Improving Integrated LTE-WiFi Network Performance with SDN based Flow Scheduling

Kang Chen[*], Jianwei Liu[†], James Martin[†], Kuang-Ching Wang[†], Hongxin Hu[†]
[*]Southern Illinois University, Carbondale, IL 62901 kchen@siu.edu
[†]Clemson University, Clemson, SC 29630 {jianwel, jmarty, kwang, hongxih}@clemson.edu

***Abstract*—Due to the explosive growth of data demand from mobile devices, cellular operators have been exploring the use of WiFi to offload traffic from the LTE network. Such an integration opens the door for exploiting the network usage diversity for further overall network performance improvement, by intelligently and dynamically scheduling flows over the most appropriate network. However, how such a function can be efficiently and systematically realize, is missing from the current standard specifications, especially on the network infrastructure side. In this paper, we aim to solve such a challenge by proposing a Software-Defined Networking (SDN) based flow scheduling system that is compatible to the 3GPP LTE-WiFi integration framework. The global view provided by SDN makes it easy to collect necessary flow information, and the flexible control of SDN enables efficient flow scheduling. We view the flow scheduling problem as an overall network utility maximization problem. We prove its hardness and propose an approximation algorithm for solving the problem. The proposed system can be incrementally deployed over existing wireless network infrastructure. With extensive simulations in NS3 and demo implementation, we prove the feasibility and effectiveness of both the framework and the scheduling algorithm.**

I. Introduction

Mobile traffic has been experiencing explosive growth in recent years. According to a Cisco report, the amount of mobile data traffic in 2019 will be 10 times larger than that in 2014 [1]. Meanwhile, the radio access in a single input and single output wireless communication system is reaching the limit of Shannon's law [2], leaving a limited space for further enhancement. To solve such a challenge, cellular operators have been seeking for small cell technologies to improve the overall network capacity in areas with heavy mobile traffic, e.g., shopping mall and university campus [2], [3]. As a result, integrating WiFi, which is the most widely deployed small cell access technology, into LTE networks has gained much interest from both industry and academia [2]–[6].

The 3GPP (3rd Generation Partnership Project) standard has already defined the framework for integrating WiFi into the LTE network [4], [7], as shown in Figure 1. In the framework, mobile traffic from both LTE and WiFi is converged at the PDN GW in the LTE EPC. The access network discovery and selection function (ANDSF) in the framework provides mobile devices with information about alternative wireless networks, including WiFi, WiMAX, and femtocells, and enforces policies for selecting and using those networks. Such an enhanced network access options can improve both the overall available network capacity and user experience. More importantly, this

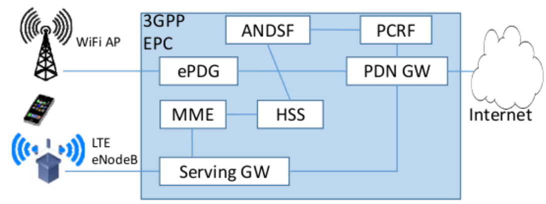


Fig. 1: Integration of WiFi into the LTE EPC.

enables the network owner to take advantage of the network usage diversity for improved overall network performance by monitoring and scheduling data flows in an area dynamically. Such diversity gain can be commonly obtained due to 1) uneven load distribution and 2) the differences between LTE and WiFi on MAC layer resource allocation (we explain those two reaons in details in Section III).

However, how the flow scheduling function can be systematically and incrementally enforced over current wireless infrastructure is not specified so far. On one hand, most existing methods focus more on algorithm and modeling development [3], [5], [8]–[16]. On the other hand, the ANDSF in 3GPP also lacks the ability to collect flow statistics for timely and efficient flow scheduling. It is more about determining and enforcing location- and device- specific network selection policy. Ideally, the new components to be developed for this purpose need to be compatible with and can be incrementally deployed over the 3GPP network framework shown in Figure 1. Thus, the proposed system can be easily and practically deployed for the most benefits.

Such a goal is not easy. First, manipulating flow paths inside cellular network backhaul is not a readily available feature. Second, doing this on mobile hosts cannot be enforced easily and thus is more challenging to operators. We thereby resort to the emerging Software-Defined Networking (SDN) [17] technique that abstracts the control over packet forwarding into a centralized controller. With such a structure, the network controller can easily access a global view of the flow status and issue commands for flexible flow management. The two features ideally fit the problem investigated in this paper. The global view supports effective network usage status collection, while the flexible management can be adopted to easily enforce flow scheduling decisions.

Therefore, in this paper, we propose to exploit SDN to realize an efficient and practical flow scheduling system for the integrated LTE-WiFi network. In detail, as shown in Figure 2,

978-1-5386-5156-8/18/$31.00 ©2018 IEEE

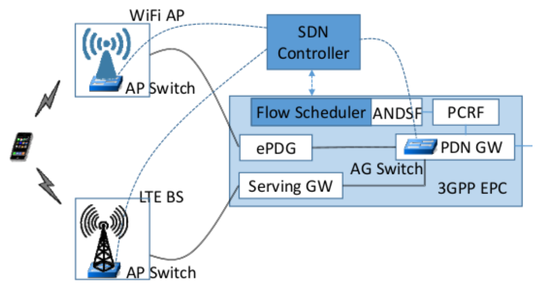


Fig. 2: System overview.

we integrate an SDN switch with each AP (defined AP Switch) and the PDN GW in the EPC (defined AG Switch). For easy description, we may use "AP" to represent an LTE BS. In the 3GPP standard [7], packets between APs, ePDG, Serving GW, and PDN GW are forwarded through the GTP tunnel. Thus, we require that SDN switches are integrated before the tunnel encapsulation in the AP and PDN GW. As a result, flows from and to mobile devices can be seen and monitored by those SDN switches. Note that though only two APs/BSs are shown in Figure 2, multiple APs/BSs can be included in the proposed system similarly.

All SDN switches are controlled by an SDN controller. A flow scheduler is designed inside the ANDSF. It communicates with the SDN controller through public northband APIs. We define uplink and downlink flows as flows coming out of and going back to the mobile device, respectively. By utilizing the global view of the controller, the scheduler can collect flow statistics effectively and make flow scheduling decisions accordingly. By exploiting the flexible control of the controller, the scheduler can easily update corresponding flow entries inside the AG Switch to enforce scheduling decisions for downlink flows. The scheduling decisions for uplink flows are enforced through the ANDSF. Consequently, flows are effectively scheduled over the most appropriate network that can lead to the overall utility maximization. In summary, the contributions of this paper include

(1) We propose to use the SDN to effectively support flow scheduling in the integrated LTE-WiFi network.
(2) We design a novel flow scheduling algorithm that can maximize the overall network utility with the information collected from SDN switches.
(3) Based on the above two components, we design an efficient flow scheduling system for the 3GPP integrated LTE-WiFi network framework.

