A Practical Traffic Management System for Integrated LTE-WiFi Networks

Rajesh Mahindra
NEC Laboratories America, Inc.
rajesh@nec-labs.com

Hari Viswanathan
Rutgers University
vhariharasudhan@gmail.com

Karthik Sundaresan
NEC Laboratories America, Inc.
karthiks@nec-labs.com

Mustafa Y. Arslan
NEC Laboratories America, Inc.
marslan@nec-labs.com

Sampath Rangarajan
NEC Laboratories America, Inc.
sampath@nec-labs.com

Abstract

Mobile operators are leveraging WiFi to relieve the pressure posed on their networks by the surging bandwidth demand of applications. However, operators often lack intelligent mechanisms to control the way users access their WiFi networks. This lack of sophisticated control creates poor network utilization, which in turn degrades the quality of experience (QoE). To meet user traffic demands, it is evident that operators need solutions that optimally balance user traffic across cellular and WiFi networks. Motivated by the lack of practical solutions in this space, we design and implement `ATOM`- an end-to-end system for adaptive traffic offloading for WiFi-LTE deployments. `ATOM` has two novel components: (i) A network interface selection algorithm that maps user traffic across WiFi and LTE to optimize user QoE and (ii) an interface switching service that seamlessly re-directs ongoing user sessions in a cost-effective and standards-compatible manner. Our evaluations on a real LTE-WiFi testbed using YouTube traffic reveals that `ATOM` reduces video stalls by 3-4 times compared to naive solutions.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless Communication*

Keywords

Cellular Networks; WiFi offload; Traffic Management

1. INTRODUCTION

Cellular networks are facing an unprecedented increase in data traffic due to the popularity of bandwidth-intensive mobile services. Although operators are continuously upgrading their networks to cope with such increase, the growth in network capacity is still considerably behind the bandwidth demand [1]. Hence, most operators around the world (e.g., China Mobile, AT&T) are aggressively deploying WLANs for additional capacity since WiFi is cheap and easy to deploy at scale [2, 3, 4]. However, sustaining good QoE in such heterogeneous deployments requires a much more sophisticated solution than simply deploying unmanaged WLANs. For next-generation mobile networks, a solution that carefully manages the network interface (e.g., WiFi vs. LTE) of user flows forms a critical component of network optimization. Although such solutions exist today, they suffer from the following limitations:

Drawbacks of Current Solutions: (i) *Naive, static and coarse-grained policies*: Operators rely on connection managers on user devices that are generally configured to select WiFi as the default interface when available [5]. Since WiFi APs are usually deployed in hot-spot areas to begin with, one can expect a large number of users to receive a strong signal from WiFi APs during peak periods. Hence such naive policies do not translate to higher user throughput, since the load of the WiFi AP is not accounted for in interface selection. In addition, most operators do not have the capability to switch the interface of a flow seamlessly (i.e., without breaking) across WiFi and cellular; the interface selection is thus decided only when initiating the connection. Hence, the selection is not adaptive to the dynamic conditions of wireless networks. Finally, the same level of throughput translates to different levels of QoE for a user depending on the application. Hence, loading all the application flows [6, 7] of a user on to the same interface does not translate to improved QoE for all the flows as the capacity of that interface has to be shared by multiple such flows from other users as well. (ii) *Lack of practical solutions:* While some studies [7, 8, 9, 10] have focused on interface selection, they only solve a part of the problem by simply providing algorithms for interface assignment assuming that a framework for seamless switching exists. In addition to the theoretical complexity of the problem, designing such a framework alone has several practical constraints and challenges that it must account for to deliver a readily deployable solution. While there exist some systems efforts that schedule user data across WiFi and cellular interfaces [11, 12], they are limited to delay-tolerant traffic and cannot support real-time applications such as video.

Challenges: (i) *Practicality:* The framework must be light-weight, scalable and designed as an overlay solution over current LTE networks without requiring additional standards support. (ii) *Adaptiveness:* To sustain high QoE, the system must dynamically choose interfaces in order to adapt to flow arrivals, departures and changing wireless link conditions. (iii) *Seamlessness:* In the event of an interface (and thus IP address) change during an ongoing user session, the framework should not break the existing connections and should seamlessly migrate user flows between WiFi and LTE. (iv) *Business interests:* Seamless flow migration currently requires that all WiFi traffic gets backhauled to the LTE core network for proper IP anchoring, thereby significantly increasing the operational costs. Thus, it is challenging to provide a dynamic solution given the lack

MobiCom'14, September 7-11, 2014, Maui, Hawaii, USA.

http://dx.doi.org/10.1145/2639108.2639120.

of incentive for operators to invest heavily on user QoE for OTT (over-the-top) traffic. Thus, *the key challenge is to not just design a scalable, dynamic and seamless traffic management solution, but to also build an end-end system that can be easily deployed in any operator's core network (i.e. being operator agnostic) without requiring tight data plane integration between WiFi and LTE.*

To address these challenges, we design `ATOM`- a system that adaptively maps user flows to the appropriate network interface to improve user QoE. `ATOM` has two key components: (i) A fine-grained traffic management solution that uses a practical algorithm for interface selection to maximize the network-wide utility. (ii) A switching service that seamlessly changes the interface for certain user flows *without the need for data plane integration*, thereby reducing backhaul costs for the operators. We observe that certain characteristics of HTTP video streaming and browsing can be exploited to enable seamless re-direction of such flows, via HTTP proxies, to avoid backhauling these traffic types from WiFi to the LTE network. However, `ATOM`'s formulation is not limited to HTTP and also supports non-HTTP flows (although such flows would not benefit from the backhaul reduction). Nevertheless, we believe that `ATOM` offers important backhaul cost savings since most of the traffic in mobile networks is video streaming over HTTP [13].

We have prototyped and evaluated `ATOM` on a heterogeneous LTE-WiFi testbed using real Web traffic. Our evaluations show that `ATOM` effectively reduces the video buffering periods for a user from an average of 8 to 2 per minute. We also evaluated the seamless interface switching functionality of `ATOM` with several Web video services. To the best of our knowledge, this is the first detailed design and implementation of a practical system that manages user traffic across LTE and WiFi networks. A noteworthy aspect of `ATOM` is that it is operator-agnostic and standards-compatible, and can hence be readily deployed for any operator looking to manage its LTE and WiFi networks efficiently. Our contributions are multi-fold: (i) We establish the hardness of the interface assignment problem and propose a greedy algorithm with performance guarantees under certain conditions. Our algorithm is scalable and practical to implement. (ii) We design and build an end-to-end dynamic traffic management system that seamlessly switches the interface for user flows and (iii) we conduct extensive evaluations using both prototype experiments and large-scale simulations.

2. BACKGROUND AND MOTIVATION

In this section, we give a brief overview of the LTE network architecture, then expand on the evolution of the integration of WiFi with LTE and motivate the need for an effective traffic management solution for LTE and WiFi networks.

2.1 LTE Networks

The top-half of Figure 1 shows a simplified 4G LTE network architecture, mainly consisting of two parts: the Evolved Packet Core (EPC) Network and the Radio Access Network (RAN). The EPC or the mobile core network consists of both the control and data plane functions. The control plane functionality is provided by the MME (Mobility Management Entity), HSS (Home Subscriber Server) and the PCRF (Policy and charging rules function). The MME handles session and subscriber management including user authentication, mobility management and idle terminal location management. The HSS includes a database that stores the user profile information while the PCRF manages the service policy and configures the QoS parameters for each user traffic flow. The data plane functionality in the EPC is split between the S-GW (Serving gateway) and the PDN-GW (Packet Data Network gateway). The S-GW acts as a local mobility anchor for user sessions as clients

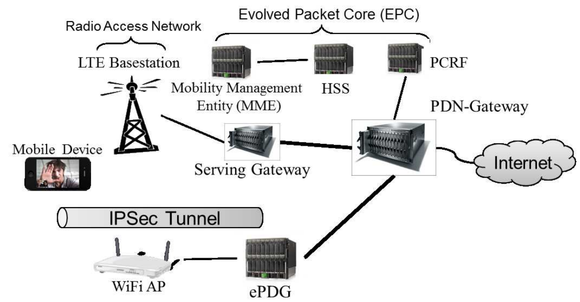


Figure 1: LTE Network Architecture.

