decouples traditional Internet Service Provider (ISP) into Service Provide (SP) and Infrastructure Provider (InP) for the purpose of providing customized services to users and managing the resources of Substrate Network (SN), respectively, is a revolutionary transformation for network operation pattern. Particularly, network virtualziation provides FiWi with the potential benefits to overcome the bottleneck of joint wireless and optical resource allocation through shielding their physical differences. The key issue of network virtualization is to embed Virtual Network (VN) requests of SPs onto the common SN managed by InP, that is, Virtual Network Embedding (VNE) as shown in Fig. 1[7, 8, 9]. However, the implementation of network virtualization on FiWi encounters a new challenge in resource allocation pattern of SN. To be specific, in virtualized FiWi access network, the bandwidth of substrate links has to be pre-determined in order to support VNE rather than allocating bandwidth according to dynamic traffic profile; therefore, it is necessary to design novel SN resource allocation schemes to make VNE feasible and achieve the global optimization of network resource.

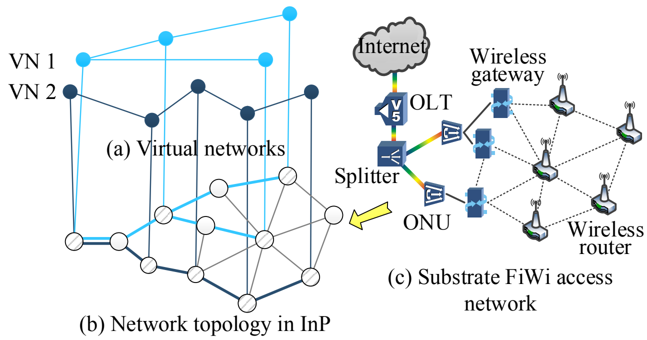

Figure 1: VNE Framework of FiWi access network.

The VNE problem has been proved to be NP-hard [10] due to large scale resource constraints. Thus, varieties of heuristic algorithms have been proposed to map VNs onto SN targeting to gain maximum InP profit. However, existing constraints cannot comprehensively formulate the VN requests in real world which are born with different requirements of Quality of Service (QoS) satisfaction; therefore, it is necessary to equip each VN with a customized QoS satisfaction requirement to reflect its tolerability of QoS decline, such that InP could provide more flexible volume of resource to VNs according to their actual demands instead of satisfying the peak-hour resource demand. As a result, the resource utilization of SN will be increased.

Most of prior VNE approaches were operated in two stages, i.e., virtual node and link embedding stages, which tended to endow virtual nodes with comprehensive weights in the first stage to benefit subsequent link embedding. However, it is difficult to determine a weight that coordinates perfectly multiple factors like resource capacity, delay and load balancing. Thus, the one-stage embedding where virtual nodes and links are embedded simultaneously were proposed. Nevertheless, existing one-stage VNE algorithms were mostly carried out from perspective of graph theory, which ignored the practical characteristics of SN. Specifically, in FiWi access network, the flexibility of channel reallocation in WMN and bandwidth reallocation in both PON and WMN provides potential opportunities for InP to reconfigure its resource to obtain more profit.

Thanks to the flexible feature of resource scheduling in FiWi, additional network reconfiguration methods, i.e., SN bandwidth reallocation and channel reallocation, are promising ways to optimize network resource distribution on the basis of existing VN reconfiguration. Note that the channel reallocation is supposed to frequently change the channels of wireless radio interfaces, while the bandwidth reallocation only changes the scheduling of working time slots among different substrate links. Moreover, SN bandwidth reallocation has no influence on the embedding of existing VNs. Thus, it is preferable to be executed compared with VN reconfiguration and SN channel reallocation. Furthermore, SN channel reconfiguration exerts the highest implementation complexity due to its additional cost (time and operation) brought by configuring hardware, which was overlooked in related work.

In addition, the mobility management plays an important role in wireless networks to ensure the QoS of end users, which is supposed to be handled by the SPs under the network virtualization scenario. However, the colony-level mobility of a group of users results in the mobile virtual nodes, which poses challenges for InP to execute the online embedding. Nevertheless, centralized control manner in network virtualization makes it possible to obtain globally optimal management while taking node mobility into consideration. The opportunities and challenges need to be analyzed.

In this paper, we emphasize on the VNE problem in virtualized FiWi access network where each VN request with customized nodes and links constraints will be mapped onto SN with the objective of maximizing InP profit. Our contributions are as follows:

- We innovatively propose Wireless Channel Allocation algorithm based on Breadth first search (WCAB) and Dynamic Bandwidth Allocation algorithm in FiWi (DBAF) to enable VNE and facilitate SN reconfigurations.

- We design the VN model to make it more practical by considering different QoS satisfaction requirements of VNs, based on which, we formulate the Integer Linear Programming (ILP) model of VNE in FiWi to achieve the optimization of network resource allocation.

- We put forward a novel heuristic algorithm that supports dynamic VN requests to realize the QoS satisfaction aware VNE with reasonable computational complexity on the basis of the basic QoS aware VNE algorithm in our prior work [11]. The channel allocation is adopted in this paper for the VNE optimization compared with [11]. Thus, totally three network reconfiguration operations, i.e., SN bandwidth reallocation, VN reconfiguration and SN channel reallocation are implemented selectively to achieve a higher VN acceptance ratio.

- We introduce a channel reallocation threshold to mitigate the implementation complexity of channel reallocation in SN and make a proper compromise between effortless network reconfiguration and high InP profit.

- We discuss the mobility of both end users and virtual nodes, propose the management framework based on zooming and virtual cells for virtual nodes with low and high mobilities respectively in virtualized FiWi access network, and analyze the opportunities and challenges as a primer for future investigations.

The reminder of this paper is organized as follows. After reviewing the related work in Section II, the initial resource allocation mechanisms including WCAB and DBAF are proposed in Section III for the channel allocation in WMN and the bandwidth allocation in FiWi, respectively. As the resource allocation framework shown in Fig. 2, the problem of VNE considering different QoS satisfaction requirements of VNs is formulated as an ILP model, that is, ILP-Q in Section IV. Moreover, the QoS satisfaction aware VNE algorithm (VNE-Q) together with the network reconfiguration mechanisms, i.e., VNE-R, is put forward in Section V where WCAB and DBAF will be utilized to optimize physical resource distribution so as to obtain more InP profit. In addition, mobility managements for mobile users and virtual nodes are also discussed in Section V. Furthermore, simulation is implemented in Section VI to analyze the performance of the VNE-R. Finally, we conclude this paper in Section VII.

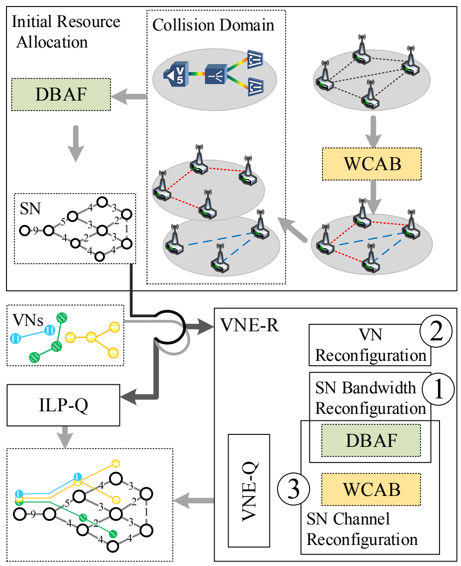

Figure 2: Resource allocation framework.

2. Related Work

2.1. FiWi Access Network and its Virtualization

The integration of optical and wireless access networks has shown a paramount importance on improving the performance of networks. On one hand, taking the optical network as the backhaul of front-end Radio Access Network (RAN) or as the wireless fronthaul directly can realize the performance guarantee of emerging applications. For example, in [12], a small-cell-compatible Wavelength-Division Multiplexing-Frequency-Division Multiplexing (WDM-FDM)-based bidirectional mobile fronthaul network was designed to handle the increased number of connected cells with higher flexibility and lower latency in 5G system. Authors in [13] utilized Radio-over-Fiber (RoF) links for the interconnection of Wireless

Sensor Network (WSN). Two Medium Access Control (MAC) protocols were designed for the WSN-RoF architecture to improve the round-trip propagation delay of optical links and deal with the existence of collision domains. Authors in [14] exploited RoF technology in 5G back and fronthaul applications to achieve the high capacity network transportation, enabling flexible management for services with high fidelity. On the other hand, the flexibly deployed wireless devices enable fibers more elastic resource managements. In [15], authors developed optimal network planning strategies to achieve survivable hybrid FiWi networks. The backup fiber deployment cost was saved due to the use of wireless routing thanks to the ubiquitous wireless coverage of end users.

Furthermore, FiWi access network witnesses a wide implementation in 5G system, Mobile Edge Computing (MEC) and Internet of Things (IoT) etc.. For instance, the combination of PON and advanced wireless access networks was applied in [16] for the efficient local data exchange. Each wireless end device was associated with one (symmetric association) or two Optical Network Units (ONUs, asymmetric association) for the data uploading and downloading. The Mixed-Integer Linear Programming (MILP) models were formulated to minimize weighted number of packet transmitted, such that packets were inclined to be transmitted through one ONU other than by OLT (two ONUs), which led to extra delay. In order to achieve the fiber speed in wireless network, authors in [17] proposed an ultra-small floor-integrated cells system to establish short-reach communication. By applying a scalable ultra-high density wireless connectivity, the effect of interference was reduced and the full frequency reuse in wireless domain was achieved. It was evaluated that a 100 Gbps wireless throughput could be reached. Authors in [18, 19] explored the possibilities of empowering integrated FiWi access networks to offer MEC capabilities, which drives the need for highly localized services within RANs in close proximity to mobile subscribers. Enhanced resource management policies were proposed to improve the offload delay, response time efficiency, and battery life of edge devices. The underlying FiWi dynamic bandwidth allocation was incorporated in offloading activities to achieve the joint optimization. In [20], authors leveraged converged FiWi access networks to design a shared communication infrastructure supporting both IoT applications and traditional services. Given the paramount importance of energy efficiency in both IoT and access networks, the possibilities and potential challenges of designing and implementing power-saving mechanisms were discussed to prolong battery life of IoT devices.

Based on above promising applications of FiWi, the joint optical and wireless resource allocation has been investigated. To address the dynamic nature of the network traffic, authors in [21] presented a bandwidth allocation policy which applied an efficient weighted-based, QoS-aware scheduling in FiWi networks with fairness support. Optimization of multiple dimensional resources including radio, optical and Baseband Unit (BBU) processing was considered in [22, 23, 24] for service provisioning in Cloud Radio over Fiber Network (C-RoFN) to enhance the responsiveness to dynamic end-to-end user demands and globally optimize multi-stratum resources effectively. However, the resource allocation policy taking both radio frequency and link bandwidth into consideration has not been touched in network virtualization.

The advantages of flexibility, heterogeneity, security and scalability endow network virtualization with the widespread application value and prospect. Particularly, virtualized FiWi access network enables a more flexible resource allocation, a shorter traffic delay and a more convenient network management. A general virtualized FiWi access network framework with three layers, i.e., physical layer, virtual layer and VN layer was proposed in [25, 26]. It has been proved that the Round-Trip Time (RTT) in FiWi was reduced exploiting network virtualization and the bandwidth utilization was improved owing to the centralized resource management. Authors in [27] stressed the load balancing problem in virtualized FiWi access network. Two modified weight round robin algorithms were proposed to cope with the delay-ensured voice traffic and the bandwidth-sensitive video traffic respectively to achieve lower traffic delay and higher network resource utilization. Generally speaking, most of these works focused on improving network performance in terms of resource utilization and traffic delay while ignoring the key issue in virtualized FiWi access network, i.e., the VNE problem.