The remainder of this paper is arranged as follows. Related work is described in Section II. Section III introduces preliminary design motivations. Section IV presents the system design. System efficiency and effectiveness are evaluated in Section V. Finally, Section VI concludes the paper.

II. Related Work

A. 3GPP Standard and Products

3GPP has been working on exploiting non-3GPP networks such as WiFi to complement the LTE network ever since the Release 8 [7], [18]. There are already commercial products developed following the 3GPP specification [19], [20]. These products are client software that enables a mobile device to discover and select the network following the policies defined locally or received from the ANDSF server. However, these products only support network discover and selection. They are not designed to create and enforce network selection policies dynamically so that mobile traffic is scheduled over different networks for overall optimization, which is a desirable feature for both mobile device users and network operators.

B. Network Selection and Flow Scheduling Algorithms

Network selection and flow scheduling in the heterogeneous network environment with multiple radio access technologies (RATs) have attracted much interest [3], [5], [8]–[16].

1) Device-side Algorithms: The work in [8] proposes a new metric that considers BS transmitted power, BS traffic load and user spectral efficiency for RAT selection in multi-RAT environment. As a result, high system and user performance is guaranteed without the need of handovers. Helou *et al* [5] further exploited network side information to enable more efficient RAT selection. A reinforcement learning approach is developed in this work to ensure the efficiency of network information acquisition. Both works in [9] and [10] model the RAT selection problem as a non-cooperative game and study its convergence, efficiency, and practical solutions. MOTA [11] provides a service model for UEs to exploit extra network signaling to collect necessary information for distributed operator and RAT selections. However, lacking a global view, these methods can only realize individual or partial global optimization on network usage.

2) Network-side Algorithms: Kosmides *et al* [12] defined different user satisfactory utility functions for different applications and formulate the network selection problem as an overall utility maximization process. Yeh *et al* [13] maximizes the on-time throughput in multi-radio heterogeneous networks by scheduling terminals following an on-time throughput based utility function. The work in [14] comprehensively investigates and summarizes key mathematical theories used for the RAT selection problem in literature, such as utility function theory, multiple attribute decision making, fuzzy logic, and game theory, etc. The work in [15] proposes a hierarchical resource management framework in which system functions are divided based on the time needed to complete them. The work in [16] proposes a conceptual framework to solve the network selection problem considering the network congestion, switching penalty, and network pricing.

ATOM [3] is a similar work with this paper. It builds proxies and agents on mobile devices and APs/BSs to collect network usage data, based on which flow scheduling is conducted for overall performance enhancement. Though we followed a similar way on wireless AP/flow throughput modeling, our major differences lie on the architecture choice. ATOM depends an additional component on mobile device to switch traffic over different paths, while our method offers this ability on the infrastructure side by novelly exploiting the features of SDN. ATOM also relies on LTE BS/WiFi AP to collect flow statistics, which may not be readily available at the moment. We think the infrastructure side solution is easier to be enforced by operators as it does not involve changes to end hosts. The SDN based scheme can be incrementally deployed in current wireless backhaul and synergizes with the emerging network function virtualization for cellular networks [21].

III. Preliminary Motivation

In an integrated LTE-WiFi network, when mobile devices choose the network individually without global optimization, the overall network performance can hardly be optimized.

We conducted field measurement in our lab to illustrate this point. The setup, as shown in Figure 3(a), includes one WiFi AP (TP Link WD3500), one indoor LTE BS (Lemko EZ LTE BS [22]), one server, and four UEs (laptops equipped with both WiFi and LTE interfaces). All UEs are covered by both the WiFi AP and the LTE BS. They downloaded a 1GB file from the server in all measurement.

A. Uneven Load Distribution

Mobile devices may all prefer one network (say WiFi) over the other network (say LTE), thereby causing one network to be overloaded and leaving the other network under-utilized. This would reduce the overall achieved throughput of the integrated LTE-WiFi network. We conducted field measurement to show this point. In this test, we purposely let all UEs have roughly the same WiFi/LTE signal strength, thus excluding its influence on achieved throughput.

We first let UEs in Figure 3(a) to access network with 80% probability on WiFi and 20% probability on LTE (denoted "UnevenDistribution"). We repeated this for 5 rounds and calculated the average value of the sum of all UEs' throughput. We then measured the same metric in an even load distribution scenario (denoted "EvenDistribution"): UE0 and UE1 use WiFi, while UE2 and UE3 use LTE. The final result is shown in Figure 3(b). We can clearly see that the uneven load distribution leads to a much lower achieved total throughput, because in this case, 1) the throughput of the LTE BS is not fully exploited (i.e., serving one UE on average) and 2) the WiFi experiences more contention.

Though individuals' network preferences can be balanced by adopting a more equalized pricing policy over LTE and WiFi, the load may still be imbalanced as UE density (and traffic load) may shift from time to time along with the mobility of UE owners. Therefore, unevenly distributed load would commonly exist in the integrated LTE-WiFi network.

B. Influences from MAC Layer Resource Allocation

WiFi and LTE employ different fairness strategies for MAC layer resource allocation. WiFi adopts packet level fairness, while LTE uses proportional fairness. The former would allocate more resources to the UE with a low link quality, thus degrading the AP's achievable throughput. Such a feature makes a WiFi AP's throughput very sensitive to the link qualities of UEs. The latter guarantees the resource share for each UE and thus is less susceptible to the issue in WiFi.

We verified this effect through field measurement too. We changed the placement of UEs in this test in which UE0/1 are close to WiFi AP and UE2/3 are close to LTE BS, as shown in Figure 3(c). Besides, UE0 (UE3) has a low link quality with the LTE BS (WiFi AP). We first measured a "proximity" scenario in which UE0 and UE1 connect to WiFi, and UE2 and UE3 to LTE. We then measured a "crossed" scenario in which the networks used by UE0 and UE3 are switched, making both WiFi AP and LTE BS to serve a remotely UE with a low link quality. Due to space limit, we only plot the "crossed" scenario in Figure 3(c). The measurement results are shown in Figure 3(d). We can see that when the remote UE joins, the WiFi AP suffers a significant overall throughput loss, while the LTE BS only presents a slight throughput reduction.