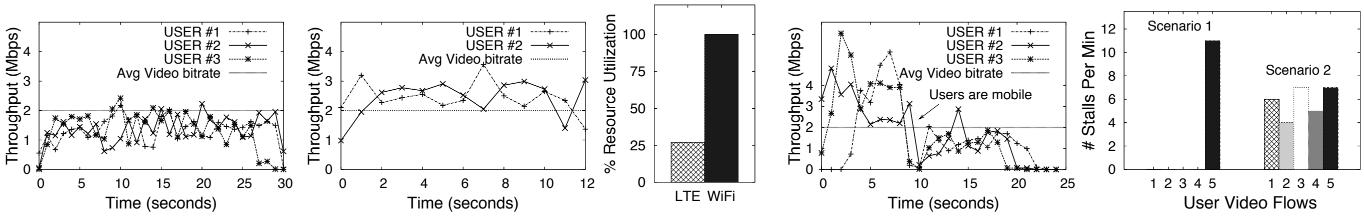
move across base stations. The PDN-GW is connected to multiple S-GWs and routes user traffic towards external networks, while also performing policy enforcement for resource management, packet filtering and charging functions. The RAN includes basestations (or eNodeBs) that perform radio resource management and interference mitigation.

2.2 WiFi Integration in Operator Networks

Several standard bodies such as 3GPP and WiFi Alliance (WFA) have defined solutions for the network integration of LTE and WiFi. These solutions are mainly classified into two types: (i) *Access control*: To enable subscriber validation, seamless authentication and billing across LTE and WiFi networks is an important step to ensure WiFi integration. However, the method of authentication varies across operators. Most of the operators provide SIM-based authentication [14] enabling them to maintain a unified subscriber database for both their LTE and WiFi networks, while other operators have adopted the traditional web-based authentication which requires the users to enter their credentials in the browser. (ii) *Data-plane integration*: To enable offloading capabilities and seamless mobility between LTE and WiFi networks, a tight data-plane integration is required across the networks. Such integration involves the backhauling of WiFi traffic to the LTE core network. Specifically, 3GPP has standardized the I-WLAN architecture [15] to integrate WiFi traffic into LTE's mobile core network. The architecture as shown in Figure 1 enables the integration using the ePDG (evolved Packet Data Gateway), which serves as a gateway connecting the WiFi access points with the PDN gateway. IPsec tunnels are established between each mobile device and the ePDG, and the IP address is anchored at the PDN gateway. Since the IP address is maintained across the WiFi and LTE networks, flows can be seamlessly migrated across the networks. The PMIPv6 protocol is employed and the ePDG updates the IP address binding at the PDN gateway after authentication and tunnel establishment with the mobile device. Although tight integration will enable operators to ensure policy control, better QoE management and seamless mobility over their networks, it has been resisted by most operators due to the significant increase in backhauling costs.

2.3 Current Deployments

In the near future, it is expected that operators will transition to using their WiFi networks for new services and revenue generation and provide better QoE for their users rather than just offloading for coverage or during congestion. Moreover, operators are quickly upgrading their network to LTE that offer superior rates than 3G networks and are deploying WiFi APs in areas of high network access. However, current deployments are not designed to use the LTE and WiFi network optimally to ensure good QoE for applications and users. Although most devices are pre-configured with connection managers, they mainly implement functions for network discovery, selection and authentication.



(a)WiFi (Static) (b) LTE (c) Resource Util (d) WiFi (Dynamic) (e) Course-grained Policies

Figure 2: Limitations of Current deployments.

Short-comings: We bring to light a few key issues with current deployments through experiments on our LTE testbed and address them in the design of ATOM. The experiments are conducted using a network of a single LTE basestation and a WiFi AP.

(i) Naive policies: Most connection managers [5] are configured with simple policies that ensure the device connects to a WiFi AP in case a connection is made. A few connection managers use the WiFi interface only if the signal strength is above some threshold. However since they do not take the current load on the AP into account, the QoE of the users could suffer during congestion. To drive our point, we setup an experiment such that 6 users are randomly distributed and are within the coverage of the WiFi AP, while 2 users are outside the coverage of the WiFi AP. All the 8 users stream videos from YouTube with an average bit-rate of about 2 Mbps. We plot the throughput obtained by 3 out of the 6 WiFi users and the 2 LTE users in Figures 2(a) and (b) respectively. We see that the throughput of WiFi users is less than the average bit-rate (2Mbps) of the video resulting in stalls in the video stream while the throughput of LTE users is above the average bit-rate resulting in a smooth stream. Figure 2(c) depicts the resource utilization: while the WiFi AP is over-utilized, the utilization of the LTE basestation is only 25%.

(ii) Static decisions: Moreover, it is not sufficient to make interface selection decision at the initiation of a user flow as wireless conditions change significantly due to user arrival/departure and mobility. To drive our point, we use a similar setup with 4 users on the WiFi AP. As shown in Figure 2(d), initially all the WiFi users receive throughput in excess of the video bit-rate. At around 10 seconds, we move a couple of the WiFi users away from the AP at walking speeds. As a result of the user mobility, the WiFi AP is unable to support the video rates of its users as shown in Figure 2(d) affecting the video of the users mid-stream. However, to enable dynamic traffic management, operators are required to poses the capability to switch the interface of user flows seamlessly across their LTE and WiFi networks. Such a capability needs tight dataplace integration of the WiFi network with the LTE network. While the integration of access control (authentication) methods for WiFi have been widely adopted by operators [4], tighter integration of data or bearer plane to the LTE network has been resisted by most operators, mainly due to: (1) Backhauling large amounts of WiFi traffic through their LTE core network significantly increases both Operational costs (OP-EX) in terms of backhaul costs and Capital costs (CAP-EX) in order to scale their LTE core gateways. (2) Most of the traffic and services on mobile networks is OTT (Over-the-top) that does not generate direct revenue for the operators. Hence there is little incentive for operators to invest significantly in order to provide QoE for such services. (3) In most scenarios, we discovered that the WiFi business units of operators are managed independently from the LTE business.

(iii) Coarse-grained policies[6, 7]: Operators will desire the ability to perform interface selection on a per-application level rather than a per-user or per-device level. This capability ensures (a) operators can provide QoE depending upon the application requirements and (b) content providers may be willing to pay mobile operators for better QoE for users accessing their applications in the future. Operators will need to differentiate the performance of such flows over other OTT traffic. We conduct an experiment to show the disadvantage of the inability to perform fine-grained traffic management. The experiment is setup with 8 LTE users within the coverage of the WiFi AP and 4 LTE users outside the WiFi coverage. All the 8 users download a large file from the WiFi AP. One of the WiFi users (User#5) also streams a YouTube video of average rate 2Mbps. All the 4 LTE users stream the same YouTube video from the LTE basestation. Figure 2(e) plots the average number of stalls in the video session of the 4 LTE users and User#5. Scenario 1 represents the case where all the traffic of User#5 is mapped to the WiFi AP since the user is within the coverage of the AP. Clearly, the video flow of User#5 suffers significantly as the WiFi AP is congested. Scenario 2 represents the case with user-level traffic management where both the flows of User#5 (video and file-download) are moved to the LTE network. This results in the LTE network getting congested and the video of all the 5 users suffer. A fine-grained traffic management solution would move the video flow of User#5 to LTE while keeping the file-download flow on the WiFi AP, resulting in good performance for the video of all the 5 users.