2.2. Two-stage VNE Algorithms

The comparison of VNE mechanisms in the literature is shown in Table 1. The VNE problem that can be formulated as ILP models [28, 29, 30] has been proved to be NP-hard. And VNE algorithms were carried out from two perspectives, i.e., two-stage and one-stage VNEs. In the two-stage VNE, all the nodes in a VN are embedded preferentially, followed by the virtual links embedding utilizing Dijkstra, k-shortest, or multi-commodity flow approaches. In terms of node embedding, greedy algorithms were mostly proposed with specific node weights that tried to take the possibility of successful link embedding into account. The simplest and most straightforward metric for node ranking is the residual CPU capacity [31, 32, 33]. Apart from that, the neighbor resource availability which was formulated as a combination of node CPU capacity and sum of bandwidth capacity on its neighbor links was linked to virtual nodes in [34, 35]. And similarly, the node potential which was introduced from field theory and the General Resource Capacities (GRC) were regarded as node weights in [36, 37] respectively. Moreover, the availability of backup resource, the mean shortest path length among the nodes and the network connectivity were considered for the node ranking to minimize the chance of not finding a solution [38]. Redundant backup resource was allocated to confirm the Service Level Agreements (SLAs) during physical failures in the infrastructure. In [39], the node ranking was operated based on the CPU capacity and the accumulative distance from candidate nodes to all the substrate nodes selected. Moreover, motivated by the fact that most VNs operate on long-term basis and have the characteristics of periodic resource demands, the VNs were embedded according to their periodic resource demands and the prediction method was proposed to allow resource sharing among VNs. Authors in [40] introduced a new link interference metric for each link to quantify the interference caused by its bandwidth scarcity to accept VN requests. And then an Interference-based VNE (I-VNE) algorithm was proposed. The link interference was considered for the node ranking, which was proved to be uniqueness. By jointly considering the temporal and spatial topology information of networks, I-VNE tried to embed each VN request with low interference to avoid rejecting future requests. In [41], five important network topology attributes and global network resources were considered altogether for the node ranking. That is, node degree, node strength (i.e., sum of all adjacent link bandwidth), distance between two nodes, node farness (i.e., the sum of its shortest distances to all the other nodes) and closeness (i.e., the reciprocal of farness of a node) and link interference. The node storage demand was taken into account in [42], and a three-dimension (i.e., computing, network and storage) VNE model was proposed. Two node ranking measurements were considered, one of which was similar to the literature by multiplying the CPU and storage capacities of the node and bandwidth capacities of its adjacent links. The other measurement ranked nodes according to the resource consumption ratio between the CPU and the storage and the substrate node of which resource consumption ratio was closest to that of virtual node was chosen to balance the CPU and storage loads of substrate nodes. Based on the cluster information of networks, the node degree and the clustering coefficient information were combined for node ranking in [43]. The breadth-first-search algorithm was exploited to embed virtual nodes to reduce the resource utilization of substrate links such that both the acceptance ratio of VN requests and the revenues of operational providers were increased. In [44], the virtual node mapping problem was formulated using the Markov decision process (MDP) framework and the Monte Carlo tree search algorithm was utilized for the node embedding. Inspired by Google's PageRank algorithm, authors in [45, 46] calculated the node weights according to the contributions of nodes and links in the VN in an iterative way. And the Markov chain with rewards ranking was proposed in [47] where a node ranking metric which well captured the amount of local resources available in a vicinity of a given node was put forward. That is, ranking of nodes were calculated by aggregating resources in their respective neighborhood in terms of accumulated reward of a suitable random walk with the node as initial state.

2.3. One-stage VNE Algorithms

Although two-stage VNE algorithms are easy to implement, they suffer from limited VN acceptance ratio due to the separated node and link embeddings, which motivates the research of one-stage VNE where virtual nodes and links are embedded jointly. Specifically, an ant colony algorithm based VNE mechanism was proposed to achieve a near-optimal solution with shorter computational time than solving ILP in [45, 48]. Authors in [49, 50] divided a VN into several sub-VNs with star-like topology which was simple and easy to be embedded. The key issue was transformed into choosing appropriate hub node for each sub-VN. And the min-cost max-flow was found in the substrate network to achieve the integrated node and link embeddings in [50]. A VNE algorithm based on subgraph isomorphism detection was proposed in [51], where an isomorphism of each VN was constructed in SN. In [52], the path ranking was applied considering the optical spectrum continuity and contiguity requirements for the embedding of optical data centers based on Optical-Orthogonal Frequency Division Multiplexing (O-OFDM). Shorter and more available substrate paths were preferred over longer and crowded paths. A randomized match of virtual links on a set of candidate substrate paths was performed and if the matches were possible, virtual nodes were mapped accordingly. In [53, 54], a meta node was added in SN for each virtual node and an enhanced network was built by adding temporary links between each meta node and all substrate nodes that satisfy the location constraint. The embedding of two nodes and the link between them would be accomplished by finding a path between their corresponding meta nodes. Considering the wavelength continuity constraint and high degree of the optical switchs in optical data centers, a Markov Chain-based algorithm for VNE in optical data centers was proposed in [55]. The computing capacity, available bandwidth on the established lightpaths and the possible bandwidth that can be provided in future were taken into account for Markov chain probabilistic model to compute the rankings of nodes. And the link embedding was determined while node embedding was completed. Moreover, authors in [56] proposed a robust VNE algorithm based on component connectivity in large-scale network to handle link failures. Two kinds of components were proposed. The giant component was the largest connected component of the network and the small components were the connected components after removing the giant component. It is reasonable that a VN should be embedded onto a component to prevent that the virtual nodes embedded in two components cannot find a path due to the disconnectivity. A k-core method was applied to identify different VN topologies so that a VN request could be embedded onto its corresponding component. In addition, a game theory-based algorithm was proposed in [8] where distributed physical networks were highlighted. However, these one-stage VNE algorithms ignored the potential flexibility of FiWi access network in resource allocation and network reconfiguration, which is expected to raise InP profit.

Table 1: Comparison of VNE mechanisms

Embedding method	Reference	Node Rank	Proposed solution	Other consideration
ILP	[28, 29, 30]	N/A	ILP solving	Memory space guarantee for OpenFlow rules [28]
Two-stage algorithm	[31]	CPU capacity	Greedy node mapping and game theory-based link mapping	Physical equipment outages
	[32]	CPU capacity	Greedy node mapping and shortest path link mapping	Failure recovery enabled VN
	[33]	CPU capacity	Greedy node mapping, shortest path for non-spittable link mapping and multi-commodity for splittable link mapping	Multimedia services in 5G networks
	[34, 35]	Neighbor resource availability	Greedy node mapping and shortest path link mapping	path splitting and migration [34]
	[36]	Node potential	Greedy node mapping and shortest path link mapping	N/A
	[37]	General Resource Capacity	Greedy node mapping and shortest path link mapping	N/A

Continued on next page

Embedding method	Reference	Node Rank	Proposed solution	Other consideration
	[38]	Combination of availability of backup resource, the mean shortest path length among nodes and connectivity	Greedy node mapping and shortest path link mapping	Redundant backup resources allocation
	[39]	Combination of CPU capacity and the accumulative distance from candidate node to all the substrate nodes selected	Greedy node mapping and shortest path link mapping	Online VNE
	[40]	Global information of both link interference and available bandwidth resource	Greedy node and link mappings	Link interference
	[41]	Combination of node degree, node strength, distance between two nodes, node farness and closeness and link interference	Greedy node mapping and shortest path link mapping	N/A
	[42]	Resource consumption ratio between the CPU capacity and the storage capacity	Greedy node mapping and shortest path link mapping	Node storage resource demand/capacity
	[43]	Combination of the degrees of the node and its neighbor nodes in the cluster	Breadth-first-search algorithm for node mapping and shortest path link mapping	Node clustreing
	[44]	Markov Decision Process (MDP)	Monte Carlo tree search algorithm for node mapping and multi-commodity flow and shortest path algorithms for link mapping	New performance metric that captures both acceptance and revenue to cost ratios
	[45, 46]	Page Rank considering node degree, node strength and distance between nodes	Greedy node mapping and shortest path link mapping	N/A
	[47]	Markov chain considering CPU capacity, bandwidth capacity and neighbor metric	Greedy node mapping and shortest path link mapping	N/A
One-stage algorithm	[45, 48]	N/A	Ant-colony search-based algorithm	Online VNE
	[49, 50]	N/A	Heuristics [49] or min-cost max-flow algorithm [50] for the embeddings of star-like sub-VNs	N/A

Continued on next page

Embedding method	Reference	Node Rank	Proposed solution	Other consideration
	[51]	N/A	Subgraph isomorphism detection	N/A
	[52]	N/A	Algorithm based on path ranking and the simultaneous node mapping	Optical spectrum continuity and contiguity requirement for optical data centers
	[53, 54]	N/A	Algorithm based on auxiliary graph	Flexible modulation modes [54]
	[55]	N/A	Markov Chain-based algorithm	The wavelength continuity constraint in optical data centers
	[56]	N/A	Component connectivity-based algorithm	N/A
	[8]	N/A	Game theory-based algorithm	Multi-domain physical networks

2.4. VNE Based on Network Reconfiguration

Several works have been proposed to improve the VN acceptance ratio through network reconfiguration. For instance, the embedding of already embedded VNs was optimized in [57] so that newly arrived VNs would be easier to embed. Moreover, an efficient VN reconfiguration mechanism aiming at minimizing the migration cost of virtual resource was put forward in [58] and authors in [59] chose the VNs that had greater influence on the embedding objective for reconfiguration to achieve load balancing in SN. Apart from the VN reconfiguration, authors in [60] proposed the paradigm of SN resource reconfiguration in WMN. The channels of wireless radios were strategically reconfigured to increase VN acceptance ratio and InP profit. However, the channel reconfiguration of wireless radio interfaces tends to experience a high cost resulting from the resetting of physical hardwares. In FiWi access network, the SN bandwidth reallocation that changes the transmission time slot of each node by exchanging control messages generates a lower implementation complexity compared with VN reconfiguration and SN channel reallocation. Thus, it is important to coordinate the execution order of different network reconfiguration approaches and make a reasonable compromise between implementation complexity and network performance.

2.5. Mobility in Network Virtualization

Literature cares little about the mobility in network virtualization, where the mobility is either a single user's behavior or a colony-level behavior. For the management of mobile users, massive investigations have been carried out in terms of power control and handover, which are technologically mature. From the colony-level, the mobility of virtual nodes is due to the mobility of its users. The zooming small cell architecture was exploited in our prior work [61] to achieve the energy reservation in 5G networks, where worth noting is the possibility of using the power control method for the management of virtual nodes with low mobility. In addition, in terms of virtual nodes with high mobility, the node migration should be applied. A handover matrix was used in [62, 63] to capture the actual mobility of users, where each element represented the probability of flow migration between two edge routers. The VNs were modeled as a set of flows with specific service priorities. Each flow that connected the virtual gateway and a virtual edge router had specified resource demand. The mobility management procedure assigned intermediate paths for the data forwarding during the handover between the new destination node and the mobility anchor. However, it is not practical to exactly predict the future locations of virtual nodes. Applying node migration after the node goes out of the coverage will cause unavoidable downtime. To address this issue, a virtual cell was constructed for each mobile user in [64], the synchronization among base stations in a virtual cell was maintained and the set of base stations in a virtual cell changed along with the mobility of the user. Thus, for a mobile virtual node, a virtual cell can also be assigned, that is, redundant resources are used for the service quality guarantee. However, this paper focuses on the VNE problem considering the impact of physical resource allocation. Handling the mobility of virtual nodes, which methods can be found in our prior work [61], is complex and out the range of this paper. Thus, the suggested solutions will be proposed in Section 4, leaving for future comprehensive investigation.

3. Substrate Network Model and Initial Resource Allocation

3.1. Overview and Notation

The heterogeneous physical network contains the multi-channel multi-radio WMN and the TDM-PON. In order to support VNE, the resource capacity, especially link bandwidth, is required to be determined initially. Moreover, due to the introduction of network virtualization, which features the indistinguishable logical resource in InP, a common characteristic (e.g., collision domain) is expected to be defined to help resource allocation in both wireless and optical subnetworks. Notation for SN and initial resource allocation is shown as Table 2.

Table 2: Notation for substrate network model and initial resource allocation

Parameter	Description
$G^S = (N^S, L^S)$	Substrate network where N^S and L^S indicate sets of substrate nodes and links respectively
X	Total number of substrate nodes
Z	Total number of substrate links
CAP_n^{CPU}	CPU capacity of substrate node n
$TYPE_n \in \{OLT, ONU, WR\}$	Type of node n, where OLT, ONU and WR denote OLT, ONU and wireless router (gateway), respectively
LCT_n	Location of node n
$CHNL_{(n,r)}^R$	The channel of radio r on node n
CAP_l^{BNDW}	Bandwidth capacity of substrate link l
LOD_l	Bandwidth load of l
D_l	Collision domain of l
$CHNL_l^L$	The channel of l
$POTN_l$	The potential bandwidth capacity of l
AVG_l	The average bandwidth capacity of l

Continued on next page

Parameter	Description
$RANG^R$	The jamming range of wireless radios under given emitting power
$DIS_{(n,n')}$	Distance between nodes n and n'
SRC_l	Source node of l
DET_l	Destination node of l
C_h	The candidate channel indexed by h

3.2. Collision Domain of Substrate Network

Based on the notation in Table 1 and the fact that the bandwidth of wireless links in WMN depends on not only the channels of radio interfaces but also the number of interfered links, the concept of collision domain [65, 66] is introduced and extended to FiWi access network. Specifically, the collision domain D_l of link l is defined as the set of links that interfere with l. To make it more clear, we divide physical link set into $L^S = L^O \cup L^W$ with L^O and L^W denoting the sets of optical and wireless links respectively. In PON subnetwork, as shown in (1), D_l contains all the optical fiber links between Optical Line Terminal (OLT) and ONUs since OLT cannot transmit packets to different ONUs simultaneously.

$$D_l = \{l' \in L^O | SRC_l = SRC_{l'}\}, \forall l \in L^O, \tag{1}$$

where $SRC_l = SRC_{l'}$ implies that the links l and l' share the same source OLT. In WMN subnetwork, the collision domain of l contains the links that are working on the same channel as l and with either their source or destination node locating in the jamming range of nodes SRC_l or DET_l, as shown in (2):

$$D_l = \{l' \in L^W | CHNL^L_{l'} = CHNL^L_l \text{and} (\min(DIS_{(SRC_l,SRC_{l'})}, DIS_{(SRC_l,DET_{l'})}, DIS_{(DET_l,SRC_{l'})}, DIS_{(DET_l,DET_{l'})}) \leq RANG^R)\}, \forall l \in L^W. \tag{2}$$

Particularly, the link itself is also included in its collision domain. It should be noted that all the links in a collision domain should be scheduled to transmit traffics in TDM mode. Thus, the sufficient condition for an interference-free link scheduling is shown as (3):

$$\max \sum_{l' \in D_l} \frac{LOD_{l'}}{CAP_{l'}} \leq 1, \forall l. \tag{3}$$

That is, the collision domain utilization of any links is supposed to be no more than 1. To ensure this condition, we denote the average bandwidth capacity of link l as (4) to estimate the average amount of bandwidth that could be used by each link in the collision domain.

$$AVG_l = \frac{POTN_l}{|D_l|} = \frac{B^D - \sum_{l' \in D_l} LOD_{l'}}{|D_l|} \tag{4}$$

where $|D_l|$ indicates the number of links in D_l and $POTN_l$ represents the maximum bandwidth that can be allocated to l. Note that B^D is the bandwidth capacity of the collision domain. Specifically, for PON, B^D equals to the bandwidth capacity of feeder fiber from an OLT. For WMN, B^D represents the capacity of the allocated wireless channel.