C. Summary

The first measurement validates the necessity of flow scheduling for load balance purpose. The second measurement shows that optimizing the performance of an integrated LTE-WiFi network is far more than load balancing. The throughput of a WiFi AP is not deterministic but rather dynamic upon associated UEs. LTE also has such a characteristic but in a smaller extent. Combing them together, we can see that dynamic flow scheduling needs to be conducted on the network operator side to improve the overall throughput in an integrated LTE-WiFi network, which motivates the design of this paper.

IV. System Design

A. System Overview

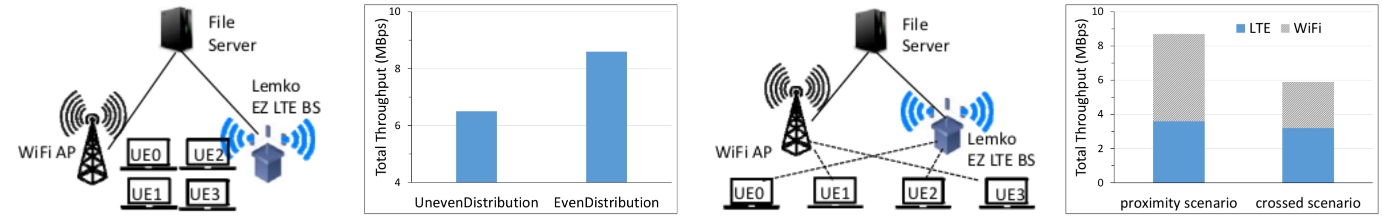
We assume an integrated LTE-WiFi network in this paper (Figure 1). We may use AP to represent an LTE BS for easy description purpose. However, we model and treat LTE BS and WiFi AP differently (Section IV-E1).

The proposed system consists of an SDN system and a flow scheduler, as shown in Figure 2. The SDN system is responsible for flow status monitoring and flow scheduling decision enforcement, which are introduced in Sections IV-C, IV-D and IV-F. The flow scheduler receives flow statistic data from the SDN system and outputs scheduling decisions that can maximize a defined utility function, the details of which are presented in Section IV-E.

The scalability of the proposed system is ensured by the fact that each deployment can independently serve a local area covered by one LTE BS and multiple WiFi APs. Thus, we can easily deploy multiple parallel systems to serve a large area. Our system works for both uplink and downlink flows, though we focus on the downlink in the following.

B. Seamless Flow Mobility

One prerequisite for effective flow scheduling is seamless mobility, i.e., moving a flow from one network to another network without disrupting the connection. Without such a function, the connection between the mobile device and the



(a) Default measurement setup with 4 UEs.

(b) Total throughput with even and uneven load distribution.

(c) Setup in the second measurement & UE association in "crossed" scenario.

(d) Total throughput in the "proximity" and "crossed" scenario.

Fig. 3: Illustration of the causes of degraded performance in integrated LTE-WiFi network.

corresponding host has to be reestablished after migrating to another network, which would greatly degrade both user experience and network usage efficiency.

This feature can be easily enabled in the integrated LTE-WiFi network by following the concept of home agent in IP flow mobility and seamless offloading (IFOM) [23]. We exploit the EPC and the logical interface (LIF) [24] on mobile device as the home agent to anchor all traffic from and to the mobile device, respectively, as shown in Figure 4. As a result, when a downlink/uplink flow is migrated, the application/the corresponding host cannot notice the change. Transparent multipath protocols such as HIP [25] and MPTCP [26] can also server this goal. We leave the study of those options to future work.

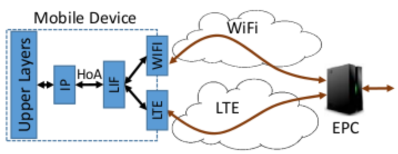


Fig. 4: Seamless flow mobility.

C. SDN System

We introduce how SDN switches and controller are organized to serve the flow scheduling in this section.

1) SDN Switches in APs (AP Switch): We integrate an SDN switch (AP Switch) into each LTE BS/WiFi AP to monitor uplink flows from mobile devices, as shown in Figure 5. Since LTE BS forwards packets to Serving GW through the GTP tunnel, the AP Switch needs to be placed before the tunnel encapsulation to access actual flows. This can be realized incrementally by installing a software SDN switch in the BS under the recent trend of BS virtualization [27].

We create a flow entry for each uplink flow to monitor it, when its first packet enters the switch. Each flow entry has a default TTL, say 3 seconds. A flow entry is deleted when its TTL expires as this means that the flow may be ended. Each flow has counters recording the time it has lived and the total bytes of data so far. The table in Figure 5 illustrates the flow entries in an AP Switch (suppose all uplink flows enter the AP Switch from port 1). Note that some metrics such as priority and TTL are omitted in the table.

2) SDN Switches before the EPC (AG Switch): We integrate an SDN switch into the PDN GW inside the LTE EPC to monitor and schedule downlink flows, denoted AG Switch. The AG Switch is placed before the encapsulation for the tunnel connecting to the Serving GW so that all downlink

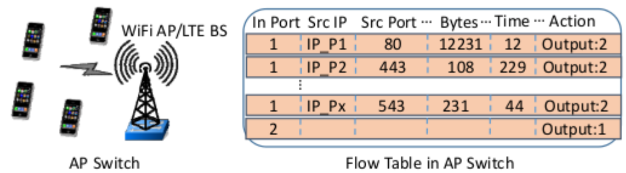


Fig. 5: The AP Switch and its flow entries (the last row represents the flow entry for all downlink flows).

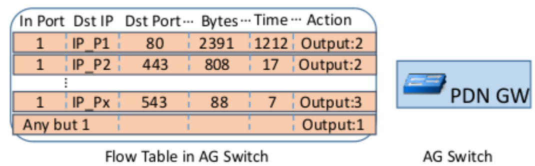


Fig. 6: The AG Switch and its flow entries (the last row represents the flow entry for all uplink flows).

flows can be seen. Such a requirement can be realized by installing a software SDN switch in the PDN GW server, especially in a virtualized EPC [27].