3. ATOM DESIGN

To address the afore-mentioned drawbacks, we propose ATOM, an end-to-end traffic management system that enables operators to flexibly and efficiently manage user traffic flows across a heterogeneous network of LTE and WiFi APs. Before describing ATOM in detail, we explain our key design considerations:

(i) *Network-centric*: ATOM is designed as a centralized solution that leverages a complete network view (of cell load, user QoE etc.) to determine the optimal interface selection. This gives ATOM an important advantage over client-based distributed solutions (e.g., [8]), which require proprietary signaling from the network (creating wireless link overhead and requiring change in standards) to obtain load information easily accessible by ATOM. ATOM is a gateway-level solution that can be deployed within the LTE mobile core network as opposed to being deployed within each LTE eNodeB. This design has the following benefits: (a) Deep packet inspection (DPI) modules, policy engines etc. are already present within the LTE EPC and ATOM interfaces with these gateways to acquire appropriate information. Hence it is better to co-locate ATOM with them. (b) Deploying ATOM in each basestation hinders deployability as it increases the basestations computational requirement.

(ii) *Scalability*: While maintaining a centralized view of the network, ATOM is carefully designed to scale to large deployments by treating each LTE cell (and the WiFi APs in its coverage) in isolation when deciding the network interface for users.Since traffic from an LTE cell is offloaded to the WiFi APs that are within its

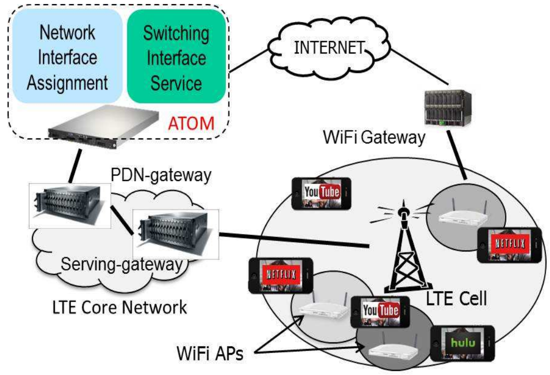


Figure 3: ATOM's Architecture.

coverage, multiple ATOM instances run in parallel where each instance independently manages the user flows in a given LTE cell. This design choice ensures that ATOM can be scaled-up as the operator deploys more LTE basestations and/or WiFi access points as opposed to traditional middle boxes that need to be scaled-out. This design also ensures further performance optimizations, wherein ATOM instances are instantiated only for congested cells.

(iii) *Seamless Switching*: ATOM incorporates a mechanism to seamlessly switch user traffic flows across LTE and WiFi interfaces. This mechanism is designed such that (1) it is cost effective (2) it enables adaptive or dynamic traffic management given that wireless conditions change over time. Moreover, it is hard to estimate or define application requirements on the onset of a TCP connection from a mobile device. Hence, ATOM sets simple static policies on the mobile device, for instance all connections are initiated over the WiFi interface. With seamless switching, the appropriate decision for an application flow can be taken while the session is ongoing. This ensures operators can define more sophisticated policies, for instance based on resolution of video selected by the user.

(iv) *Pricing*: Users are generally charged based on two plans: (i) Price per byte: This is a fixed amount per KB of data (ii) Tiered data-caps: Users have a data limit (e.g., 3GB per month) paying a fixed monthly price. Moreover, operators offer WiFi free of cost to their current customers. However, such a model may change as operators offer similar carrier-grade service on their WiFi networks as they do on their LTE networks. With this in mind, we design ATOM to incorporate general pricing mechanisms to ensure that its design is applicable to either of the above pricing scenarios.

Considering the above, ATOM is instantiated as a gateway-level solution in the operator's access network external to the basestations as shown in Figure 3. Since the gateway will typically handle traffic for multiple basestations, it hosts multiple ATOM instances, each handling traffic for one LTE basestation. ATOM's design has two components: (i) Network Interface Assignment (NIA) component and (ii) the Interface Switching Service (ISS).

4. NETWORK INTERFACE ASSIGNMENT

This is the component that manages all user flows that belong to a given LTE cell. Specifically, it takes as input the signal strength of every user to its potential set of WiFi APs and the LTE basestation, relative QoS priority (or weights) and the current network interface of each user flow. It then computes the appropriate network interface (i.e., a specific WiFi AP or the LTE basestation) for each user flow. In this section, we formulate the network assignment as a utility optimization problem with a per-flow utility function to ensure differentiated QoS across applications.

Network Model: ATOM operates at the level of a LTE cell where one or more WiFi APs are deployed within the coverage of that cell as in Figure 3. ATOM also handles scenarios where the coverage of several WiFi APs overlap resulting in certain users having the option to connect to multiple WiFi APs. Hence, NIA computes the specific WiFi AP or LTE basestation that is used by each user flow. Since NIA operates at coarse time-scales (T is in orders of seconds), it leaves the fine-grained packet scheduling function to be performed by the LTE basestation and the WiFi APs locally. To allow this decoupling, the throughput is modeled as the average throughput of the client over the time T based on the scheduling policy. The problem can be formulated as:

$$x^* = \arg\max_x \quad \sum_{j=0}^{B}\sum_{i=1}^{N} x_{ji} U(t_{ji}) \qquad (1)$$

$$\text{subject to} \quad \sum_{j=0}^{B} x_{ji} = 1$$

where B is the total number of WiFi APs within the coverage of the LTE basestation (represented by $j = 0$). The indicator variable $x = \{x_{ji}, \forall j\}$ denotes the association vector for user flows i.e., $x_{ji} = 1$ if flow i is assigned AP j. t_{ji} is the average throughput estimated for flow i when associated with the AP j. The constraint ensures that exactly one WiFi AP or LTE basestation is chosen for a user flow. Different flows of a user are allowed to pick potentially different WiFi APs. In practice, this can be realized using the virtualization capability found in most WiFi cards to create virtual WiFi networks that can run on a single WiFi physical interface [16]. The challenge in solving the above optimization lies in the utility (and throughput) function that couples the decisions of user flows assigned to the same interface.

Throughput and Fairness Models: LTE and WiFi have different MAC protocols with potentially distinct fairness (bandwidth sharing) policies that directly affects the throughput of the user flows.

LTE eNodeBs typically employ proportional fair scheduling. They also schedule the resources to the user flows in proportion to a weight that defines the relative priorities of the flows. In this case, the throughput of a user can be shown to depend on the total number of the other users and their relative weights as follows.

$$t_{ji} = \frac{w_i \times r_{ij}}{\sum_{i \in N_j} w_i} \qquad \forall i \in N_j \qquad (2)$$

where w_i is the weight for user flow i; r_{ij} is the average link-layer rate (or the PHY rate) of user i on AP j (the eNodeB in this case) depending on the average signal-to-noise ratio (SNR) of the user on that AP and N_j is the total number of active users on AP j.

On the other hand, WLANs when operated distributively, typically use a throughput-based fairness model. Here, all the users served by the same AP get the same throughput at steady state. This is because the APs implement a round-robin scheduling scheme for the downlink packets. In this case, the average downlink throughput of a WiFi user can be expressed as:

$$t_{ji} = \frac{L}{\sum_{i \in N_j} \frac{w_i L}{r_{ij}}} \qquad \forall i \in N_j \qquad (3)$$

where L is the average size of a packet in bits.

However, when the operator controls both the LTE and WiFi networks, then it is possible to instrument a uniform fairness policy (say proportional fairness) across both these networks. In this case, the throughput of WiFi users would follow a throughput model similar to that for LTE. Also, we assume that interference between neighboring LTE cells and WiFi APs is taken care of through their respective interference management algorithms (frequency reuse in

LTE and channel selection in WiFi), so as to not affect the throughput models.