3.3. Initial Resource Allocation

Previous works execute VNE on the basis of the assumption that all the substrate links are equipped with definite bandwidth, ignoring the process of the shared resource allocation to links in the collision domain in FiWi. To address this issue, we propose the WCAB and DBAF algorithms for the channel allocation of WMN subnetwork and the bandwidth allocation of both WMN and PON subnetworks respectively to support subsequent VNE procedure. In addition, the flexibility of resource allocation within each collision domain can be exploited by InP to adjust its resource distribution. Thus, the VNE can be improved, which is considered in section V.

3.3.1. WCAB Algorithm

Based on (3), the objective of WMN channel allocation is formulated as (5):

Minimize:

$$\max \sum_{l' \in D_l} \frac{LOD_{l'}}{CAP_{l'}}, \forall l \tag{5}$$

where the maximum collision domain utilization is minimized. The objective facilitates that the adjacent radios tend to be allocated with different channels so that each radio suffers little interference from its neighbors.

The pseudo-code procedure of the WCAB algorithm is shown as Algorithm 1. Initially, in case a link becomes invalid due to channel confliction and then breaks the whole network connectivity, we propose to build a spanning tree using Breadth First Search (BFS) and assign channels to the links on the tree preferentially (lines 1 - 3). For example in Fig. 3 where the potential substrate links are shown in Fig. 3 (a), assume that there are 1, 1, 2, 1 and 1 radios on nodes a, b, c, d and e, respectively, if the channels C_1, C_2 and C_3 are assigned to the links $<a,b>$, $<c,d>$ and $<c,e>$, then the links $<a,c>$, $<b,c>$ and $<b,d>$ would be forced to be invalid because no residual radio can be used as well as no common channel exists between their source and destination nodes. As a result, the network is divided into two parts as shown in Fig. 3 (b). However, through building the spanning tree, no isolated node would be generated as shown in Fig. 3(c)-(d). Moreover, for the sake of reserving original topology and avoiding traffic migration, the links whose traffic loads are higher than 0 are also assigned the same channel C_1 as links on the spanning tree. After that, the channel update will be carried out to minimize the maximum collision domain utilization. To be specific, according to the number of spare radios on the end nodes, the optional channel with minimum utilization will be utilized (lines 6 - 15). Furthermore, several times of channel update (lines 17 - 20) will be carried out to further decrease the collision domain utilization and improve the network connectivity.

The complexity of WCAB is bounded by the running times of the channel update operation. For each substrate link, in the worst case, $O(Z^2)$ steps are costed for the calculation of collision domain utilization and $O(Z)$ channels are required to be coped with. Thus, the complexity of WCAB can be represented

Algorithm 1 WCAB Algorithm

Input: G^S.
Output: $CHNL_l^L, CHNL_{(n,r)}^R, \forall l \in L^W$, all wireless nodes and radios.

1: Build a tree using BFS with OLT acting as the root;
2: $CHNL_l^L \leftarrow C_1$ for wireless links on the tree or with $LOD_l > 0$;
3: $CHNL_{(SRC_l,1)}^R \leftarrow C_1, CHNL_{(DET_l,1)}^R \leftarrow C_1$;
4: **for** all $l \in L^W$ with $CHNL_l^L = C_1$, **do**
5: Release the radios whose channels have not been utilized by any wireless links by deleting their channel settings;
6: **if** there exist spare radios r and r' on nodes SRC_l and DET_l respectively, **then**
7: Find the channel C_h with the minimum collision domain utilization of l;
8: $CHNL_l^L \leftarrow C_h$;
$CHNL_{(SRC_l,r)}^R \leftarrow C_h, CHNL_{(DET_l,r')}^R \leftarrow C_h$;
9: **else if** there exists only one spare radio (e.g., radio r on node SRC_l,) **then**
10: For all radios on node DET_l, find the radio r' which has the minimum collision domain utilization of l;
11: $\text{CHNL}_l^L \leftarrow CHNL_{(DET_l,r')}^R, CHNL_{(SRC_l,r)}^R \leftarrow CHNL_{(DET_l,r')}^R$;
12: **else if** there exist the same channels on nodes SRC_l and DET_l, **then**
13: For all the shared channels, find the channel C_h with the minimum collision domain utilization of l;
14: $CHNL_l \leftarrow C_h$;
15: **end if**
16: **end for**
17: **for** all $l \in L^W$, **do**
18: Implement the steps of lines 5 - 15;
19: **end for**
20: Repeat the steps of lines 17 - 19 till no wireless link channel can be changed.

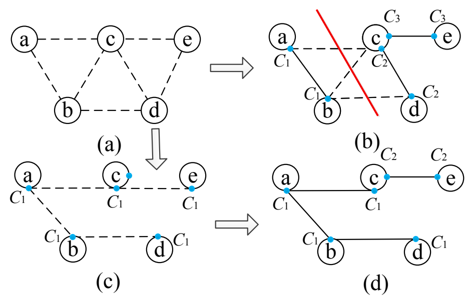

Figure 3: Example of conflicted channel allocation in WMN.

as $O(Z \cdot (Z^2 + Z)) = O(Z^3)$.

3.3.2. DBAF Algorithm

The DBAF algorithm in FiWi access network is described in Algorithm 2, which is executed based on collision domains of PON and WMN after WCAB is applied. In order to satisfy the condition in (3), the link that has the minimum collision domain utilization gains a high priority to be scheduled. Two auxiliary variables, $FLAG_l^L$ and $FLAG_l^D$, are used to represent if link l and the collision domain of l have completed the allocation respectively. For example in Fig. 4, there are four links with the numbers in the parenthesis denoting the link loads. The collision domains of links $<a,b>$ and $<c,d>$ are $D_{<a,b>} = \{<a,b>,<c,d>,<e,f>\}$ and $D_{<c,d>} = \{<a,b>,<c,d>,<g,h>\}$, respectively. Assume $B^D = 10$ so that $AVG_{<a,b>} = (10-7)/3 = 1$ and $AVG_{<c,d>} = (10-4)/3 = 2$. If $<c,d>$ is chosen, then $CAP_{<a,b>}^{BNDW} = CAP_{<c,d>}^{BNDW} = 2+2 = 4$ which conflicts with the bandwidth capacity of $D_{<a,b>}$ with $4{+}4{+}3 > 10$, thus link $<a,b>$ whose $AVG_{<a,b>}$ is lower should be allocated first and the bandwidth allocation result is shown as Fig. 4 (c) with the numbers on links indicating the allocated bandwidth. After the bandwidth allocation, the sufficient condition for interference-free scheduling will be checked.

The DBAF algorithm achieves the computing of AVG_l by $O(Z^2)$ steps and the search of the link with minimum AVG_l with $O(Z)$ steps (line 3). Moreover, the operation of bandwidth setting needs at most $O(Z)$ steps when all links are within only one collision domain (line 4). The operation of AVG_l computation

Algorithm 2 DBAF Algorithm

Input: G^S.

Output: $CAP_l^{BNDW}, \forall l$.

Initialize $FLAG_l^L \leftarrow 0$, $FLAG_l^D \leftarrow 0$, $\forall l$;

while there exists a link that has not been allocated, **do**

  For all $l \in L^S$, $FLAG_l^D = 0$, compute AVG_l and find the l with the minimum AVG_l;

  For all $l' \in D_l$ with $FLAG_{l'}^L = 0$, set $CAP_{l'}^{BNDW} \leftarrow LOD_{l'} + AVG_l$, $FLAG_{l'}^L \leftarrow 1$ and $FLAG_l^D \leftarrow 1$;

end while

if $\sum_{l' \in D_l} LOD_{l'}/CAP_{l'}^{BNDW} \leq 1, \forall l$ **then**

  return true; // the resource reconfiguration succeeds

else

  return false; // the resource reconfiguration fails

end if

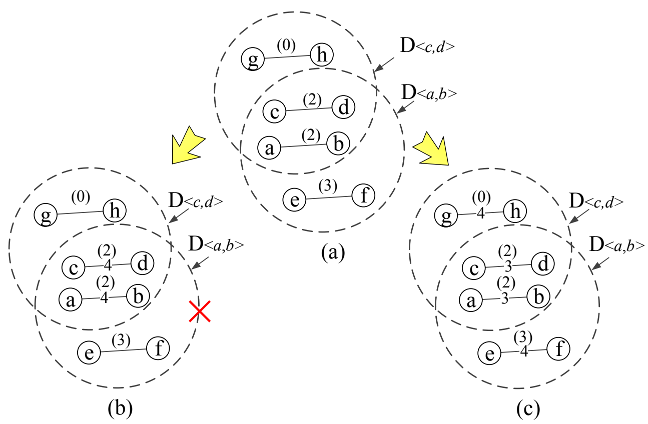

Figure 4: Illustration of link bandwidth allocation order.

and bandwidth setting will be carried out at most $O(Z)$ times when all the substrate links are working on different channels. Thus, the complexity of WCAB is bounded by $O(Z \cdot (Z^2 + Z + Z)) = O(Z^3)$.

4. VNE Problem Formulation

4.1. Notation

On the basis of the physical network that features definite link bandwidth capacity after conducting the initial resource allocation (WCAB and DBAF), embedding of VNs becomes feasible. In virtualized FiWi access network, each VN is formulated as a non-directional graph G^V, representing a specified class of service. The QoS satisfaction is an reflection of QoS degradation [67], which is widely used to guarantee the fairness of multiple users when the network capacity is not enough. Thus, we adopt the QoS satisfaction aiming at serving as many virtual networks as possible, which is consistent with the objective of this paper. The QoS satisfaction requirement of a VN specifies the low bound of allocated resource for the guarantee of the worst tolerable QoS and can be derived based on the expected and worst tolerable requirements on specified QoS metrics in terms of delay, such as the mean delay and the maximum-delay violation probability. Examples will be given in next subsection. Therefore, the QoS satisfaction requirement is endowed to VNs according the requirements of specified delay metrics by service providers. And based on the QoS satisfaction requirement of each VN, the QoS is provided from different levels by controlling the proportion of provided resource over the expected resource demand. However, it is worth noting that best-effort services should be provided to VNs while satisfying the QoS satisfaction requirements, such that better QoS is available for VNs.

In terms of the formulation of mobility, it should be clear that if it refers to the behavior of a single user or a colony. For mobile users, SPs are responsible for the mobility management, which is technologically mature and independent with the VNE. However, a mobile virtual node is a result of the mobility of users that access to the specified kind of service, that is, the colony-level mobility. The mobility of virtual nodes is recognized by SPs through evaluating the best locations and the coverage of nodes according to the distribution of users. But in case SPs cannot handle the location changes using existing granted resources, the requirement needs to be posted to InP, who is responsible for the resource reconfiguration to handle the mobility. Thus, a time factor is required to formulate the resource demand of virtual node and different locations are demanded in different time for the same virtual node. A straightforward method to handle the node mobility is to re-embed a VN according to its real time demands. More comprehensive analysis is given in subsection 5.3.