Again, similar to that used for AP Switch, we create a flow entry for each downlink flow in the AG Switch for monitoring purpose. We use one flow entry for all uplink flows as they are not monitored here (but in AP Switches). Figure 6 illustrates the flow entries in an AG Switch. These flow entries provide a convenient way to enforce the flow scheduling decisions for downlink flows. To change the network to be used to forward packets of a downlink flow to the mobile device, we only need to change the output port of its flow entry to the port that connecting to the corresponding AP/BS. The detail will be introduced in Section IV-F.

3) Controller: An SDN controller is designed in the system to control all SDN switches. The controller communicates with SDN switches and the flow scheduler through its southbound and northbound APIs, respectively. The controller collects flow statistics from SDN switches and reports that to the flow scheduler. The flow scheduler makes scheduling decisions and informs the controller (for downlink flows) and the ANDSF (for uplink flows) for enforcement.

4) Overhead of the SDN System: The AG Switch, the AP Switch, and the controller are standard SDN deployment on the infrastructure side, as shown in Figure 2, which holds abundant resources. Further, the number of SDN switches is the same as that of BSs/APs, which is quite small. Thus, we believe the extra overhead incurred by them is acceptable.

D. SDN-based Flow Monitoring

We exploit the SDN controller to collect flow statistics for scheduling. As shown in Figure 2, all uplink flows must pass through an AP Switch and all downlink flows must pass through the AG Switch. Therefore, we let the SDN controller periodically, i.e., every T_c seconds, pull all AP Switches and the AG Switch to collect the $\{Time, Byte\}$ of the flow entry for each uplink/downlink flow, as shown in Figures 5 and 6, which denote the time the flow has lived and the total amount of bytes it has transferred so far, respectively. The value of T_c can be tuned to balance the overhead and measurement accuracy. We set it empirically in the test.

Suppose the collected information of flow f_{ij} (i.e., node N_i's j-th flow) at the end of n-th and (n+1)-th interval are $\{Time^n_{ij}, Byte^n_{ij}\}$ and $\{Time^{n+1}_{ij}, Byte^{n+1}_{ij}\}$, respectively. Then, the size of flow f_{ij} in the (n+1)-th interval is

$$\tilde{s}^{n+1}_{ij} = \frac{Byte^{n+1}_{ij} - Byte^n_{ij}}{Time^{n+1}_{ij} - Time^n_{ij}} \tag{1}$$

The average size of flow f_{ij} at the end of the (n+1)-th interval is calculated as

$$s^{n+1}_{ij} = \sum_{i=n-q+2}^{i=n+1} \tilde{s}^i_{ij}/q \tag{2}$$

The above equation means that the average size of a flow (AS) is calculated as its average size in the most recent q intervals. q is set based on how significantly the flow size fluctuates.

By combining the pulled information of all flows, the flow scheduler can collect the following information.

(1) The average size of each uplink/downlink flow, i.e., the AS calculated by Equation (2).

(2) The average throughput of the uplink/downlink of an AP, which is the sum of the average sizes of all uplink/downlink flows running over the AP.

E. Flow Scheduler

The flow scheduler can be placed in the ANDSF in the LTE EPC, as shown in Figure 2, or a separate server in the backhaul. We exclude short flows for scheduling and leave them to default network of their devices. We again use **AP** to uniformly represent both WiFi AP and LTE BS in this section.

1) Flow Scheduling Modeling: The scheduling problem can be viewed as finding the association between flows and APs so that the overall network utility is maximized. Specifically, the objective of flow scheduling is to

$$\begin{aligned} \text{Maximize} \quad & \sum_i^N \sum_j^{F_i} \sum_k^{M_i} x_{ijk} * U(t_{ijk}) \\ \text{subject to} \quad & \sum_k^{M_i} x_{ijk} <= 1 \\ & x_{ijk} \in \{0, 1\} \end{aligned} \tag{3}$$

where N denotes the total amount of mobile devices, F_i denotes the number of flows of mobile device UE_i, M_i represents the indexes of APs that mobile device UE_i connects to, t_{ijk} is the achievable throughput of flow f_{ij} if it is associated to the k-th AP (i.e., AP_k), $U(t_{ijk})$ is the network utility contributed by flow f_{ij}, and x_{ijk} is a binary value represents whether flow f_{ij} is associated to AP_k. To solve such an optimization problem, we need to find the throughput of a flow when it is associated to different APs, i.e., t_{ijk}, and define the network utility appropriately.

Flow Throughput We calculate the achievable throughput of flow f_{ij} when it is scheduled on AP_k by

$$t_{ijk} = Th_{ik} * \frac{w_{ij} * s_{ij} * x_{ijk}}{\sum_{y \in F_i} (w_{iy} * s_{iy} * x_{iyk})} \tag{4}$$

where Th_{ik} is the physical layer rate that mobile device UE_i can achieve at AP_k when f_{ij} is scheduled on AP_k, w_{ij} denotes the weight of flow f_{ij}, s_{ij} denotes the average size of flow f_{ij}, and F_i denotes the number of flows on UE_i.

We consider two levels of resource allocation in this step. Firstly, we calculate the amount of data rate that UE_i can achieve at AP_k, i.e., Th_{ik}. APs in different networks adopt different MAC layer scheduling algorithms to allocate its wireless resources to associated mobile devices. For example, LTE BS adopts the proportional fairness algorithm, while WiFi AP usually uses the throughput fairness [3]. Thus, we calculate Th_{ik} separately in LTE and WiFi.

We follow [28], [29] to deduce the Th_{ik} in LTE. We use L_k to represent the number of active UEs associated with AP_k. We let $C_{ik} = f(SINR_{ik})$ denotes UE_i's maximal achievable rate at AP_k based on its signal-to-noise ratio to AP_k (i.e., $SINR_{ik}$). Generally, C_{ik} is achieved when AP_k only servers UE_i, i.e., when it gets all resources of AP_k. However, in reality, an LTE AP often is shared by all associated active UEs. With the proportional fairness, the resource on an LTE AP can be regarded as equally shared among associated active UEs in a long term [29]. This means that we can assume that UE_i gets $1/L_k$ of all resources at AP_k on average. Consequently, Th_{ik} can be calculated by

$$Th_{ik} = \frac{C_{ik}}{L_k} = \frac{f(SINR_{ik})}{L_k} \tag{5}$$

The MAC layer of WiFi tends to provide even transmission opportunity for all the UEs in the long term [30]. In another words, it implements *throughput fairness* [31]. For example, if two UEs (UE_1 and UE_2) have physical layer rates of 10Mbps and 20Mbps, respectively. Even though UE_1 has a lower rate, it will hold the channel to send the same amount of data as UE_2 when it wins the CSMA/CA competition. In a longer period, UE_1 gets 2/3 of air time, while UE_2 can only get $1/3$, making them achieve roughly the same throughput. Therefore, we deduce Th_{ik} in WiFi as the following,

$$Th_{ik} = \frac{1}{\sum_{i \in L_k} \frac{1}{r_{ik}}} \tag{6}$$

where r_{ik} denotes the maximal physical layer rate user i can reach with AP_k. It is also a function of $SINR_{ik}$.