Choice of Utility Function: While the design of NIA would work with concave utility functions in general, we incorporate the logarithm function as the utility function for all user flows. Although applications might have diverse requirements for QoE and operators will want to provide differentiated services to different application flows, NIA employs the log utility function generally to all application flows since: (a) It ensures a simple system design (b) recent advancements in end-to-end adaptation by application flows (for instance adaptive video streaming [17]) allows modeling most application traffic as elastic. Such a function ensures that the marginal utility of a flow decreases as the throughput increases. (c) log functions are extensively used as the utility function for resource management in wireless networks [18, 19]. Hence, NIA defines the utility function for every user flow as the product of the weight of the flow and the log of the average throughput obtained by the flow. Operators can set the weights of the flows accordingly to differentiate among applications and/or users.

$$U(t_i) = w_i \times \log(t_i) \tag{4}$$

Pricing Model: The notion of pricing the different interfaces based on their consumption can be easily incorporated in our utility framework. The utility of an interface assignment for a user flow i can be updated as $(U(t_{ij}) - E_{ij})$, where E_{ij} is the associated cost for flow i using the interface j and is defined based on the pricing model of the operator: (i) *Pricing per byte*: E_{ij} can be made to capture consumption in the current epoch as $E_{ij} = C_j w_i$ [8], where C_j is the cost per unit weight of the flow. Since the actual flow throughput in an epoch depends on multiple factors, the cost is typically based on the weights [8], which influences how throughput is shared. (ii) *Tiered Data-caps*: On the other hand, E_{ij} can capture data usage till the previous epoch as $E_{ij} = C_j \frac{D_{kj}}{n_k}$. C_j would now be the cost per unit KB of data (given by dividing the data cap of the user by the monthly cost of the plan), D_k is the total data usage till the previous epoch by user k on network j, and n_k is the total number of flows at user k, thereby splitting the cost of a user equally across all its flows. Hence, the associated cost of an interface E_{ij} is higher for the flows of the users with higher data usage in the past on that interface. Instead of a linear function, one could also consider other functions of data usage. Note that the pricing is mainly used to serve as a deterrent in picking an interface. By appearing as a constant in a given epoch, it does not directly influence the per-epoch optimization problem.

4.1 Problem Hardness

Considering even the simplest topology with one LTE eNodeB and one WiFi AP, the complexity for solving the problem grows exponentially with the number of user flows. Intuitively, the problem is hard because the correct choice of a network interface for a given user flow depends on the exact combination of other user flows assigned to the APs. Specifically for WiFi, the throughput of a user flow depends on the PHY rates of the other users attached to the AP (throughput fairness) and in the case of LTE, the throughput of a user flow depends on the weights of the other users attached to the eNodeB (see Equations (2) and (3)). The proof that Problem 1 is NP-Hard for a network of an LTE basestation and a WiFi AP is deferred to the Appendix. The complexity of the problem further increases when considering multiple WiFi APs within a LTE cell, especially the case where some of the APs may have overlapping coverage. Note that Problem 1 is NP Hard even for the case where the WiFi APs employ Proportional Fair scheduling. However, for a certain case when the user weights are unity and both LTE and WiFi perform proportional fairness scheduling, the problem is optimally solvable (proof similar to the load balancing problem in [20]). But this case is not applicable to ATOM, since ATOM is designed to provide differentiated QoS to applications and users.

Algorithm 1 NIA Algorithm

1: INPUT: $\forall i \in \mathcal{N}$: # of Active user flows, $\forall j \in \mathcal{B}$: # of WiFi APs,
   $\mathcal{S}_0$: Set of Active flows not within the coverage of WiFi,
   $\mathcal{S}_j$ Set of Active flows within the coverage of AP j.
2: OUTPUT: User flow Association $\mathcal{A}_j, \forall j \in \mathcal{B}$
3: $\pi \leftarrow \{B\}, \mathcal{A}_0 \leftarrow \{\mathcal{S}_0\}$
4: $\mathcal{L}_j \leftarrow \{\mathcal{S}_j\}, \forall j \in \mathcal{B}$
5: % Outer Loop
6: **for** $x \in [1 : |\mathcal{B}|]$ **do**
7:   % Inner Loop
8:   **for** $j \in \pi$ **do**
9:     $\mathcal{A}_j = \emptyset$
10:     $\mathcal{A}_{0j} = \mathcal{A}_0 \cup \mathcal{L}_j$
11:     **for** $i \in \mathcal{L}_j$ **do**
12:       $i^* = \arg\max_{(i)s.t.\ i \notin \mathcal{A}_j} \{\sum_{k \in \mathcal{A}_j \cup i} U(t_{jk}) + \sum_{k \in \mathcal{A}_{0j} - i} U(t_{0k}) - \sum_{k \in \mathcal{A}_j} U(t_{jk}) - \sum_{k \in \mathcal{A}_{0j}} U(t_{0k})\}$
13:       $\mathcal{A}_j \leftarrow \mathcal{A}_j \cup i^*$
14:       $\mathcal{A}_{0j} \leftarrow \mathcal{A}_{0j} - i^*$
15:     **end for**
16:     $U_j = \sum_{i \in \mathcal{A}_j} U(t_{ji}) + \sum_{i \in \mathcal{A}_0} U(t_{0i})$
17:   **end for**
18:   $b \leftarrow \arg\max_j U_j$
19:   $\pi \leftarrow \pi - b$
20:   $\mathcal{L}_j \leftarrow \mathcal{S}_j - \mathcal{A}_b, \forall j \in \mathcal{B}, j \neq b$
21:   $\mathcal{A}_0 \leftarrow \mathcal{A}_{0b}$
22: **end for**

4.2 Algorithm

NIA employs a practical yet efficient greedy algorithm. The algorithm executes in two steps as shown in Algorithm 1. It takes as input the number of active user flows N, the number of WiFi APs B within the coverage of the LTE cell, the subset of active user flows S_j that are within the coverage of the WiFi AP j and the subset S_0 that includes flows which are not in the coverage of any WiFi AP. Note that the sets S_j may not be independent since some users may be covered by more than one WiFi AP. Initially all active user flows that are not within the coverage of a WiFi AP are assigned to the LTE eNodeB (i.e., $A_0 \leftarrow S_0$). π represents the set of all the WiFi APs whose users have not been assigned an interface yet and L_j represents the set of user flows that belong to a WiFi AP's coverage but have not been assigned an interface yet (i.e., $L_j \subseteq S_j$). The final solution is given by the subsets $\mathcal{A}_0$ and $\mathcal{A}_j$ that consist of the flows that are assigned to the LTE cell and WiFi AP j respectively.

In the outer loop at every step, NIA considers each WiFi AP $\in \pi$ in isolation. It finds the best combination of user flows across the LTE cell and a particular WiFi AP. It then finalizes the interface assignment for all the user flows of that WiFi AP, which yields the highest utility among all the WiFi APs that are part of the set π (step 18). Having fixed the interface assignment for user flows of a WiFi AP in a single round, the initial condition is reset with this assignment. Specifically, the WiFi AP for which the interface assignment is finalized is removed from the set π (Step 19). The user flows that are assigned to an interface are removed from the set L_j of the other WiFi APs (Step 20) that also cover these flows so that they are not considered in the following rounds. The user flows assigned to the LTE basestation are added to the set A_0 (Step 21). The steps are repeated for each of the remaining WiFi APs until the user flows of all WiFi APs have been assigned an interface. As discussed above, since the assignment of user flows to a LTE basestation and a single WiFi AP is also computationally complex, NIA employs a greedy algorithm to compute the assignment in the inner loop (steps 8-17).