Table 3: Notation for virtual network requests

Parameter	Description
G^V	Set of VN requests
$G^V_k = (V^V_k, E^V_k)$	Virtual network indexed by k where V^V_k and E^V_k indicate sets of virtual nodes and links respectively
$REQ^{CPU}_{(k,i)}$	CPU demand of virtual node i in G^V_k (i.e., $v_{(k,i)}$)
$REQ^{TYPE}_{(k,i)} \in \{A, T\}$	Type of virtual node $v_{(k,i)}$, where A represents the access node (i.e., OLT), and T represents the transmission node (i.e., ONU or wireless router)
$REQ^{LCT}_{(k,i)}$	Preferred location of $v_{(k,i)}$
$\Delta LCT_{(k,i)}$	Embedding location offset constraint of $v_{(k,i)}$
$REQ^{BNDW}_{(k,j)}$	Bandwidth demand of virtual link j in G^V_k (i.e., $e_{(k,j)}$)
$REQ^{LEN}_{(k,j)}$	Embedding path length constraint of $e_{(k,j)}$
DUR_k	Duration time of G^V_k
$REQ^{QS}_k \in (0, 1]$	QoS satisfaction requirement of G^V_k
$QS^V_{(k,i)}$ or $QS^E_{(k,j)}$	QoS satisfaction of node $v_{(k,i)}$ or link $e_{(k,j)}$

We target to formulate VNE problem in FiWi as an ILP in this section. Apart from the notation characterizing SN and VN requests in Tables 2 and 3 respectively, parameters and variables related with VNE problem are listed as follows:

- ξ_k: Binary variable, taking 1 if G^V_k is embedded successfully, and 0 otherwise;
- $\chi^n_{k,i}$: Binary variable, taking 1 if virtual node $v_{k,i}$ is embedded onto substrate node n, and 0 otherwise;
- $\eta^l_{k,j}$: Binary variable, taking 1 if virtual link $e_{k,j}$ is embedded onto a substrate path that traverses link l, and 0 otherwise;
- $\psi^n_{k,i}$: The volume of CPU that n provides to $v_{k,i}$, $\psi^n_{k,i}$=0 denotes that $v_{k,i}$ is not embedded onto n;

- $\varphi^l_{k,j}$: The volume of bandwidth that l provides to $e_{k,j}$, $\varphi^l_{k,j}$=0 denotes that $e_{k,j}$ is not embedded onto l;
- $LEN_{(k,j)}$: Embedding path length of link $e_{k,j}$;
- I_n: Set of links that come into n;
- O_n: Set of links that go out of n;
- $Profit(G^V_k)$: InP profit obtained by embedding G^V_k;
- α: Income of each unit of resource;
- β: Cost of each unit of resource;
- γ: QoS penalty of each unit of resource;
- ρ^V: Weight of CPU;
- ρ^E: Weight of bandwidth;
- T^{CNR}: Threshold of channel reallocation ;
- μ: Coefficient of channel reallocation, which is a tunable factor for T^{CNR};
- $SRC_{k,j}$: Source node of $e_{k,j}$;
- $DET_{k,j}$: Destination node of $e_{k,j}$.

4.2. Examples for QoS Satisfaction

In this subsection, we take the mean delay and maximum-delay violation probability [68] as examples to interpret the QoS satisfaction. Considering the node CPU and link bandwidth resources, the mean delay and maximum-delay violation probability will be analyzed to reflect the resource requirements and infer the QoS satisfaction requirements. Note that the remaining part of this paper is based on the QoS satisfaction requirements that are potentially extracted from the QoS requirements using methods in this part.

4.2.1. QoS satisfaction based on mean delay

Denote $REQ^{DE}_{(k,i)}$ and $REQ^{DW}_{(k,i)}$ the expected and worst tolerable average delay requirements of virtual node $v_{k,i}$ respectively. The expected and minimum resource demands of $v_{k,i}$ to guarantee the mean delay of data processing can be expressed as (6) and (7) [69]:

$$REQ^{CPU}_{(k,i)} = \frac{\omega_{(k,i)}}{REQ^{DE}_{(k,i)}}, \tag{6}$$

$$REQ^{CPU\_least}_{(k,i)} = \frac{\omega_{(k,i)}}{REQ^{DW}_{(k,i)}}, \tag{7}$$

where $REQ^{CPU}_{(k,i)}$ and $REQ^{CPU\_least}_{(k,i)}$ imply the expected and minimum resource demands of $v_{k,i}$ respectively and $\omega_{(k,i)}$ is the data size to be processed on $v_{k,i}$. Based on the expected and minimum CPU resource demands, the QoS satisfaction requirement of $v_{k,i}$ is:

$$QS^{V}_{(k,i)} = \frac{REQ^{CPU\_least}_{(k,i)}}{REQ^{CPU}_{(k,i)}}. \tag{8}$$

Thus, allocating resource to virtual node with (expected) CPU demand $REQ^{CPU}_{(k,i)}$ and QoS satisfaction requirement $QS^{V}_{(k,i)}$ is equivalent to the problem of allocating resource that approaches $REQ^{CPU}_{(k,i)}$ as much as possible and no less than $REQ^{CPU\_least}_{(k,i)}$.

For the communication delay of traffics on virtual link $e_{k,j}$, denote $REQ^{DE}_{(k,j)}$ and $REQ^{DW}_{(k,j)}$ the expected and worst tolerable mean delay requirements of virtual link $e_{k,j}$ respectively. According to the queueing theory, the communication delay of $e_{k,j}$ includes the waiting and transmitting delay and can be expressed as (9).

$$w = \frac{1}{\mu - \lambda}, \tag{9}$$

where w is the total delay, λ and μ imply the packet arrival rate and the service rate respectively. Considering the fact that each virtual link may be embedded onto multiple physical links, we can derive the expected and minimum bandwidth demands as (11), which is derived from (10), and (12):

$$REQ_{(k,j)}^{DE} = REQ_{(k,j)}^{LEN} \cdot \frac{1}{REQ_{(k,j)}^{BNDW}/pk - \gamma_{(k,j)}^{BNDW}/pk}, \tag{10}$$

$$REQ_{(k,j)}^{BNDW} = \frac{REQ_{(k,j)}^{LEN} \cdot pk}{REQ_{(k,j)}^{DE}} + \gamma_{(k,j)}^{BNDW}, \tag{11}$$

$$REQ_{(k,j)}^{BNDW\_least} = \frac{REQ_{(k,j)}^{LEN} \cdot pk}{REQ_{(k,j)}^{DW}} + \gamma_{(k,j)}^{BNDW}, \tag{12}$$

where $REQ_{(k,j)}^{BNDW}$ and $REQ_{(k,j)}^{BNDW\_least}$ are the expected and minimum bandwidth demands respectively, pk is the average size of packets and $\gamma_{(k,j)}^{BNDW}$ is the volume of data to be transmitted per unit time in the unit of bit/s. Thus, the QoS satisfaction requirement of $e_{k,j}$ can be expressed as:

$$QS_{(k,j)}^{E} = \frac{REQ_{(k,j)}^{BNDW\_least}}{REQ_{(k,j)}^{BNDW}}, \tag{13}$$

which means that allocating resource to virtual link with (expected) bandwidth demand $REQ_{(k,j)}^{BNDW}$, embedding path length constraint $REQ_{(k,j)}^{LEN}$ and QoS satisfaction requirement $QS_{(k,j)}^{E}$ is equivalent to the problem of allocating resource that approaches $REQ_{(k,j)}^{BNDW}$ as much as possible and no less than $REQ_{(k,j)}^{BNDW\_least}$. Overall, given the data size that needs to be processed on each virtual node and the volume of data that needs to be transmitted per unit time on each virtual link, satisfying the QoS satisfaction requirements during the virtual network embedding makes sure the mean delay requirements being guaranteed.

4.2.2. QoS satisfaction based on maximum-delay violation probability

The maximum-delay violation probability is an effective metric for the reliability guarantee for random traffics, which can be defined as:

$$P\left\{D\left(\infty\right) > D_{\max}\right\} \le \varepsilon, \tag{14}$$

where $D(t)$ is the delay at time t, D_{max} represents the maximum delay constraint and ε stands for the delay violation probability requirement, that is, QoS requirement. Note that the maximum-delay violation probability captures the dynamic property of traffics and can be adopted to both computing and communication systems. The equivalent resource can be used for the resource allocation [70]. For Poisson process, the equivalent resource can be formulated in (15).

$$E(\theta) = \frac{\lambda}{T_f \theta}(e^{\theta} - 1), \theta > 0 \tag{15}$$

where $E(\theta)$ with coefficient $\theta > 0$ denotes the equivalent resource of traffics, i.e., the required resource guaranteeing the maximum-delay violation probability constraint. λ is the traffic arrival rate in the unit of packets/s and T_f represents the time length of a frame. The large derivation can be used for the calculation of θ, according to [70], we have:

$$P\left\{D\left(\infty\right) > D_{\max}\right\} \leq e^{-\theta E(\theta) D_{\max}} \leq \varepsilon \tag{16}$$

Therefore, value of θ and the equivalent resource can be shown as in (17) and (18) respectively.

$$\theta = \ln\left[\frac{T_f \ln\left(1/\varepsilon\right)}{\lambda D_{\max}} + 1\right] \tag{17}$$

$$E(\theta) = \frac{\ln\left(1/\varepsilon\right)}{D_{\max}\ln\left[\frac{T_f \ln(1/\varepsilon)}{\lambda D_{\max}} + 1\right]} \tag{18}$$

Given the expected and worst tolerable maximum-delay violation probability requirements of virtual nodes and links, according to (18), we can then obtain the expected and minimum equivalent resources for virtual nodes and links respectively. Finally, the QoS satisfaction requirements are defined as the ratio of minimum values of equivalent resource to the expected values. As a result, the worst tolerable maximum-delay violation probability requirements can be guaranteed when the resources are allocated based on the expected resource requirements and the QoS satisfaction requirements.

4.3. ILP Model for VNE in FiWi

In terms of VNE in FiWi, it is specified that different VNs feature different tolerances of QoS satisfaction, which provides an opportunity for InP to adjust the volume of allocated resource according to its resource availability. Note that a penalty factor is necessitated for the profit of InP to compensate for the QoS satisfaction decline; therefore, the InP profit can be formulated as (19) where three terms, i.e., InP revenue $Rev(G_k^V)$, InP resource cost $Cst(G_k^V)$ and QoS penalty $Pnl(G_k^V)$, are considered.

$$Profit(G_k^V) = DUR_k \cdot [Rev(G_k^V) - Cst(G_k^V) - Pnl(G_k^V)], \tag{19}$$

$$Rev(G_k^V) = \alpha \left[\rho^V \sum_{\forall i} REQ_{(k,i)}^{CPU} + \rho^E \sum_{\forall j} REQ_{(k,j)}^{BNDW} \right], \tag{20}$$

$$\begin{aligned} Cst(G_k^V) = \beta \Bigg[& \rho^V \sum_{\forall i} (QS_{(k,i)}^V \cdot REQ_{(k,i)}^{CPU}) \\ & + \rho^E \sum_{\forall j} (QS_{(k,j)}^E \cdot REQ_{(k,j)}^{BNDW} \cdot LEN_{(k,j)}) \Bigg], \end{aligned} \tag{21}$$

$$\begin{aligned} Pnl(G_k^V) = \gamma \Bigg[& \rho^V \sum_{\forall i} (1 - QS_{(k,i)}^V) \cdot REQ_{(k,i)}^{CPU} \\ & + \rho^E \sum_{\forall j} (1 - QS_{(k,j)}^E) \cdot REQ_{(k,j)}^{BNDW} \cdot LEN_{(k,j)} \Bigg]. \end{aligned} \tag{22}$$

Based on above, the ILP model for VNE problem in FiWi access network is formulated as follows:

Objective:

$$\text{Maximize} : \sum_{k \in G^V} Profit(G_k^V) \cdot \xi_k. \tag{23}$$

Node embedding constraints:

$$\chi_{k,i}^n = \xi_k, \text{ if } REQ_{(k,i)}^{TYPE} = A \text{ and } TYPE_n = OLT, \forall k, i, n, \tag{24}$$

$$\sum_{n \in N^S} \chi_{k,i}^n = \xi_k, \forall k, i, \tag{25}$$

$$\sum_{i \in N_k^V} \chi_{k,i}^n \leq \xi_k, \forall k, n, \tag{26}$$

$$\sum_{n \in N^S} \psi_{k,i}^n = \xi_k \cdot REQ_{(k,i)}^{CPU} \cdot QS_{(k,i)}^V, \forall k, i, \tag{27}$$

$$\sum_{k \in G^V} \sum_{i \in N_k^V} \psi_{k,i}^n \leq CAP_n^{CPU}, \forall n, \tag{28}$$

$$REQ_k^{QS} \leq QS_{(k,i)}^V \leq 1, \forall k, i, \tag{29}$$

$$\chi_{k,i}^n = \psi_{k,i}^n / (REQ_{(k,i)}^{CPU} \cdot QS_{(k,i)}^V), \forall k, i, n. \tag{30}$$