Secondly, after obtaining resources from the AP, say AP_k, each mobile device, say UE_i, allocates it to its flows. We assume that a weighted FIFO queue is used to forward packets from different flows. Each flow is assigned a weight $w_{ij} \in [1, Q]$ denoting its importance, where Q represents the maximal weight value. Then, $w_{ij} * s_{ij}$ represents flow f_{ij}'s ability to compete for resources from the data rate UE_i obtains from AP_k. Thus, the second part of Equation (4), i.e., $\frac{w_{ij}*s_{ij}}{\sum_{y \in F_i}(w_{iy}*s_{iy}*x_{iyk})}$, represents the resource allocation at flow level on UE_i.

Network Utility Following the work in [3], we define the network utility for flow f_{ij} over AP_k as the following due to its wide adoption in wireless networks.

$$U(t_{ijk}) = w_{ij} * log(t_{ijk}) \tag{7}$$

where w_{ij} is the weight of the flow. We take the above definition as an example. Actually, different network utility functions that optimize other factors such as latency, energy, and billing can be designed similarly.

2) Flow Scheduling Solution: Solving the optimization problem in Equation (3) could find the optimal flow scheduling plan. The input to the algorithm the flow statistics, the t_{ijk} calculated in Equation (4), and available networks to each UE. The final output is a vector of x_{ijk} indicating the network selection for each flow. We use empirical measurement to get C_{ik} in different scenarios in the experiment (see Section V).

NP-Hardness However, even with all input, this problem is NP-hard. This is because, as indicated in Equation (4), when a flow migrates from one AP to another, it affects the resource allocation (and the utility values) of all flows on both the previous AP and the new AP. To prove the NP-hardness, we can reduce the complexity of the problem by assuming that each mobile device only has one flow. In this case, the second level of resource allocation in Equation (4) is removed. Then, this problem can be mapped to the problem of dividing a set of flows into two subsets of flows with equal total weights, which is a subset sum problem (which is NP hard). Therefore, the scheduling problem is NP-hard too.

Online Greedy Algorithm We then propose an online greedy algorithm, as shown in Algorithm 1, with two stages, i.e. the *transfer stage* and the *swapping stage*. The *transfer stage* first put all flows on LTE. It then sorts the flows based on their $SINR_{ik}$ to its best WiFi AP_k in decreasing order and tries a transfer operation to WiFi for every flow f_s sequentially. If the utility can be improved, f_s will be put onto WiFi. The rationale of this design is that putting a weakly connected UE to the WiFi AP would greatly degrade its maximal throughput, as explained in Section III-B. The *swapping stage* tries to switch the association of flows in order to reach a better configuration. The rationale is that as flows are handled sequentially, a later flow may fit the capacity of WiFi better together with other WiFi flows better than a flow already there. Then, swapping them can increase the overall utility. We propose a swapping operation that works as follows. The scheduler scans all remaining flows associated with LTE and tries to swap it with every flows in the WiFi AP that the UE of that flow is associated with.

Specifically, we first sort all flows in decreasing order of its UE's signal quality with the best AP (line 1) and then put them all on LTE (line 2-4). Then, we schedule UEs to WiFi one by one until the overall utility stops increasing (line 5-11). After we can no longer find an flow that scheduling it to the WiFi AP can lead to an utility increase, we turn to the *Swapping stage* (line 12). In this step, we iterates all possible scheduling arrangement for each UE's flows to find the best scheduling arrangement (line 12-19). The *swapping stage* actually only checks at the UE level, thereby greatly reducing the complexity. The overall time complexity is O($\bar{F}^2$), where $\bar{F}$ is the total number of flows, which is much smaller than the brute force algorithm.

Algorithm 1: Online Greedy Algorithm.

```
   // Sort flows based on its UE's SINR to the best WiFi AP
1  for ∀ m ∈ F̄ do
2  |   P :=P ← {L_m → LTE};
3  end
4  m = 0;
5  P* :=P ← {L_m → WiFi};
6  while U(P*) > U(P) do
7  |   m++;
8  |   P := P*;
9  |   P* :=P* ← {L_m → WiFi};
10 end
   // swap stage
11 for ∀ i ∈ N do
12 |   for ∀ configuration p_i of flows on UE_i do
13 |   |   P** :=P* ← p_i;
14 |   |   if U(P**) > U(P*) then
15 |   |   |   P*:=P**;
16 |   |   end
17 |   end
18 end
```

F. Enforce Scheduling Decisions

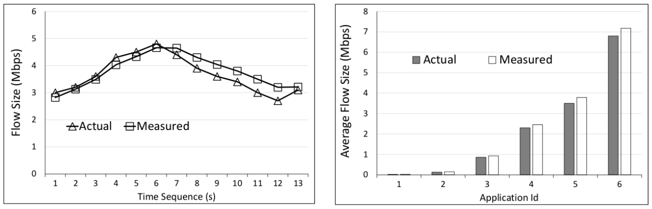
Flow scheduling decisions can be easily enforced in this system. For downlink flows, the flow scheduler notifies the controller the optimal flow association. Then, the controller updates the action of the corresponding flow entry in the AG Switch to forward packets to the new output port (Figure 6). For example, suppose the LTE BS and WiFi AP that a mobile device associates with connect to the AG Switch on port 2 and 3, respectively, which is recorded in a table in the controller. Then, to move a downlink flow from LTE to WiFi, we only need to change the output port of the flow entry from 2 to 3. The seamless flow mobility (Section IV-B) ensures that the flow will not be disconnected after the switch. For uplink flows, we let the LIF on mobile devices (shown in Figure IV-B) to receive commands from the ANDSF to enforce flow scheduling decisions similarly.