The inner loop performs the assignment of user flows for each pair of WiFi AP (whose flows have not been assigned an interface yet) and the LTE basestation. Initially, no user flows are assigned to the WiFi AP (Step 9). The assignment for the LTE basestation (Step 10) is initialized with the user flows that are already assigned to LTE (A_0) and the unassigned user flows that are within the coverage of the WiFi AP j (L_j). Starting with these initial assignments, NIA moves user flows one by one from the LTE basestation to the WiFi AP such that the incremental utility is maximized. For each user flow, the incremental utility is the difference in utility with the current assignment (i.e. LTE) and the interface assignment with the user flow moved to the WiFi AP (Step 12). NIA stops moving flows from the basestation to the WiFi AP when none of the remaining flows result in a positive increase in the marginal utility. After this step, NIA commits the utility for the particular WiFi AP as shown in step 16.

Performance Guarantee: Given the complexity of the problem in the general case, it is hard to claim a worst-case guarantee for our algorithm. However, extensive evaluations show that average-case performance is convincing. The algorithm runs in $\mathcal{O}(B^2N)$ where B is the number of WiFi APs and N is the number of flows.

5. INTERFACE SWITCHING SERVICE

The goal of the ISS framework is to provide a service to the NIA to enable dynamically switching the interface of user flows to ensure effective traffic management. Every T seconds, based on the decisions made by the NIA, the ISS switches the network for the appropriate user flows. The fundamental problem in providing seamless connectivity across networks is maintaining the end-to-end TCP connection since the IP address of the user changes. While standards bodies such as 3GPP adopt the approach of maintaining the same IP address by anchoring all the traffic through a common gateway, ISS takes a different, yet seamless and low-overhead approach for HTTP-based traffic flows based on two key observations: (i) Mobile operators are resisting tight integration of the data planes of their LTE and WiFi networks to avoid significant increase in backhauling costs for the WiFi traffic (as discussed in Section 2). (ii) HTTP is the dominant mobile protocol (over 90% traffic carried over HTTP [21]). More importantly, HTTP-based video traffic accounts for more than 60% of the total bytes carried on mobile networks and is expected to increase to more than 75% [22, 13]. Although UDP protocol is more suited for video streaming, HTTP/TCP protocol has been employed widely to leverage existing benefits of HTTP, namely caching, CDNs, traversal through NAT, content naming etc. Keeping the above mentioned observations in mind, ISS intelligently leverages certain characteristics of HTTP-based video streaming and web-browsing (discussed below) to design a switching service that switches network interface of flows without anchoring the connection through a single gateway, thereby avoiding backhauling of WiFi traffic through the LTE core network. Please note that although ISS takes a different approach from the 3GPP standard based approaches like I-WLAN, it is a complementary solution and can be deployed as an overlay over existing I-WLAN deployments. This ensures that backhauling can be avoided for atleast HTTP-based flows using ISS, while remaining flows are backhauled using I-WLAN to ensure dynamic interface switching. Given that bulk of the internet traffic is HTTP based including video flows, ISS provides significant cost savings for the operators by avoiding backhauling HTTP flows.

Quick Primer on HTTP: Traditionally HTTP-based videos used to be treated as file downloads. However, with recent advancements, two popular schemes have emerged: (i) HTTP progressive download (PD): In this scheme, video players typically request the video in byte ranges instead of downloading the whole file. HTTP-PD was introduced for video pacing i.e., the client requests chunks of videos at a download rate that matches the playing rate and avoids wasting bandwidth in case the user quits the player before the video ends. HTTP-PD also allows users to seek to a later point in the video. (ii) Dynamically adaptive streaming over HTTP (DASH): The design of DASH [17] is aligned with HTTP-PD, however it allows the player to request different encoded versions of the video ensuring adaptability to network conditions. The original video is encoded into multiple bit-rates and divided into segments or chunks that typically contain 4-10 seconds of video. First, the player downloads a file containing the URL for each chunk for every encoded version of the video. The player sends HTTP-GET requests to the server to download the chunk of the appropriate bit-rate according to measured TCP throughput. Similarly, browsing traffic typically consists of several relatively small sized objects (e.g., html, images etc.) and each object is requested by an individual HTTP-GET request.

Leveraging HTTP: The ISS framework leverages the above characteristics of HTTP-based video streaming and browsing wherein the content within a session is downloaded using multiple HTTP GET requests over time. Specifically, when the interface or network of these flows have to be switched, subsequent HTTP-GET requests of these flows can be performed over the new interface. Although, typically HTTP-GET requests are multiplexed over existing TCP connections, sending HTTP requests over multiple TCP connections in parallel is supported by HTTP. Hence, the subsequent HTTP-GET requests are made over one or more TCP connections that are set up over the new interface or network. Although this applies only to HTTP-based video streaming and browsing flows, these traffic flows do not need to be backhauled to the LTE network, thereby saving significant costs for the operator. This is especially important, given that video traffic accounts for a sizeable portion of the total bytes carried by mobile networks and web traffic is the most popular traffic type.

5.1 Design of ISS

ISS is designed using HTTP proxies in the LTE network and a HTTP proxy at the mobile device to enable seamless interface switching on existing mobile networks as shown in Figure 4. The applications and the browser on the mobile device are configured to use the *HTTP proxy* on the device, which ensures that the HTTP requests are sent over the appropriate interface. In other words, all HTTP traffic generated from the device is routed through the HTTP proxy on the device. The HTTP proxy is a light-weight user-space program that is capable of proxying the HTTP request from the application or the browser to either the network proxy or directly to the content servers. The HTTP proxy listens for commands to switch interfaces from the *Control Logic* on the device. Similarly, on the LTE network-side the ISS framework consists of a HTTP Proxy and a Control Logic. The Control Logic exposes an interface for the NIA to send commands for switching the network interfaces of user flows based on the output of the algorithm. The network-side Control Logic maintains a persistent TCP connection with the Control Logic on every device through the LTE network to relay the commands from the NIA to the appropriate devices as shown in Figure 4. The HTTP Proxy within the LTE network is employed for HTTP traffic that excludes video streaming and browsing to ensure seamless switching for other types of traffic. Most mobile operators already deploy HTTP proxies for optimizations and caching purposes. ISS switches the user flows based on the traffic type:

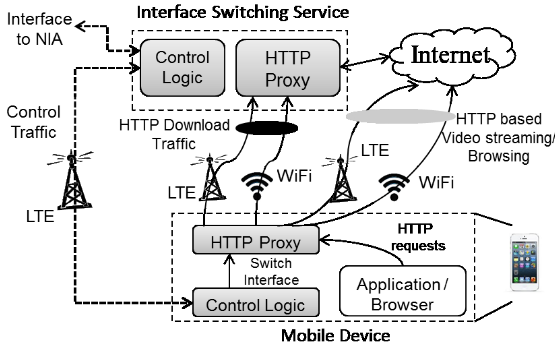


Figure 4: Interface Switching Service.

1. HTTP-based downloads: These flows include downloading of medium to large files (e.g., Dropbox), software updates etc. To ensure that such flows are seamlessly switched between LTE and WiFi, they are always routed through the in-network HTTP proxy as shown in Figure 4. Routing HTTP traffic through the same proxy for both LTE and WiFi ensures that the flow is anchored at a single server and hence the interface switching at the device is transparent to the content servers. Specifically upon instruction to switch interfaces from the ISS, the HTTP proxy on the device sets up a TCP connection with the in-network proxy using the new interface. The in-network proxy tears down the TCP connection over the current interface before sending data over the new TCP connection to keep the HTTP session alive.