Link embedding constraints:

$$\sum_{l \in I_n} \varphi_{k,j}^l - \sum_{l \in O_n} \varphi_{k,j}^l = QS_{(k,j)}^E \cdot REQ_{(k,j)}^{BNDW} \cdot (-\chi_{k,SRC_{k,j}}^n + \chi_{k,DET_{k,j}}^n), \forall k, j, n, \tag{31}$$

$$\sum_{l \in I_n} \eta_{k,j}^l \leq \xi_k, \forall k, j, n, \tag{32}$$

$$\sum_{l \in O_n} \eta_{k,j}^l \leq \xi_k, \forall k, j, n, \tag{33}$$

$$\sum_{k \in G^V} \sum_{j \in L_k^V} \varphi_{k,j}^l \leq CAP_l^{BNDW}, \forall l, \tag{34}$$

$$\sum_{l \in L^S} \eta_{k,j}^l \leq REQ_{(k,j)}^{LEN}, \forall k, j, \tag{35}$$

$$REQ_k^{QS} \leq QS_{(k,j)}^E \leq 1, \forall k, j, \tag{36}$$

$$\eta_{k,j}^{l} = \varphi_{k,j}^{l} / (REQ_{(k,j)}^{BNDW} \cdot QS_{(k,j)}^{E}), \forall k, j, l. \quad (37)$$

The objective in (23) tends to maximize the total profit of InP from embedding VNs. Equations (24) - (30) elaborate the node embedding constraints, where (24) states that the virtual node with $REQ_{(k,i)}^{TYPE} = A$ should be embedded to OLT. Equations (25) and (26) indicate that one virtual node is limited to be embedded to only one substrate node and virtual nodes from the same VN should be embedded to different substrate nodes, respectively. Equation (27) implies that InP should embed all the virtual nodes with satisfied QoS satisfaction if it accepts the VN. The constraints of CPU capacity and QoS satisfaction are given in (28) and (29), where the QoS satisfaction is defined as the ratio of the volume of resource that InP actually provides over the total resource demand. Equation (30) states the constraint between variables. Moreover, the link embedding constraints including flow conservation (31), path disjointness (32)-(33), bandwidth capacity (34), path length (35), QoS satisfaction (36) and the definition of related variables (37) are respectively formulated.

5. Heuristic VNE Algorithms

Since the VNE problem has been proved to be NP-hard, heuristic algorithms are essential to shorten the solution time and facilitate dynamic VN requests. Initially, based on the resource topology determined by WCAB and DBAF, the QoS satisfaction aware VNE algorithm (VNE-Q) is proposed to enable InP to provide the best-effort service to the VNs that are constrained by different QoS satisfaction requirements. Moreover, aiming at higher InP profit, the network reconfiguration mechanisms are further proposed, i.e., VNE-R, where WCAB and DBAF are exploited again for the channel and bandwidth reallocations.

5.1. QoS Satisfaction aware VNE Algorithm

Based on our prior work [11] where the procedure of VNE-Q in FiWi access network is presented, the maximum InP profit is able to be achieved by adopting the embedding solution with the maximum InP profit for each virtual node and its related links during each embedding stage. For page limitation, we directly analyze the complexity of VNE-Q algorithm.

Firstly, for each VN G_k^V with K virtual nodes, the virtual nodes ranking (e.g., the bubble sorting) costs $O(K^2)$ steps. Besides, in each embedding stage, each virtual node owns at most X candidate substrate nodes. And for each virtual node, there are at most $K-1$ links that will be embedded together when the VN is a complete graph. Moreover, for each virtual link, $O(Z + X \log X)$ steps will be costed by the Dijkstra algorithm. Thus, the complexity of VNE-Q can be represented as $O(K^2+K\cdot X\cdot(K-1)\cdot(Z+X\cdot logX)) = O(K^2\cdot X\cdot(Z+X\cdot logX)))$. Finally, it can be bounded by $O(X^2 \cdot X \cdot (X^2 + X \cdot logX)) = O(X^5 + X^4 \cdot logX)$ if both virtual and substrate networks are complete graphs with the same node number X.

5.2. VNE-Q Algorithm Based on Network Reconfiguration

The integration of wired and wireless subnetworks in FiWi improves the flexibility and diversity of physical resource, which provides an opportunity to increase InP profit through optimizing the resource allocation of SN. The procedure of VNE-Q algorithm based on network reconfiguration in FiWi access network is shown in Algorithm 3. Firstly, considering the complexity of channel switch of wireless radios on network devices, the bandwidth reallocation (lines 10 - 16) is preferably adopted to avoid network congestion when VNE-Q fails in the embedding (line 8). VNs would be embedded onto an extended network $G^{S'}$ and the Algorithm 2 will be implemented to achieve feasible network bandwidth reconfiguration.

Moreover, it is worthwhile to note that the operations of bandwidth reallocation and channel reallocation cannot relieve the virtual node congestion (i.e., there is not any substrate node whose residual CPU capacity is higher than the CPU demand of virtual node); therefore, the virtual network reconfiguration should be directly implemented for the node congestion (line 6). Taking the online feature of VNs into consideration, the embedding results of prior embed-

Algorithm 3 VNE-Q Algorithm based on Network Reconfiguration

Input: $G_k^V, G^S, \alpha, \beta, \gamma, \rho^V, \rho^E$.
Output: $\xi_k, \chi_{k,i}^n, \eta_{k,j}^l, QS_{(k,i)}^V, QS_{(k,j)}^E, \forall i, j, n, l$.

For G_k^V, rank its virtual nodes according to their CPU demand and the network connectivity;
2: Embed n and its related links using VNE-Q;
if the embedding succeeds, **then**
4: $n \leftarrow n+1$, go to line 2 till all the nodes are embedded;
else if node embedding fails **then**
6: go to line 17;
else
8: go to line 10;
end if
10: **Substrate network bandwidth reallocation:**
Build $G^{S'}$ by replacing CAP_l^{BNDW} in G^S with $POTN_l$ and re-embed n and its related links on $G^{S'}$;
12: **if** the embedding succeeds, **then**
Reallocate the bandwidth of $G^{S'}$ using Algorithm 2;
14: **else**
go to line 17
16: **end if**
Virtual network reconfiguration:
18: Release some VNs and re-embed them together with G_k^V;
if the embedding succeeds, **then** $\xi_k \leftarrow 1$, **return true**;
20: **else** recover the embedding of the released VNs and go to line 22;
end if
22: **Substrate network channel reallocation:**
if the revenue of G_k^V is higher than T^{CNR}, **then**
24: Build $G^{S''}$ by replacing CAP_l^{BNDW} in G^S with ∞ and re-embed G_k^V on $G^{S''}$;
if G_k^V is embedded successfully, **then**
26: Apply Algorithms 1 and 2 on $G^{S''}$;
else
28: $\xi_k \leftarrow 0$, reject G_k^V, **return false**;
end if
30: **end if**

ded VNs generally leave an influence on subsequent ones. Thus, it is feasible to accept more VNs by changing the embedding order, that is, releasing and re-embedding the existing VNs. For example in Fig. 5 where numbers beside nodes and on the links indicate node CPU demands (residual CPU capacity) and link bandwidth demands (residual bandwidth capacity) respectively. Assume there are two VNs labeled by VN 1 and VN 2 (the QoS satisfaction requirements are 1), whose access nodes are node A and D. If VN 1 has already been embedded on SN as shown in Fig. 5 (a), then there is not any substrate node that can satisfy the CPU demand of node F in VN 2. However, if we release VN 1 and embed VN 2 and VN 1 in sequence again, the feasible embedding solutions of VN 1 and VN 2 can be found as shown in Fig. 5 (b). Thus, the virtual network reconfiguration can redistribute the physical resource to VNs and then improve VN acceptance ratio. In addition, in terms of releasing VNs, the following tips should be considered. First, only VNs whose residual duration time is longer than given threshold are supposed to be released because it is unnecessary to migrate a VN whose resource would be recycled soon. Second, the VNs whose InP revenues are lower than the congested VN can be migrated since VNs with higher InP revenue will be harder to embed due to high resource demands.

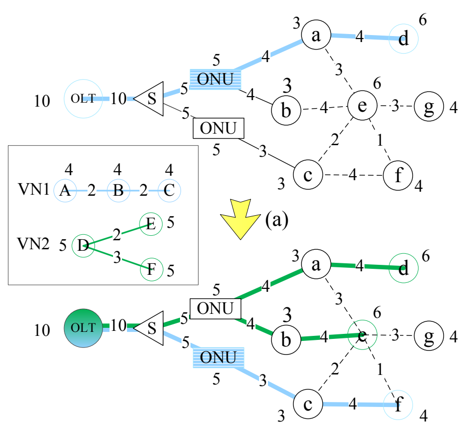

Figure 5: Illustration of virtual network reconfiguration.

Once link bandwidth reallocation and virtual network reconfiguration fail to embed G_k^V, the SN channel reallocation will be executed (lines 22 - 30). In order to restrict the executing frequency of channel switch of radio interfaces, a reallocation threshold is introduced to specify that only the VNs whose InP revenues are higher than T^{CNR} can be optimized through channel reallocation. T^{CNR} is defined as (38):

$$T^{CNR} = \mu \cdot DUR_{avg} \cdot (\rho^V \cdot X_{avg} \cdot REQ_{avg}^{CPU} + \rho^E \cdot Z_{avg} \cdot REQ_{avg}^{BNDW}). \tag{38}$$

where DUR_{avg} represents the average duration time of VNs. REQ_{avg}^{CPU} and REQ_{avg}^{BNDW} denote the average CPU demand of virtual nodes and the average bandwidth demand of virtual links, respectively. X_{avg} and Z_{avg} indicate the average numbers of nodes and links in VNs. The adjustable coefficient μ that cooperates with the average VN revenue determines the possibility for the implementation of channel reallocation. After the congested VN is embedded onto an extended infinite-capacity network $G^{S''}$, Algorithms 1 and 2 will be carried out to find a feasible network resource allocation solution. Note that only when WCAB (Algorithm 1) and DBA (Algorithm 2) return true, can the VN be regarded as a successful embedding. Otherwise, all the physical resource will be recovered as well as the VN would be rejected.

These network reconfiguration mechanisms, i.e., substrate network bandwidth reallocation, virtual network reconfiguration and substrate network channel reallocation, can be easily implemented in parallel with the complexity of $O(Z^3) \leq O(X^6)$, $O(X^5 + X^4 \cdot logX)$ and $O(Z^3)$, respectively; therefore, the complexity of the VNE-Q algorithm based on network reconfiguration in FiWi access network is $O(X^6)$.

5.3. Discussion on Mobility Management in Virtualized FiWi Access Network

In this subsection, we propose a framework for the managements of mobile users and mobile nodes with low and high mobile rates, respectively, in virtualized FiWi access network. And then the opportunities and challenges are analyzed. Figure 6 depicts the mobility management of users. Given that a VN with five virtual nodes (i.e., nodes a, b, c, d and e) has been successfully embedded onto the physical network. A mobile user who has the authority of requiring services provided by the VN moves from locations L1, L2 to L3. For the scenario of low mobility, the power control mechanism can be applied to the router. For example, when the user, who accesses to virtual node b, moves from L1 to L2, the router that node b embeds can increase the transmitting power such that its coverage is enlarged and can cover L2. When the user moves fast and even the largest allowed transmitting power cannot cover the new location L3, the handover occurs. The user will change to access to node d, which provides the same service as node b because they belonged to the same VN.

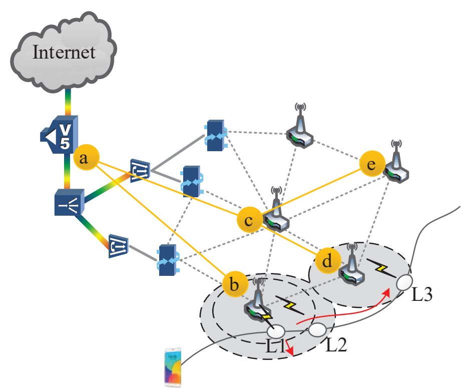

Figure 6: Mobility management of end users.

The mobility management of users is technologically mature in cellular networks and can almost be directly exploited by SPs in network virtualization. However, protocols need to be designed for the admission control and the handover of moving users in network virtualization because each SP has its restriction of resource controlling, i.e., only the resource that have been allocated to the SP can be used. In addition, when the locations of users shows a colony-level change, it is better to adjust the allocated resource as well as the locations of virtual nodes for the purpose of high resource utilization and better QoS, that is the mobility management of virtual nodes.