V. Performance Evaluation

Since the system proposed in this paper is for Integrated LTE-WiFi network, we name it as ILW-SDN in the test.

A. SDN System Effectiveness

We first tested the effectiveness of SDN system in collecting flow statistics and enforcing seamless flow migration with a small deployment in our lab, as shown in Figure 7. We



(a) One flow. (b) Multiple flows.

Fig. 8: Measuring flow sizes through SDN switches.

used one TP Link WiFi AP and one Lemko EZ LTE BS. Since we cannot modify the AP/BS, we connect each of them to a desktop simulating the AP Switch with OVS [32] installed. The traffic to and from the mobile device is anchored at an anchor point desktop, which performs the function of the AG Switch (through OVS) and the NAT. We use Floodlight [33] as the controller of those switches. In the test, we developed UDP based streaming applications on the application server located on our campus sending traffic to the testing mobile device (a laptop).

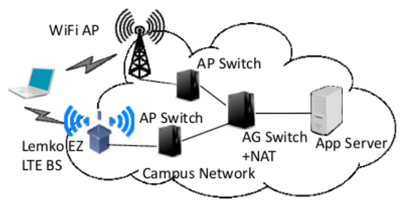


Fig. 7: Testing the SDN system.

We first only started one application (generating one flow) on the application server, and its data rate changes every 1 second. We then measured the size of the data flow periodically based on the statistics on the AG Switch, the result of which is shown Figure 8(a). We see that the measured flow size can effectively track the actual flow size. We then started 6 applications with different average data rates on the server ranging from 200Kbps to 6.8Mbps. Figure 8(b) shows the actual average size and the measured value for each application, which shows a small measurement error. Those results show that the size of each flow can be effectively measured through the flow statistics in the SDN switch.

We further tested the effectiveness of transparent flow migration. In this test, we only started one application on the application server echoing to the client. We migrated its traffic between the WiFi AP and the LTE BS after every 5 messages by changing the flow entry on the AG Switch. The round trip time of each message is measured and shown in Figure 9. We see that no message is dropped and the round trip time switches between around 3.5ms and 75ms (i.e. the delay through WiFi AP and LTE BS). This means that the flow is transparently migrated between the two networks without being interrupted through the proposed SDN system.

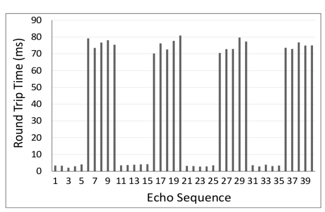


Fig. 9: Round trip time through different networks (i.e., APs).

B. Scheduling Performance

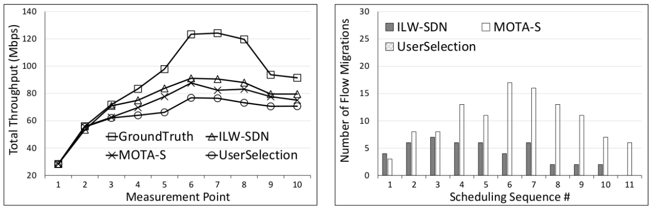
We use the NS3 [34] to evaluate the overall performance of the proposed system. We tested the integrated LTE-WiFi scenario with one LTE base station and 4 WiFi APs. WiFi APs are evenly distributed in the coverage of LTE, representing 4 sites with the WiFi coverage. Every WiFi AP has 5 UEs randomly placed under its coverage. This means that all UEs can access both LTE network and WiFi Network.

We developed applications on the application server to simulate mobile traffic for video, music, and file stream and downloading. The average data rate of each application is randomly selected between [500Kbps, 5Mbps], representing different video streaming qualities (e.g., 240P and 720P) and different file downloading services. We have also created some small and short flows as background traffic, which are not included in the scheduling following the design in Section IV-E. In order to reflect an intense network usage, we varied the number of flows running to each UE in the range of [3, 5]. Since mobile users may start and stop an application at any time, we set the start time of an application randomly in [0s, $SimTime/2$], where $SimTime = 50s$ denotes the simulation time. The live time of an application is randomly selected in [T_s, $SimTime - StartTime$], where T_s is the scheduling period and $StartTime$ denotes the start time of the application. We empirically set T_s to 4 seconds in the test. We measure throughput in the middle of two scheduling.

We compare ILW-SDN with two heuristic methods. The first one, denoted MOTA-S, is similar to MOTA [11] in which the base station shares its load to connected UEs for network selection. The probability of selecting an AP (LTE BS or WiFi AP) for a flow is reversely proportional to the load of the AP. The second one, denoted UserSelection, simulates the process that individual users select the network based on personal preferences. In this method, each UE has a randomly selected probability in the range of [0, 0.15] to select LTE for its flows. This is reasonable considering most people prefer the WiFi when it is available. For fairness consideration, the weight of each application's flow is set to 1 to show each method's ability to allocate wireless resources effectively.

1) Overall Scheduling Efficiency: We first measure the overall scheduling efficiency of different methods, which is shown by the system overall achieved throughput (i.e., sum of the overall achieved throughput of all UEs) and the number of resulted flow migrations in each flow scheduling. The former represents the effectiveness of resource allocation while the latter shows the efficiency of each algorithm. Note that since all flows have the same weight, the overall achieved utility generally is proportional to the achieved overall throughput. Therefore, we did not show it directly. The test results are shown in Figure 10(a) and Figure 10(b), respectively. In Figure 10(a), the "GroundTruth" represents the overall date rate measured from the applications that generate flows.

We see from Figure 10(a) that all methods lead to similar overall throughput as GroundTruth in the beginning. This is because the overall load is low in the beginning when most flows have not started. However, when more flows are generated and the overall load increases, ILW-SDN can always allow almost all flows to get transferred successfully, while the other two methods lead to lower overall throughput. This



(a) Overall throughput

(b) Number of flow migrations (UserSelection has no flow migration and thus is not shown)

Fig. 10: Overall scheduling performance.

is because when the overall load becomes large, ILW-SDN can dynamically schedule flows on different UEs based on their predicted throughput in different networks (Equations (5) and (6)), while the other two methods lack a global view to conduct flow scheduling. Specifically, for MOTA-s, each node selects a network for a flow based on the load information offered by APs/BS. Since there is no coordination among those APs/BS, users may all choose to avoid a congested one, making uneven load distribution. Therefore, it shows lower overall throughput than ILW-SDN. For UserSelection, the network selection is only based on user preference, which can easily cause a specific network (AP/BS) get overloaded, thereby leading to the least overall system throughput, i.e., LTE can easily get congested.