2. HTTP-based video streaming and browsing: Unlike the previous traffic type, these flows are not routed through the in-network HTTP proxy as shown in Figure 4. After receiving a command from the control logic to switch the network interface for a specific web session, the HTTP proxy on the device simply requests the subsequent objects from the new interface, while continuing to receive existing objects from the current interface. In a similar fashion, for video flows, the HTTP proxy on the device simply requests the subsequent video chunks from the new interface.

By leveraging HTTP proxies, ATOM realizes a seamless interface switching service that can be readily deployed. Further, with video traffic not requiring an in-network HTTP proxy, ATOM avoids backhauling the bulk of the traffic (being video) from the core network. Although, we can avoid backhauling of HTTP-based download flows using HTTP byte-range manipulation at the proxy on the device, we avoid HTTP header modifications to ensure a simple design and proper operation across different platforms and applications.

3. Non HTTP-traffic: ATOM resorts to using standards-based approaches such as I-WLAN to support seamless switching for non-HTTP traffic. Hence, these flows will need to be backhauled through the mobile core network.

6. PROTOTYPE

Our test-bed consists of a LTE basestation (or eNodeB), openEPC software EPC [23], Madwifi-based WiFi APs and Linux laptops as clients with both Verizon Pantech LTE dongles [24] and Broadcom WiFi cards (see Figure 5). The eNodeB is a 3GPP Release 9 compliant LTE small cell on the 700 MHz band. Considerable effort, involving code modifications to the openEPC components, was spent to integrate the eNodeB (closed-source) with the EPC to ensure interoperability with commercially available LTE clients (closed-source). Our EPC network [23] consists of MME, HSS, PCRF for control plane and S-GW and PDN-GW for data plane functions. In addition, the Internet gateway provides connectivity to the Internet and includes key functions such as NAT and DNS.

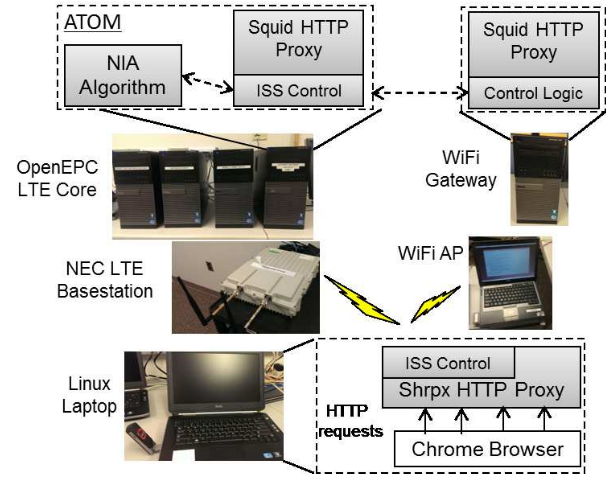


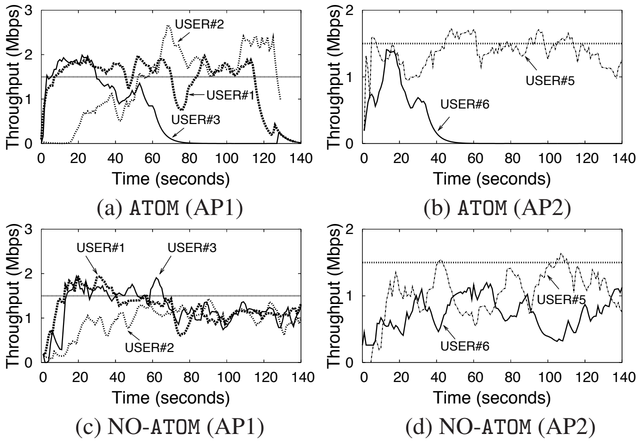
Figure 5: ATOM Prototype.

The LTE clients are Pantech USB dongles with USIM cards programmed with the appropriate identification name and secret code to connect with the eNodeB. Since the eNodeB and the clients communicate on Verizon's licensed band, we use custom built frequency converters. These convert the frequency in both downlink and uplink from 700 MHz to 2.6 GHz, where we have an experimental license to conduct over the air experiments.

Network: We implemented ATOM on the Internet gateway that connects directly to the PDN-gateway. NIA is implemented within the Click modular router using C++, while the ISS-control is a standalone C++ application. NIA periodically gathers the following information from ISS (a) number of user flows active on LTE basestation and WiFi APs, (b) current interface used by each flow (c) weights of application flows and (d) link-layer or PHY rate of each flow. A control logic component is also implemented within the WiFi gateway that provides information about active user flows over the WiFi APs and the link-layer rate of each WiFi user (collected from the APs) to the ISS-control. Once NIA has all the information in an epoch, it executes the algorithm to assign the network interface to each flow and sends a message to the ISS-control with information about all user flows that need to be switched to a new interface. The prototype also includes two Squid [25] HTTP proxies in the network side for both LTE and WiFi networks.

Client Device: We implement ISS-control within the Shrpx based HTTP proxy module [26] that runs as a user-space process. The Chrome browser is configured using the PAC (Proxy Auto Configuration) file to use the Shrpx proxy as the default proxy for all applications. Hence, all HTTP requests from Chrome are made to the Shrpx proxy. Initially when the device comes online, the ISS-control establishes a persistent TCP connection and registers using a unique ID with the ISS-control on the network side. When a new application flow is initiated, the Shrpx proxy always connects using the WiFi network if available. We now explain the steps involved when the interface of a particular flow is moved from WiFi to LTE.

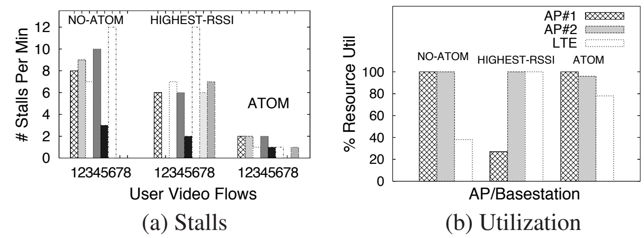
In the case of *HTTP-based download* flows, the following steps are involved: (a) Upon the initiation of a new connection from the browser, the shrpx proxy initiates a TCP connection to the Squid proxy on the LTE network through the WiFi interface.(b) Upon receiving a command from the ISS-control to switch the network interface from WiFi to LTE, the Shrpx proxy establishes a new TCP connection with the same Squid Proxy through the LTE network. (c) The Squid proxy then terminates the previous TCP connections over WiFi, before sending HTTP data over the LTE network to ensure seamless continuity of the HTTP session.



(a) ATOM (AP1) (b) ATOM (AP2)

(c) NO-ATOM (AP1) (d) NO-ATOM (AP2)

Figure 6: ATOM ensures good throughput.



(a) Stalls (b) Utilization

Figure 7: ATOM ensures high QoE and network util.

In the case of *HTTP-based video streaming or browsing* flows, the following steps are involved: (a) Upon the initiation of a new connection from the browser, the shrpx proxy initiates a TCP connection to the Squid proxy on the WiFi network through the WiFi interface avoiding backhauling traffic through the LTE network. (b) Upon receiving a command from the ISS-control to switch the network interface from WiFi to LTE, the Shrpx proxy establishes a new TCP connection with a Squid Proxy on the LTE network through the LTE interface. (c) The Shrpx proxy then forwards all HTTP requests for subsequent objects from the browser to the Squid prpxy through the LTE network. (d) The Shrpx proxy breaks the TCP connections with the Squid proxy over the WiFi network after all the pending HTTP requests have been downloaded. For both traffic types, the same procedure is repeated when a connection has to be switched to WiFi from LTE.