It is specified that a virtual node is embedded onto a physical node that satisfies the location constraints, which are displayed as a best location requirement and an allowed location offset. While the best location requirement of a virtual node changes, InP needs to check if the original embedding covers the new required location and make some adjustment accordingly. The mobility management of virtual nodes is illustrated in Fig. 7. Given that a VN with four nodes (i.e., nodes a, b, c and d) has been successfully embedded onto the substrate network. We assume a node b with low mobility and a node d with high mobility. For virtual nodes with low mobility, the concept of zooming cell can be adopted. That is, the continuous service providing is guaranteed by adjusting the transmitting power of the embedded router dynamically and making sure the location constraints of node b are satisfied. The reference of VNE algorithm based on zooming cell can be found in our prior work [60]. In terms of mobility management of virtual node with fast moving speed, for example node d in Fig. 7, traditional methods apply node migration mechanisms by determining a new embedding location. However, it is hard to predict the future locations of virtual nodes practically, leading to unavoidable downtime. To address this issue, we propose to construct a virtual cell for each virtual node for the mobility management for the first time. Assume virtual node d is originally embedded on d1 and the virtual cell is marked as the red circular line that contains d1. The virtual cell is a set of routers among which the synchronization is always maintained. When node d moves, it changes its working router (one of the synchronizing nodes in the virtual cell) dynamically. Moreover, the set of nodes in the virtual cell changes along with the movement of the virtual node. For example, the red circular line that contains d2 and d3 is the virtual cell when the node is at L4 and the red circular line that contains d3 but no d2 is the virtual cell when the node is at L5. As a result, no downtime would be incurred as long as appropriate virtual cells are constructed for moving virtual nodes.

However, there are some challenges for the management of virtual nodes both with low and high mobilities. Specifically, for low mobility, the interference among wireless nodes needs to be considered in the power control mechanism.

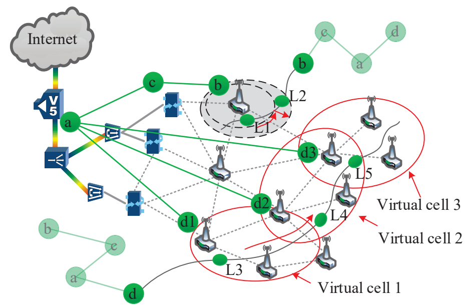

Figure 7: Mobility management of virtual nodes based on zooming and virtual cells.

Moreover, the fairness among virtual nodes that are embedded onto the same substrate node should be assured in terms of power allocation. In addition, for high mobility, the challenges are as follows. First, in order to construct virtual cell, it is better to predict the moving trend of virtual nodes such that routers with high embedding probabilities could be included. Moreover, taking an insight into the virtual cell mechanism, it is actually to guarantee the continuity of services through sacrificing redundant resources, since resources are needed for the synchronization. Thus, there is a tradeoff between less downtime and high resource utilization, which is worth for future research. Furthermore, the synchronization maintaining among nodes in the virtual cell calls for efficient protocols to reduce communicating cost. Last but not the least, it is important to select the best working node among all the synchronizing nodes to facilitate virtual cell construction and reduce resource cost. Overall, the mobility management in network virtualization is a complex topic and requires further investigation.

6. Performance Evaluation

6.1. Simulation Settings and Comparative Approaches

In this section, both static and dynamic VN requests are taken into account to evaluate the effectiveness and robustness of proposed VNE-Q algorithm based on network reconfiguration, which is annotated by VNE-R. And comparisons are made with the optimum solution gained from solving the ILP and other heuristic approaches as follows.

- ILP-Q: The optimal embedding solution obtained by solving the ILP of QoS satisfaction aware VNE problem (in subsection IV.B) using IBM ILOG CPLEX;
- Two-Stage VNE: The typical two-stage VNE algorithm [59, 71] that implements the embedding of virtual nodes and virtual links separately;
- One-Stage VNE: The simplified VNE-Q algorithm (in subsection V.A) that implements the embedding of virtual nodes and virtual links jointly without the consideration of QoS satisfaction, that is, $QS_k = 1, \forall k$;
- VNE-Q: The QoS satisfaction aware VNE algorithm;
- VNE-PR: The VNE-Q algorithm with SN bandwidth reallocation mechanism;
- VNE-PVR: The VNE-Q algorithm with SN bandwidth reallocation and VN reconfiguration mechanisms;

Common parameter settings related with substrate and virtual networks are shown as Table 4, and the simulations are implemented based on the settings of $\alpha = 5, \beta = 1, \gamma = 2, \rho^V = 1$ and $\rho^E = 1$.

Table 4: Parameter settings

Parameter	Value
Substrate Network	
Number of OLTs	1
Number of wireless gateways to each ONU	2
Number of radios on each wireless router	[2,3]
CPU capacity of OLT	[500,1000]
CPU capacity of ONU and wireless router	[50,100]
Bandwidth capacity of optical fiber	1000 Mbps
Bandwidth capacity of cable	54 Mbps
Bandwidth capacity of wireless link	54 Mbps

Continued on next page

Parameter	Value
Transmission span of wireless radio	100 m
Interference span of wireless radio	100 m
Number of candidate channels in WMN	11 for IEEE 802.11a
Virtual Network	
Possibility of virtual link existing between any two virtual nodes	0.5
Duration	Average 20 in exponential distribution
CPU demand of virtual node	[10,20]
Bandwidth demand of virtual link	[2,10] Mbps
Embedding path length constraint	[5,10] hops

6.2. Performance Analysis for Static VNs

Firstly, 20 static VNs are generated as service requests for a small-scale SN where 2 ONUs and 17 wireless routers (gateways) are randomly deployed in a 200m $\times$ 200m square area.

Randomly generating 2 to 6 virtual nodes for each VN, Fig. 8 shows the InP profit under different QoS satisfaction requirements and Fig. 9 is the corresponding VN acceptance ratio. It is obvious that, with the increase of average QoS satisfaction requirement, both InP profit and VN acceptance ratio of ILP-Q, VNE-Q and VNE-R witness the decreasing trend since higher QoS satisfaction requirement prevents InP from embedding more VNs due to the increased volume of resource that InP has to provide for each VN. In comparison, both Two-Stage VNE and One-Stage VNE algorithms maintain a steady performance because they completely ignore the QoS satisfaction.

In addition, the InP profit and VN acceptance ratio of ILP-Q are the highest because solving ILP enables to achieve global optimal solutions. While the proposed VNE-R, where $\mu = 0$ is considered, produces the near-optimal solution that is remarkably better than that of Two-Stage VNE, One-Stage VNE and VNE-Q, which can be explained as follows. First, One-Stage VNE achieves the embedding of virtual nodes and links in the manner of joint optimization which outperforms the Two-Stage VNE. Besides, VNE-Q allows the QoS satisfaction

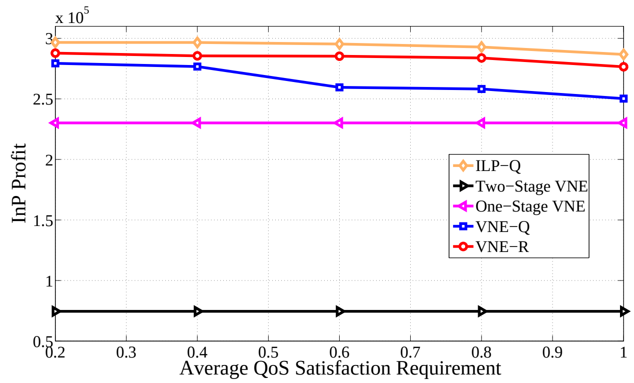

Figure 8: Comparison of InP profit under different REQ_k^{QS}.

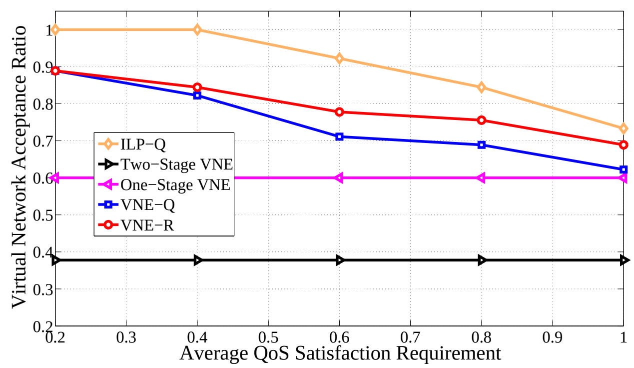

Figure 9: Comparison of VN acceptance ratio under different REQ_k^{QS}.

decline by introducing penalty, resulting in more accepted VNs as well as higher InP profit. Furthermore, the network reconfiguration mechanisms in VNE-R promote the adjusting of resource configuration on the basis of VNE-Q from both virtual and physical networks perspectives. Thus, there is the highest InP profits in VNE-R.

With the QoS satisfaction requirement ranging from 0.5 to 1, Figs. 10 and 11 demonstrate the comparisons of InP profit and VN acceptance ratio under different VN sizes (i.e., number of virtual nodes in each VN), respectively. In Fig. 11, the VN acceptance ratio decreases gradually with the increase of average VN size due to the difficulty of large-scale VN embedding. However, the InP

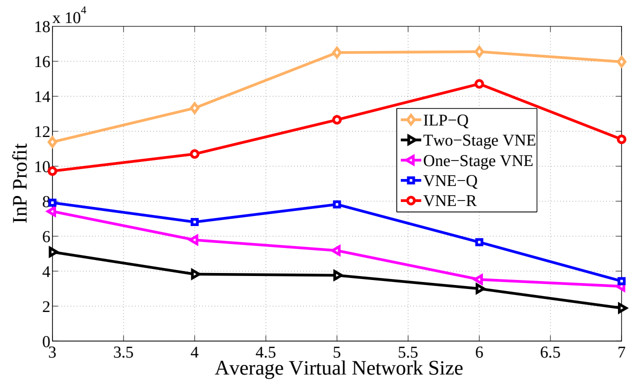

Figure 10: Comparison of InP profit under different VN sizes.

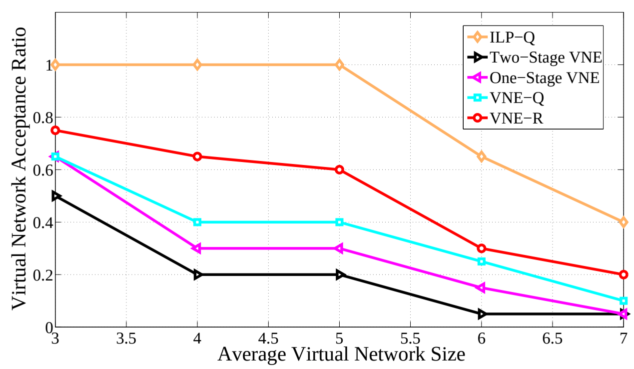

Figure 11: Comparison of VN acceptance ratio under different VN sizes.

profits of ILP-Q, VNE-R and VNE-Q in Fig. 10 first grow up and then decline. The peak profits of these three algorithms occur when the VN size reaches to 5, 6 and 5 respectively. That is because larger VN size contributes to higher InP profit even though the VN acceptance ratio declines. However, when the VN size gets large enough, it is difficult for InP to accept more VNs such that the InP profit drops. Obviously, VNE-R can achieve higher InP profit and VN acceptance ratio than VNE-Q, One-Stage VNE and Two-Stage VNE and performs nearest to ILP-Q.

6.3. Performance Analysis for Dynamic VNs

In dynamic VNs scenario where VN requests arrive according to Poisson distribution, we deploy 4 ONUs and a mesh-like 7×7 wireless routers (gateways) array in a 500m $\times$ 500m square area. The QoS satisfaction requirement varies from 0.5 to 1.

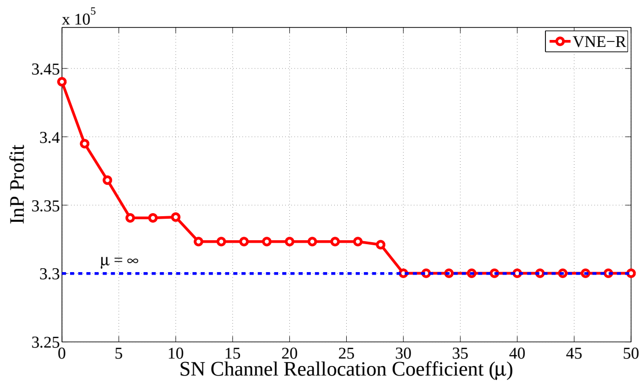

Figure 12: InP profit under different μ.

In terms of the evaluation in the effectiveness of channel reallocation threshold in VNE-R, the influence of channel reallocation coefficient μ on InP profit is analyzed as shown in Fig. 12. The InP profit decreases with the increase of μ for the reason that higher T^{CNR} prevents VNs from being re-embedded by substrate network channel reallocation. Moreover, when μ goes higher than 30, the InP profit of VNE-R converges to the same value as $\mu = \infty$, implying that there is not any VN that can trigger the channel reallocation. In Fig. 13 the performance of VNE-R with $\mu = 10$ and $\mu = 0$ are evaluated and compared with VNE-PVR, VNE-PR, VNE-Q, One-Stage VNE and Two-Stage VNE. Note that the performance of VNE-R when $\mu = \infty$ is identical to that of VNE-PVR. It is significant that with the elapse of simulation time, the InP profit increases due to more VNs accepted. In addition, the InP profit of VNE-R with $\mu = 0$ is the highest owing to the always implemented channel reallocation, which outperforms the VNE-R with $\mu = 10$ and other five approaches. Besides, in VNE-PR and VNE-PVR, the operations of substrate network bandwidth reallocation and virtual network reconfiguration contribute to a higher InP profit compared with One-Stage VNE and Two-Stage VNE approaches.