Figure 10(b) further shows the advantage of ILW-SDN. Note that UserSelection is excluded from discussion here since it does not schedule any flows and consequently has no flow migration. We can see that ILW-SDN shows much fewer flow migration than MOTA-S, especially in the middle part of the experiment. This is because the global view enables ILW-SDN to only migrate flows that can improve the overall network usage, while the AP/BS-feedback based scheme in MOTA makes user change networks frequently. All above results demonstrate that in general, ILW-SDN can effectively schedule flows over available wireless networks.

2) AP Load: We further measured the load distributed to each AP/BS to show in detail how the three scheduling methods work. Such load is measured from the application server side. The results are shown in Figure 11, in which the five vertical bars in each observation point represent the load on LTE and AP1 to AP4 sequentially from left to right.

Note that in the NS3 simulation, the capacity of the LTE BS is 17 Mbps, while that of a WiFi AP is between 20 Mbps and 24 Mbps (measured beforehand). We see from Figure 11(a) that with ILW-SDN, except the AP3, all other APs (including the LTE BS) rarely get overloaded, i.e., their load is always below the capacity. This is because ILW-SDN considers each AP's capacity in the flow scheduling. Further, when overloading an AP/BS is inevitable, ILW-SDN tries to overload as few as possible. We can see that only AP3 is overloaded in ILW-SDN, which actually is because that there are too many large flows for UEs associated under AP3.

On the other hand, MOTA-S and UserSelection cannot avoid overloading APs/BS. We see that in MOTA-S, several APs in MOTA-S get repetitively overloaded (in every the other observation point). This reflects the drawback of MOTA-S. In detail, UEs avoid one AP/BS when it is overloaded, making the AP/BS have a low load in the next period. However, after one more period, UEs would switch back since it appears to be under-loaded, thus making it overloaded again. For UserSelection, since all nodes prefer WiFi to LTE, several WiFi APs get overloaded from time to time. The above results show that ILW-SDN can reasonably distribute the load to APs/BS to improve the overall system throughput.

3) UE Throughput: We further pick two representing UEs and show their throughput in Figure 12(a) and Figure 12(b), respectively. We found that both UEs' flows get transferred smoothly with ILW-SDN and get congested from time to time in Random and Even. Such results further justify ILW-SDN's ability to exploit multiple wireless networks to ensure the bandwidth requirement of individual UEs.

4) Scheduling Cost and Complexity: The cost of ILW-SDN comes from three parts: acquire necessary information, make scheduling decisions, and push scheduling decisions to UEs.

For the first part, ILW-SDN needs to periodically query AG Switches (as shown in Figure 2) for the statistics of ongoing flows. We argue that such a cost is acceptable since 1) the information is queried not very frequently, 2) the information of a flow often is smaller than 100 bytes, and 3) both the scheduler and the AG switches are in the backhaul that can provide abundant network resources.

For the second part, as mentioned in Section IV-E2, the flow scheduling problem is NP hard. However, we have proposed an online greedy solution with complexity $O(\bar{F}^2)$, where $\bar{F}$ denote the total number of flows on all UEs. We further found that the scheduling process in the simulation can be done in less than 0.1 seconds on a normal laptop. Moreover, as mentioned in Section IV-A, each ILW-SDN system works for a local area with one LTE BS and multiple WiFi APs. Thus, we conclude that the computation complexity of ILW-SDN is acceptable for real deployment.

For the third part, the number of flow migrations in our test is shown in Figure 10(b). We see that ILW-SDN has a small number of flow migrations (the total number of flows > 50). This means that ILW-SDN only migrates about 1/9 of flows in each round for the most. Since the size of a flow scheduling command is about several bytes, such a cost is acceptable.

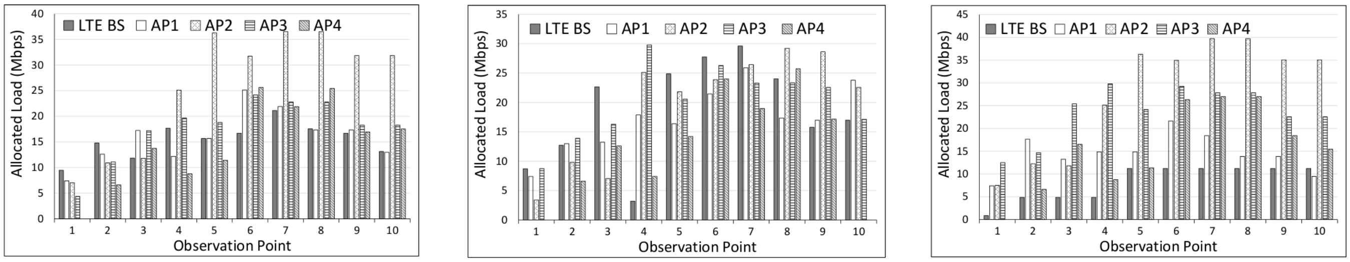
Combining above results, we conclude that the extra cost of ILW-SDN is acceptable for real deployment.

Acknowledgements

This research was supported in part by Clemson Post-doctoral Research fund, U.S. NSF grants 1531127, and SIU startup fund.

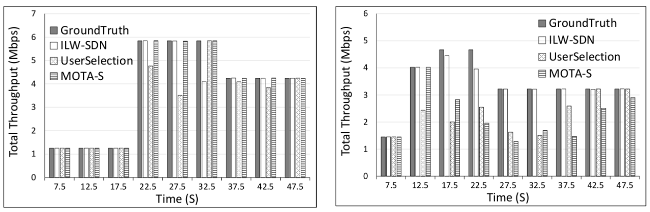
VI. Conclusion

In this paper, we exploit SDN to design an effective flow scheduling system for integrated LTE-WiFi network. The system has two major parts: an SDN system and a flow scheduler. The former provides flow statistics to the flow scheduler and helps enforce flow scheduling decisions.



(a) ILW-SDN (b) MOTA-S (c) UserSelection

Fig. 11: Total size of flows allocated to each LTE BS/WiFi AP.



(a) UE 1 (b) UE 19

Fig. 12: Throughput of individual UEs.