Currently, we employ SPDY [27] as the protocol between Shrpx and the Squid proxies. Although the network (both LTE and WiFi) proxies are not required for HTTP-based video streaming or Web browsing traffic types, they are employed in our prototype since Shrpx is currently not designed to connect to multiple servers. Since most Web servers require multiple simultaneous TCP connections, Shrpx is configured to relay the HTTP requests to the respective Squid proxy based on the current network interface used by the device. An important aspect of our implementation is that it can be readily deployed by instrumenting existing mobile protocols and is completely standards compatible.

In summary, ATOM executes as follows: (i) User flows always initiate the connection from WiFi if available and register with the ISS in the network; (ii) NIA executes periodically to select interfaces for all active user flows and (iii) NIA sends a command to the ISS including user flows with new interface information.

7. PERFORMANCE EVALUATION

In this section, we demonstrate the efficacy of ATOM using experiments on our prototype and large-scale simulations.

7.1 Prototype Evaluation

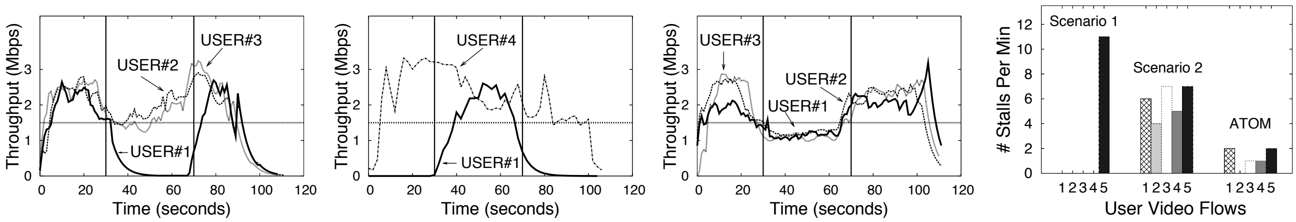
We consider two types of workloads including video streaming from YouTube and HTTP based file downloads. We evaluate ATOM using metrics such as throughput and number of stalls due to buffering in the video streams (interested readers can see our prototype demo in [28]). NO-ATOM represents the baseline that maps user flows to WiFi APs if the user is within the coverage of a WiFi AP.

1) Static Experiment: We setup a network of one LTE eNodeB and two WiFi APs with a total of 11 users; 5 users are within the coverage of WiFi AP1, 4 users are within the coverage of WiFi AP2 and 2 users can only access the eNodeB. AP1 is placed close to the eNodeB while AP2 is placed further from the eNodeB and all the users are distributed randomly. In WiFi AP1, 4 users steam YouTube videos of average bit-rate 1.5 Mbps while 1 user downloads a large-file. In WiFi AP2, 2 users stream YouTube videos of average bit-rate 1.5 Mbps while 2 users download large-files. Both the LTE users stream YouTube videos of average bit-rate 1.5 Mbps. We compare the throughput of WiFi users for the case with and without ATOM (NO-ATOM).

Figure 6(c) and (d) plot the throughput for 3 video streaming flows on WiFi AP1 and 2 video streaming flows on WiFi AP2. Clearly, the throughput cannot be sustained to meet the average bit-rate of the video since both the APs are congested. On the other hand, ATOM ensures that flows of Users#3 and 4 from WiFi AP1 and User#6 from WiFi AP2 are moved to the eNodeB to ensure that the throughput received by all users meets their requirement of 1.5 Mbps. Figures 6(a) and (b) plot the throughput received by the users on the WiFi AP1 and 2 respectively. Hence, by effectively distributing user flows across LTE and WiFi APs, ATOM decreases the number of stalls from an average of $8 - 10$ stalls per minute with NO-ATOM to at most $1 - 2$ stalls per minute (Figure 7(a)). ATOM also improves the resource utilization of the eNodeB from 40% to almost 80% as seen in Figure 7(b).

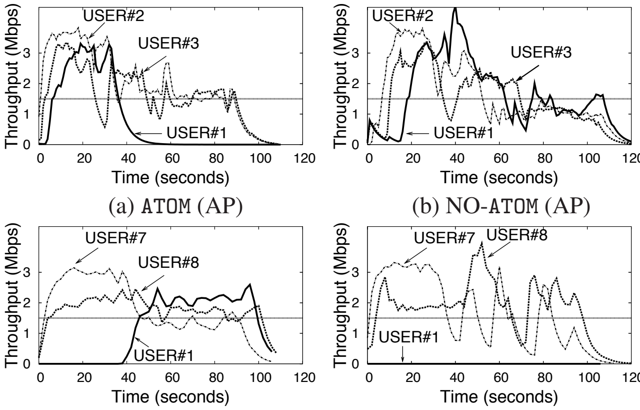
We repeated the same experiment by selecting the interface based on the strongest signal, labeled as Highest-RSSI. In this case, most of the users of WiFi AP1 (Users#1,3 and 4) select the eNodeB while users of WiFi AP2 chose WiFi since users of WiFi AP2 are placed further from the eNodeB than those of WiFi AP1. Hence, the eNodeB gets congested resulting in high number of stalls for all users including Users#7 and 8 that can only connect to the eNodeB (see Figure 7(a)). Although this scenario improves the utilization of the eNodeB (Figure 7(b)), the overall network utilization is lower than ATOM as WiFi AP1 is largely under-utilized. Hence, even in static conditions, current solutions cause severe degradation in user QoE and network under-utilization.

2) Network Dynamics: (i) *User Flows Arrival/Departure*: In this experiment, we have a network of one eNodeB and one WiFi AP. There are 4 users streaming YouTube videos of average bit-rate of 1.5 Mbps. Users#1, 2 and 3 are within WiFi coverage while User#4 can only access the eNodeB. We introduce 2 WiFi users with background traffic at around 30 seconds into the experiment such that the flows are active for about 40 seconds. Figure 8(c) plots the throughput for the 3 WiFi users with NO-ATOM. Initially, the users receive throughput above their requirement of 1.5 Mpbs. However, during the time period from 30 to 70 seconds, the throughput of all the 3 users falls below 1.5 Mbps due to the presence of the 2 background flows on the WiFi AP. On the other hand, ATOM moves the video flow of User#1 to the eNodeB at around 30 seconds as shown in Figure 8(a), (b). ATOM is aware that the eNodeB has sufficient capacity to support the video flow of User#1 without affecting the video of User#4. The throughput achieved by Users#1 and 4 on the eNodeB is shown in Figure 8(b); traffic from the video



(a) ATOM(AP) (b) ATOM (LTE BS) (c) NO-ATOM(AP) (d) Fine-grained traffic management

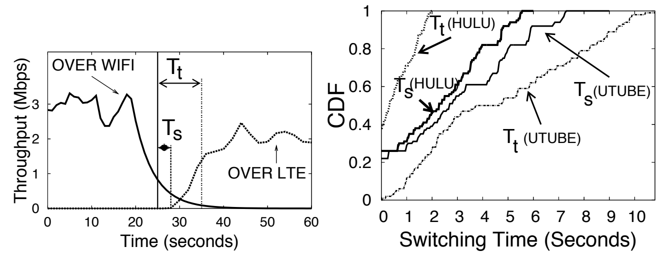
Figure 8: Efficacy of ATOM with network dynamics.



(a) ATOM (AP) (b) NO-ATOM (AP)

(c) ATOM (LTE BS) (b) NO-ATOM (LTE BS)

Figure 9: Efficacy of ATOM with user mobility.