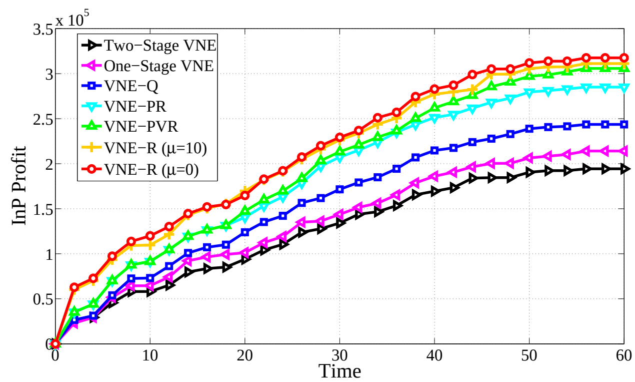

Figure 13: Comparison of InP profit along with time.

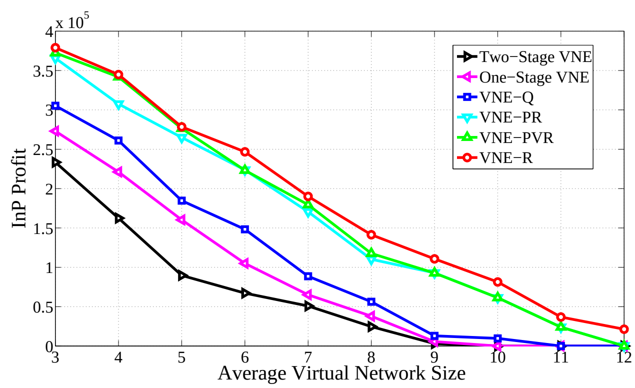

Figure 14: Comparison of InP profit under different VN sizes for dynamic VNs.

On the basis of setting $\mu = 0$ and VN arrival rate with 2 VNs in each time unit, Figs. 14 and 15 show the InP profit and VN acceptance ratio under different VN sizes where both InP profit and VN acceptance ratio decline as the average VN size increases. And obviously the performance of VNE-R is the best of all regardless VN size.

In addition, randomly generating 2∼6 virtual nodes for each VN, Figs. 16 and 17 indicate the InP profit and VN acceptance ratio under different VN arrival rates. While the VN acceptance ratio drop with the increase of VN

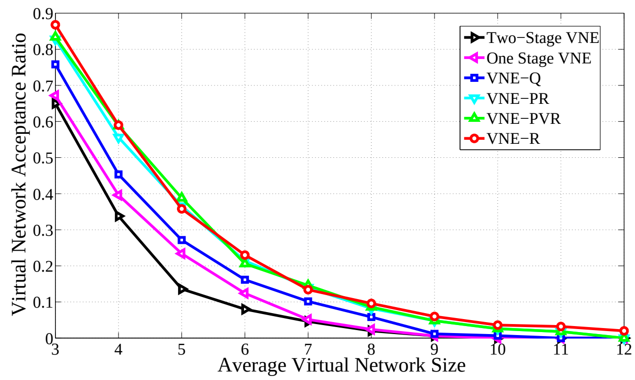

Figure 15: Comparison of VN acceptance ratio under different VN sizes for dynamic VNs.

arrival rate, the InP profit experience an upward trend because even though the limited resource in InP make it hard to accept too much VNs arrived within one time window, the total number of accepted VN requests remains increasing when the average VN arrival rate is higher, resulting in an ascending InP profit. Moreover, in aspect of the effectiveness of network reconfiguration, it can be observed in Fig. 14 that VNE-R gains a significant 54.26% $\sim$ 18.63% higher InP profit than that of VNE-Q and achieves 123.84% $\sim$ 45.77% higher InP profit compared with Two-Stage VNE algorithm when the average VN arrival rate varies from 0.5 to 6.

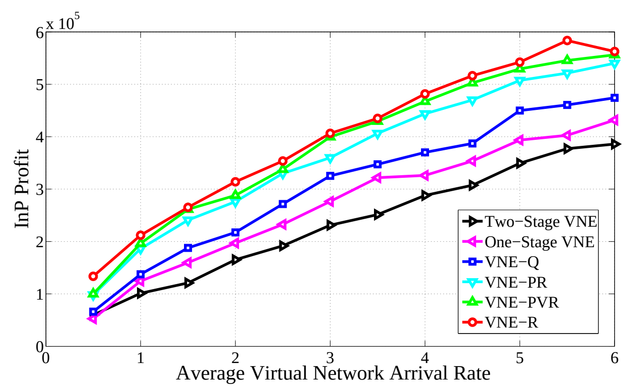

Figure 16: Comparison of InP profit under different VN arrival rates.

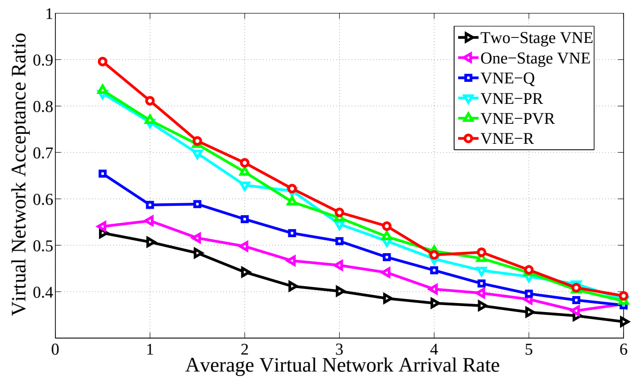

Figure 17: Comparison of VN acceptance ratio under different VN arrival rates.

7. Conclusion

In this paper, we focused on the problem of VNE in FiWi access network for the service providing with tolerable QoS satisfaction. First, two resource allocation algorithms, i.e., WCAB and DBAF, were proposed to facilitate the VNE and network reconfiguration. Then, aiming at maximizing InP profit, the ILP model for the problem of VNE with QoS satisfaction requirement was formulated, according to which, the QoS satisfaction aware VNE algorithm as well as the network reconfiguration mechanisms were further proposed to support online VN requests. Moreover, a primer of mobility management in network virtualization was investigated with proposed solutions for FiWi. Finally, we conducted extensive simulation tests and the results showed that the proposed VNE algorithms could achieve a significant higher InP profit than the typical two-stage VNE algorithm in a wide range of average VN arrival rates.

References

[1] M. Maier, M. Levesque, Dependable Fiber-Wireless (FiWi) access networks and their role in a sustainable third industrial revolution economy, IEEE Transactions on Reliability 63 (2) (2014) 386–400. doi:10.1109/TR.2014.2315953.

[2] Y. Yu, C. Ranaweera, C. Lim, E. Wong, L. Guo, Y. Liu, A. Nirmalathas, Optimization and deployment of survivable Fiber-Wireless (FiWi) access networks with integrated small cell and WiFi, in: Proc. IEEE International Conference on Ubiquitous Wireless Broadband (ICUWB), 2015, pp. 1–5. doi:10.1109/ICUWB.2015.7324491.

[3] N. Correia, G. Schutz, A. Barradas, Correlation-based energy saving approach for smart fiber wireless networks, IEEE/OSA Journal of Optical Communications and Networking 7 (6) (2015) 525–539. doi:10.1364/JOCN.7.000525.

[4] A. Jarray, A. Karmouch, Decomposition approaches for virtual network embedding with one-shot node and link mapping, IEEE/ACM Transactions on Networking 23 (3) (2015) 1012–1025. doi:10.1109/TNET.2014.2312928.

[5] L. Nonde, T. E. H. Elgorashi, J. M. H. Elmirghani, Cloud virtual network embedding: Profit, power and acceptance, in: Proc. IEEE Global Communications Conference (GLOBECOM), 2015, pp. 1–6. doi:10.1109/GLOCOM.2015.7417788.

[6] C. Besiktas, D. Gözüpek, A. Ulas, E. Lokma, Secure virtual network embedding with flexible bandwidth-based revenue maximization, Computer Networks 121 (5) (2017) 89 – 99. doi:https://doi.org/10.1016/j.comnet.2017.04.020.

[7] S. Abdelwahab, B. Hamdaoui, M. Guizani, T. Znati, Efficient virtual network embedding with backtrack avoidance for dynamic wireless networks, IEEE Transactions on Wireless Communications 15 (4) (2016) 2669–2683. doi:10.1109/TWC.2015.2507134.

[8] F. Esposito, D. D. Paola, I. Matta, On distributed virtual network embedding with guarantees, IEEE/ACM Transactions on networking 24 (1) (2016) 569–582. doi:10.1109/TNET.2014.2312928.

[9] A. Marotta, F. D. Andreagiovanni, A. Kassler, E. Zola, On the energy cost of robustness for green virtual network function placement in 5g virtualized infrastructures, Computer Networks 125 (9) (2017) 64 – 75. doi:https://doi.org/10.1016/j.comnet.2017.04.045.

[10] Y. Li, X. Zheng, D. j. Lu, Virtual network embedding based on multi-objective group search optimizer, in: Proc. International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA), 2015, pp. 598–601. doi:10.1109/BWCCA.2015.27.

[11] P. Han, L. Guo, Y. Liu, X. Wei, J. Hou, X. Han, A new virtual network embedding framework based on qos satisfaction and network reconfiguration for fiber-wireless access network, in: Proc. IEEE International Conference on Communications (ICC), 2016, pp. 1–7. doi:10.1109/ICC.2016.7511480.

[12] M. Xu, J. Yan, J. Zhang, F. Lu, J. Wang, L. Cheng, D. Guidotti, G. Chang, Bidirectional fiber-wireless access technology for 5g mobile spectral aggregation and cell densification, IEEE/OSA Journal of Optical Communications and Networking 8 (12) (2016) B104–B110. doi:10.1364/JOCN.8.00B104.

[13] T. de Andrade, N. da Fonseca, L. Villas, O. Branquinho, Protocols for wireless sensors networks connected by radio-over-fiber links, IEEE Systems Journal 12 (1) (2018) 905–915. doi:10.1109/JSYST.2016.2601303.

[14] F. Kabonzo, M. Dilshad, Impact of radio over fiber technology for integrated 5g front and back-haul applications, in: Proc. IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2017, pp. 1077–1081. doi:10.1109/IAEAC.2017.8054179.

[15] Y. Yu, C. Ranaweera, C. Lim, L. Guo, Y. Liu, A. Nirmalathas, E. Wong, Hybrid fiber-wireless network: an optimization framework for survivable deployment, IEEE/OSA Journal of Optical Communications and Networking 9 (6) (2017) 466–478. doi:10.1364/JOCN.9.000466.

[16] J. Wang, K. Lu, J. Wang, C. Qiao, A joint network coding and device association design for optimal local data exchange in fiber-wireless access network, Journal of Lightwave Technology 35 (11) (2017) 2046–2062. doi:10.1109/JLT.2016.2647254.

[17] G. Torfs, et al., Atto: Wireless networking at fiber speed, Journal of Lightwave Technology 36 (8) (2018) 1468–1477. doi:10.1109/JLT.2017.2783038.

[18] B. P. Rimal, D. P. Van, M. Maier, Cloudlet enhanced fiber-wireless access networks for mobile-edge computing, IEEE Transactions on Wireless Communications 16 (6) (2017) 3601–3618. doi:10.1109/TWC.2017.2685578.

[19] B. P. Rimal, D. P. Van, M. Maier, Mobile edge computing empowered fiber-wireless access networks in the 5g era, IEEE Communications Magazine 55 (2) (2017) 192–200. doi:10.1109/MCOM.2017.1600156CM.

[20] D. P. Van, B. P. Rimal, J. Chen, P. Monti, L. Wosinska, M. Maier, Power-saving methods for internet of things over converged fiber-wireless access networks, IEEE Communications Magazine 54 (11) (2016) 166–175. doi:10.1109/MCOM.2016.1500635CM.

[21] A. Sarigiannidis, P. Nicopolitidis, Quality-of-service-aware fair bandwidth allocation scheme for fibre wireless networks, IET Networks 5 (3) (2016) 56–63. doi:10.1049/iet-net.2015.0090.

[22] H. Yang, W. Bai, Y. Tan, A. Yu, J. Zhang, Y. Lee, Y. Ji, Multi-dimensional resources integration for service provisioning in 5g cloud radio over fiber network, in: Poc. European Conference on Optical Communication (ECOC), 2016, pp. 1–3. doi:10.1038/srep30678.