It installs an SDN switch after each BS/AP and before the EPC. By exploiting the global view provided by SDN, the controller can easily acquire statistics about ongoing flows. Such information is fed to the flow scheduler, in which an efficient flow scheduling algorithm is developed to select the most suitable network for each flow. By exploiting the flexible control of SDN, flow scheduling decisions can be enforced efficiently. Extensive experiments illustrate that the proposed system can efficiently allocate network resources to flows for overall network usage optimization. In the future, we plan to deploy the system on our campus at a larger scale.

REFERENCES

[1] Cisco, "VNI mobile forcast 2014-2019," http://www.cisco.com/assets/sol/sp/vni/forecast_highlights_mobile/index.html.

[2] "Architecture for mobile data offload over wi-fi access networks," http://www.cisco.com/c/en/us/solutions/collateral/service-provider/service-provider-wi-fi/white_paper_c11-701018.html.

[3] R. Mahindra, H. Viswanathan, K. Sundaresan, M. Y. Arslan, and S. Rangarajan, "A practical traffic management system for integrated LTE-WiFi networks," in *Proc. of MobiCom*, 2014.

[4] "WLAN traffic offload in LTE," http://www.rohde-schwarz.us/en/applications/wlan-traffic-offload-in-lte-application-note_56280-30866.html.

[5] M. El Helou, M. Ibrahim, S. Lahoud, K. Khawam, D. Mezher, and B. Cousin, "A network-assisted approach for rat selection in heterogeneous cellular networks," *JSAC*, 2015.

[6] O. Galinina, A. Pyattaev, S. Andreev, M. Dohler, and Y. Koucheryavy, "5g multi-rat lte-wifi ultra-dense small cells: Performance dynamics, architecture, and trends," *JSAC*, 2015.

[7] "3GPP," http://www.3gpp.org/.

[8] A. Orsino, G. Araniti, A. Molinaro, and A. Iera, "Effective RAT selection approach for 5G dense wireless networks," in *Proc. of VTC*, 2015.

[9] E. Aryafar, A. Keshavarz-Haddad, M. Wang, and M. Chiang, "RAT selection games in HetNets," in *Proc. of INFOCOM*, 2013.

[10] I. Malanchini, M. Cesana, and N. Gatti, "Network selection and resource allocation games for wireless access networks," *IEEE Transactions on Mobile Computing*, 2013.

[11] S. Deb, K. Nagaraj, and V. Srinivasan, "MOTA: Engineering an operator agnostic mobile service," in *Proc. of MobiCom*, 2011.

[12] P. Kosmides, A. Rouskas, and M. Anagnostou, "Utility-based RAT selection optimization in heterogeneous wireless networks," *Pervasive and Mobile Computing*, 2014.

[13] S.-p. Yeh, A. Y. Panah, N. Himayat, and S. Talwar, "Qos aware scheduling and cross-radio coordination in multi-radio heterogeneous networks," in *Proc. of VTC*, 2013.

[14] L. Wang and G.-S. Kuo, "Mathematical modeling for network selection in heterogeneous wireless networksa tutorial," *Communications Surveys & Tutorials, IEEE*, 2013.

[15] A. Zakrzewska, A. P. Avramova, H. Christiansen, Y. Yan, A. Checko, A. Dogadaev, S. Ruepp, M. S. Berger, and L. Dittmann, "A framework for joint optical-wireless resource management in multi-rat, heterogeneous mobile networks," in *Proc. of ICC*, 2013.

[16] M. H. Cheung, R. Southwell, and J. Huang, "Congestion-aware network selection and data offloading," in *Proc. of CISS*, March 2014, pp. 1–6.

[17] N. Feamster, J. Rexford, and E. Zegura, "The road to SDN: an intellectual history of programmable networks," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 2, pp. 87–98, 2014.

[18] "Access network discovery and selection function (ANDSF) management object (mo)," in *3GPP TS 24.312 Rel 12 Version 12.9.0*.

[19] "Smart switch," http://www.roke.co.uk/mobile/smartswitch.html.

[20] "Qualcomm connectivity engine," https://www.qualcomm.com/media/documents/files/3g-lte-wifi-offload-framework.pdf.

[21] S. Sun, M. Kadoch, L. Gong, and B. Rong, "Integrating network function virtualization with sdr and sdn for 4g/5g networks," *IEEE Network*, vol. 29, no. 3, pp. 54–59, 2015.

[22] "Lemko ez lte product," https://www.lemkocorp.com/solutions/.

[23] C. Sankaran, "Data offloading techniques in 3GPP Rel-10 networks: A tutorial," *Communications Magazine, IEEE*, 2012.

[24] A. De la Oliva, C. J. Bernardos, M. Calderon, T. Melia, and J. C. Zuniga, "Ip flow mobility: smart traffic offload for future wireless networks," *IEEE Communications Magazine*, vol. 49, no. 10, 2011.

[25] R. Moskowitz, P. Nikander, P. Jokela, and T. Henderson, "Host identity protocol," Tech. Rep., 2008.

[26] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, "TCP extensions for multipath operation with multiple addresses," March 2011.

[27] F. Granelli, A. A. Gebremariam, M. Usman, F. Cugini, V. Stamati, M. Alitska, and P. Chatzimisios, "Software defined and virtualized wireless access in future wireless networks: scenarios and standards," *Communications Magazine, IEEE*, vol. 53, no. 6, pp. 26–34, 2015.

[28] T. Bu, L. Li, and R. Ramjee, "Generalized proportional fair scheduling in third generation wireless data networks," in *Proc. of INFOCOM*, 2006.

[29] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. Andrews, "User association for load balancing in heterogeneous cellular networks," *IEEE TWC*, vol. 12, no. 6, pp. 2706–2716, June 2013.

[30] E. Garcia, D. Viamonte, R. Vidal, and J. Paradells, "Achievable bandwidth estimation for stations in multi-rate ieee 802.11 wlan cells," in *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, June 2007, pp. 1–8.

[31] W. Wang, X. Liu, J. Vicente, and P. Mohapatra, "Integration gain of heterogeneous wifi/wimax networks," *Mobile Computing, IEEE Transactions on*, vol. 10, no. 8, pp. 1131–1143, Aug 2011.

[32] "Open vSwitch," https://openvswitch.org/.

[33] "Floodlight," http://www.projectfloodlight.org/floodlight/.

[34] "NS3," https://www.nsnam.org/.