[23] H. Yang, Y. He, J. Zhang, Y. Ji, W. Bai, Y. Lee, Performance evaluation of multi-stratum resources optimization with network functions virtualization for cloud-based radio over optical fiber networks, Optics express 24 (8) (2016) 8666—8678. doi:10.1364/oe.24.008666.

[24] H. Yang, J. Zhang, Y. Ji, Y. Lee, C-rofn: multi-stratum resources optimization for cloud-based radio over optical fiber networks, IEEE Communications Magazine 54 (8) (2016) 118–125. doi:10.1109/MCOM.2016.7537186.

[25] Q. Dai, G. Shou, Y. Hu, Z. Guo, A general model for hybrid fiber-wireless (fiwi) access network virtualization, in: Proc. IEEE International Conference on Communications Workshops (ICC), 2013, pp. 858–862. doi:10.1109/ICCW.2013.6649354.

[26] Q. Dai, J. Zou, G. Shou, Y. Hu, Z. Guo, Network virtualization based seamless networking scheme for fiber-wireless (fiwi) networks, China Communications 11 (5) (2014) 1–16. doi:10.1109/CC.2014.6880456.

[27] X. Meng, G. Shou, Y. Hu, Z. Guo, Efficient load balancing multipath algorithm for fiber-wireless network virtualization, in: Proc. International Conference on Information and Communications Technologies (ICT), 2014, pp. 1–6. doi:10.1049/cp.2014.0579.

[28] L. R. Bays, L. P. Gaspary, R. Ahmed, R. Boutaba, Virtual network embedding in software-defined networks, in: Proc. IEEE/IFIP Network Operations and Management Symposium (NOMS), 2016, pp. 10–18. doi:10.1109/NOMS.2016.7502791.

[29] H. Cao, Y. Zhu, G. Zheng, L. Yang, A novel optimal mapping algorithm with less computational complexity for virtual network embedding, IEEE Transactions on Network and Service Management 15 (1) (2018) 356–371. doi:10.1109/TNSM.2017.2778106.

[30] H. Cao, Z. Qu, Y. Xue, L. Yang, Efficient virtual network embedding algorithm based on restrictive selection and optimization theory approach, China Communications 14 (10) (2017) 39–60. doi:10.1109/CC.2017.8107631.

[31] O. Soualah, N. Aitsaadi, I. Fajjari, A novel reactive survivable virtual network embedding scheme based on game theory, IEEE Transactions on Net-

work and Service Management 14 (3) (2017) 569–585. doi:10.1109/TNSM.2017.2717019.

[32] N. Shahriar, R. Ahmed, S. R. Chowdhury, M. M. A. Khan, R. Boutaba, J. Mitra, F. Zeng, Connectivity-aware virtual network embedding, in: Proc. Networking Conference (IFIP Networking) and Workshops, 2016, pp. 46–54. doi:10.1109/IFIPNetworking.2016.7497249.

[33] J. Li, N. Zhang, Q. Ye, W. Shi, W. Zhuang, X. Shen, Joint resource allocation and online virtual network embedding for 5g networks, in: Proc. IEEE Global Communications Conference (GLOBECOM), 2017, pp. 1–6. doi:10.1109/GLOCOM.2017.8254072.

[34] M. Yu, Y. Yi, J. Rexford, M. Chiang, Rethinking virtual network embedding: Substrate support for path splitting and migration, SIGCOMM Comput. Commun. Rev. 38 (2) (2008) 17–29. doi:10.1145/1355734.1355737.

[35] Y. Zhu, M. Ammar, Algorithms for assigning substrate network resources to virtual network components, in: Proc. IEEE International Conference on Computer Communications (INFOCOM), 2006, pp. 1–12. doi:10.1109/INFOCOM.2006.322.

[36] Z. Wang, Y. Han, T. Lin, H. Tang, S. Ci, Virtual network embedding by exploiting topological information, in: Proc. IEEE Global Communications Conference (GLOBECOM), 2012, pp. 2603–2608. doi:10.1109/GLOCOM.2012.6503509.

[37] L. Gong, Y. Wen, Z. Zhu, T. Lee, Toward profit-seeking virtual network embedding algorithm via global resource capacity, in: Proc. IEEE Conference on Computer Communications (INFOCOM), 2014, pp. 1–9. doi:10.1109/INFOCOM.2014.6847918.

[38] S. R. Chowdhury, R. Ahmed, M. M. A. Khan, N. Shahriar, R. Boutaba, J. Mitra, F. Zeng, Dedicated protection for survivable virtual network em-

bedding, IEEE Transactions on Network and Service Management 13 (4) (2016) 913–926. doi:10.1109/TNSM.2016.2574239.

[39] Z. Xu, W. Liang, Q. Xia, Efficient embedding of virtual networks to distributed clouds via exploring periodic resource demands, IEEE Transactions on Cloud Computing PP (99) (2016) 1–1. doi:10.1109/TCC.2016.2535215.

[40] L. Yin, Z. Chen, L. Qiu, Y. Wen, Interference based virtual network embedding, in: Proc. IEEE International Conference on Communications (ICC), 2016, pp. 1–6. doi:10.1109/ICC.2016.7510692.

[41] H. Cao, L. Yang, H. Zhu, Novel node-ranking approach and multiple topology attributes-based embedding algorithm for single-domain virtual network embedding, IEEE Internet of Things Journal 5 (1) (2018) 108–120. doi:10.1109/JIOT.2017.2773489.

[42] P. Zhang, H. Yao, Y. Liu, virtual network embedding based on computing, network and storage resource constraints, IEEE Internet of Things Journal PP (99) (2017) 1–1. doi:10.1109/JIOT.2017.2726120.

[43] P. Zhang, H. Yao, Y. Liu, Virtual network embedding based on the degree and clustering coefficient information, IEEE Access 4 (2016) 8572–8580. doi:10.1109/ACCESS.2016.2632421.

[44] S. Haeri, L. Trajković, Virtual network embedding via monte carlo tree search, IEEE Transactions on Cybernetics 48 (2) (2018) 510–521. doi:10.1109/TCYB.2016.2645123.

[45] X. Cheng, S. Su, Z. Zhang, K. Shuang, F. Yang, Y. Luo, J. Wang, Virtual network embedding through topology awareness and optimization, Computer Networks 56 (6) (2012) 1797–1813. doi:https://doi.org/10.1016/j.comnet.2012.01.022.

[46] H. Cao, Y. Zhu, L. Yang, G. Zheng, A efficient mapping algorithm with novel node-ranking approach for embedding virtual networks, IEEE Access 5 (2017) 22054–22066. doi:10.1109/ACCESS.2017.2761840.

[47] F. Bianchi, F. L. Presti, A markov reward model based greedy heuristic for the virtual network embedding problem, in: Proc. IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), 2016, pp. 373–378. doi:10.1109/MASCOTS.2016.55.

[48] J. Wang, W. Chen, H. Cong, Z. Zhan, J. Zhang, An ant colony system based virtual network embedding algorithm, in: Proc. IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017, pp. 1805–1810. doi:10.1109/SMC.2017.8122878.

[49] I. Houidi, W. Louati, D. Zeghlache, A distributed and autonomic virtual network mapping framework, in: Proc. Fourth International Conference on Autonomic and Autonomous Systems (ICAS), 2008, pp. 241–247. doi:10.1109/ICAS.2008.40.

[50] S. R. Chowdhury, S. Ayoubi, R. Ahmed, N. Shahriar, R. Boutaba, J. Mitra, L. Liu, Mule: Multi-layer virtual network embedding, in: Proc. International Conference on Network and Service Management (CNSM), 2017, pp. 1–9. doi:10.23919/CNSM.2017.8256005.

[51] J. Lischka, H. Karl, A virtual network mapping algorithm based on subgraph isomorphism detection, in: Pro. ACM Workshop on Virtualized Infrastructure Systems and Architectures, 2009, pp. 81–88. doi:10.1145/1592648.1592662.

[52] P. Soto, J. F. Botero, Greedy randomized path-ranking virtual optical network embedding onto eon-based substrate networks, in: Proc. IEEE Colombian Conference on Communications and Computing (COLCOM), 2017, pp. 1–6. doi:10.1109/ColComCon.2017.8088197.

[53] M. Chowdhury, M. R. Rahman, R. Boutaba, Vineyard: Virtual network embedding algorithms with coordinated node and link mapping, IEEE/ACM Transactions on Networking 20 (1) (2012) 206–219. doi:10.1109/TNET.2011.2159308.

[54] R. Lin, S. Luo, J. Zhou, S. Wang, A. Cai, W. D. Zhong, M. Zukerman, Virtual network embedding with adaptive modulation in flexi-grid networks, Journal of Lightwave Technology PP (99) (2017) 1–1. doi:10.1109/JLT.2017.2764940.

[55] T. Truong-Huu, M. Gurusamy, Markov chain based algorithm for virtual network embedding in optical data centers, in: Proc. IEEE International Conference on High Performance Computing and Communications (HPCC), 2016, pp. 899–906. doi:10.1109/HPCC-SmartCity-DSS.2016.0129.

[56] X. Wang, M. Song, D. Yuan, X. Liu, Robust virtual network embedding based on component connectivity in large-scale network, China Communications 14 (10) (2017) 164–179. doi:10.1109/CC.2017.8107641.

[57] J. van de Belt, H. Ahmadi, L. E. Doyle, A dynamic embedding algorithm for wireless network virtualization, in: Proc. IEEE Vehicular Technology Conference (VTC-Fall), 2014, pp. 1–6. doi:10.1109/VTCFall.2014.6965811.

[58] L. Xu, G. Tan, X. Zhang, A cost sensitive approach for virtual network reconfiguration, in: Proc. IEEE Computing, Communications and IT Applications Conference (ComComAp), 2014, pp. 191–196. doi:10.1109/ComComAp.2014.7017194.

[59] Y. Zhu, M. Ammar, Algorithms for assigning substrate network resources to virtual network components, in: Proc. IEEE International Conference on Computer Communications (INFOCOM), 2006, pp. 1–12. doi:10.1109/INFOCOM.2006.322.

[60] G. D. Stasi, S. Avallone, R. Canonico, Virtual network embedding in wireless mesh networks through reconfiguration of channels, in: Proc. IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2013, pp. 537–544. doi:10.1109/WiMOB.2013.6673410.

[61] P. Han, L. Guo, Y. Liu, Green virtual network embedding framework based on zooming small cells in fiber-wireless access network for 5g, in: Proc. International Conference on Transparent Optical Networks (ICTON), 2017, pp. 1–4. doi:10.1109/ICTON.2017.8025010.

[62] G. Chochlidakis, V. Friderikos, Robust virtual network embedding for mobile networks, in: Proc. IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), 2015, pp. 1867–1871. doi:10.1109/PIMRC.2015.7343603.

[63] G. Chochlidakis, V. Friderikos, Mobility aware virtual network embedding, IEEE Transactions on Mobile Computing 16 (5) (2017) 1343–1356. doi:10.1109/TMC.2016.2591525.

[64] N. Meng, H. Zhang, H. Lu, Virtual cell-based mobility enhancement and performance evaluation in ultra-dense networks, in: Proc. IEEE Wireless Communications and Networking Conference, 2016, pp. 1–6. doi:10.1109/WCNC.2016.7564915.

[65] S. Avallone, An energy efficient channel assignment and routing algorithm for multi-radio wireless mesh networks, in: Proc. IEEE Global Telecommunications Conference (GLOBECOM), 2010, pp. 1–5. doi:10.1109/GLOCOM.2010.5684055.

[66] S. Avallone, G. D. Stasi, A. Kassler, A traffic-aware channel and rate reassignment algorithm for wireless mesh networks, IEEE Transactions on Mobile Computing 12 (7) (2013) 1335–1348. doi:10.1109/TMC.2012.107.

[67] M. Moser, Declarative scheduling for optimally graceful qos degradation, in: Proc. IEEE International Conference on Multimedia Computing and Systems, 1996, pp. 86–94. doi:10.1109/MMCS.1996.534959.

[68] M. Ozmen, M. C. Gursoy, Wireless throughput and energy efficiency with random arrivals and statistical queuing constraints, IEEE Transactions on Information Theory 62 (3) (2016) 1375–1395. doi:10.1109/TIT.2015.2510027.

[69] L. Tong, Y. Li, W. Gao, A hierarchical edge cloud architecture for mobile computing, in: Proc. IEEE International Conference on Computer Communications (INFOCOM), 2016, pp. 1–9. doi:10.1109/INFOCOM.2016.7524340.

[70] C. She, C. Yang, Ensuring the quality-of-service of tactile internet, in: Proc. IEEE Vehicular Technology Conference (VTC), 2016, pp. 1–5. doi:10.1109/VTCSpring.2016.7504239.

[71] M. Yu, Y. Yiz, J. Rexfordy, M. Chiang, Rethinking virtual network embedding: substrate support for path splitting and migration, in: Proc. ACM SIGCOMM, 2008, pp. 17–29. doi:10.1145/1355734.1355737.