(a) Throughput (YouTube) (b) Switching Time

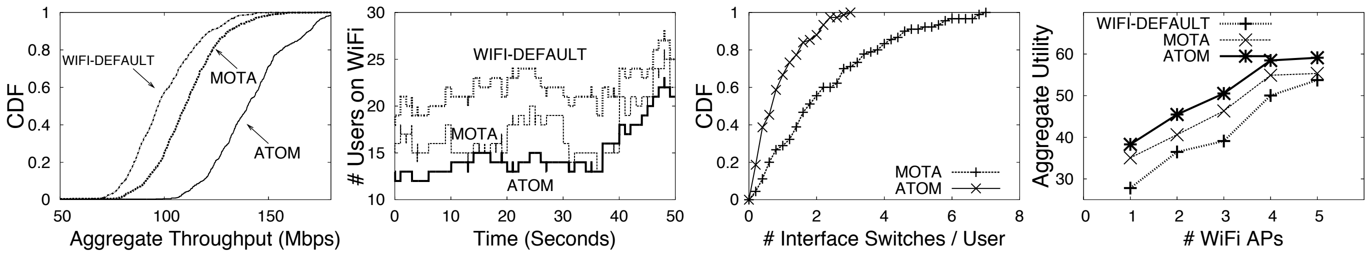
Figure 10: Benchmarking the ISS.

flow of User#1 starts around 30 seconds on the eNodeB. Around 70 seconds, ATOM moves the video flow of User#1 back to the WiFi AP since the 2 background flows ended during that time releasing resources of the WiFi AP. After 70 seconds, the video flows of Users#1,2 and 3 receive throughput above 1.5 Mbps as clear from Figure 8(a) resulting in a smooth video for all the 3 users.

(ii) *User Mobility*: We set up a network with one eNodeB and one WiFi AP. There are 8 users and 6 of them are within the coverage of the WiFi AP. All users stream YouTube videos of average bitrate 1.5 Mbps. Initially, all 6 users are placed close to the WiFi AP such that they receive good throughput and hence a smooth video. At about 30 seconds, 2 users are moved away from the WiFi AP at walking speeds such that they are still in the coverage of the WiFi AP. We plot the throughput obtained by 3 (Users#1, 2 and 3) out of the 6 users over WiFi in Figure 9(b). Clearly, as the users move away from the WiFi AP, more resources are needed to meet the throughput requirement of the users causing network congestion. This in turn causes stalls in the WiFi users' video streams. However the 2 LTE users, specifically Users#7 and 8 get sufficient throughput as shown in Figure 9(d) to stream the video smoothly as the eNodeB has enough resources. On the other hand, ATOM moves Users#1 and 4 to the eNodeB relieving the congestion in the WiFi AP. As seen from Figure 9(a), Users#2 and 3 receive throughput above their requirement of 1.5 Mpbs sustaining good video quality. Although User#1 is moved to the eNodeB at around 40 seconds into the experiment as shown in Figure 9(c), the eNodeB has sufficient capacity to support the video rates of Users#1, 7 and 8. Hence, ATOM adapts to link quality fluctuations due to user mobility.

3) Fine-grained Adaptation: In this experiment, we show the ability of ATOM to perform fine-grained traffic management. There are 8 LTE users within the coverage of the WiFi AP and 4 LTE users outside the WiFi coverage. All 8 users download a large file from the WiFi AP. One of the WiFi users (User#5) also streams a YouTube video of average rate 2 Mbps. All 4 LTE users stream the same YouTube video from the eNodeB. Figure 8(d) plots the average number of video stalls for 4 LTE users and User#5. Scenario 1 represents the case where all the traffic of User#5 is mapped to the WiFi AP since the user is within the coverage of the AP. Clearly, the video flow of User#5 suffers significantly as the WiFi AP is congested. Scenario 2 represents the case with user-level traffic management where both the flows of User#5 (video and file-download) are moved to the LTE network. This results in the LTE network getting congested and the video of all the 5 users suffer. With ATOM's flow-level interface management, the video flow of User#5 is moved to LTE while the file-download flow is kept on WiFi. Since ATOM operates at the granularity of user flows, the increased flexibility allows ATOM to ensure good QoE by reducing the average stalls per user from 6-8 with NO-ATOM to 1-2.

4) Benchmarking the ISS: We investigate the switching time taken by the ISS specifically for HTTP-based video streaming since video traffic accounts for significant percentage of the total traffic. We measure the switching time using two metrics defined as: (i) Start Time (T_s): It is the time taken for downlink traffic to start on the new interface. (ii) Termination Time (T_t): It is the time taken for traffic to completely stop on the current interface. Both the metrics are measured relative to the time that the command to switch the interface is received by the client. Figure 10(a) shows how we measure T_s and T_t by plotting the throughput of a video flow that is moved from WiFi AP to LTE basestation at around 25 seconds. T_s is the time taken for traffic of the flow to start over LTE and T_t is the time taken for traffic to completely stop over WiFi. We setup the experiment by streaming a single video over WiFi initially and configure the ISS to switch the interface of the flow every 30 seconds between WiFi and LTE. We repeat the experiment for several different videos from YouTube (represents HTTP-PD) and Hulu (represents adaptive video streaming). Figure 10(b) plots the CDF of the two metrics T_s and T_t for the different video streams. Clearly, the median switching times are within a couple of seconds and hence, within the expected time-scale for the execution of ATOM. Notice that the times are larger for HTTP-PD streams (YouTube) than the adaptive video streams (Hulu). On further investigation, we noticed that players supporting adaptive video streaming typically request video chunks of lower size (typically 2-4 secs of video) than those requested by regular video streams like YouTube. Also, adaptive video players request the chunks at a rate that closely



(a) Throughput (b) # Assigned WiFi Users (c) Avg Interface Switches (d) Aggr Utility

Figure 11: Large-scale evaluation of ATOM.

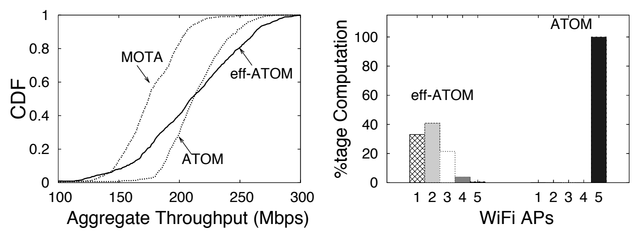
matches the play-out rate to ensure adaptiveness to changing network conditions. This behavior results in Hulu streams having a lower T_s and T_t than those shown by YouTube streams. Given the growing popularity of adaptive video streaming, we expect most video services to show results similar to those shown with Hulu (we did see similar results with CBS, Netflix). Moreover, note that although YouTube streams have a relatively higher value of T_t, as seen in Figure 10(a), the amount of traffic downloaded during that time (25 to 35 seconds) is significantly less than the average rate of the video (2 Mbps) since the traffic consists of the residual bytes for video chunks that were requested before receiving the interface switching command and all subsequent chunks are requested from the new interface.

7.2 Simulations

Set-up: We developed MATLAB® code simulating a network of one eNodeB and multiple WiFi APs (randomly distributed within the eNodeB coverage) and used 3GPP path-loss models to generate user SNRs. We use different rate tables for LTE and WiFi to choose the best link-layer rate for a user based on its SNR. We distribute the users in a uniformly random fashion within the cell such that there is a non-zero probability of a user not falling in the range of any WiFi APs. The inter-access times of flows for each user are exponentially distributed and each user flow is active for 120 seconds. When active, each flow has backlogged downlink traffic. The SNRs vary across different flows over time. The number of users and the parameter of the exponential distribution are jointly chosen such that the number of active user flows in the system varies from 20 to 40 in steady state. The MAC scheduler executes every 10 milliseconds and uses a PF-based policy for the eNodeB and a RR-based policy for the WiFi APs. ATOM is executed every second and it is assumed that interface switching occurs instantaneously.

Reference schemes: We compare ATOM with the following schemes for interface selection: (i) WiFi-Default: This is the case where the users always connect to an available WiFi AP. (ii) MOTA: MOTA [8] is a client-side solution that asynchronously executes the interface selection decision at the client to maximize the utility of a user. MOTA requires additional signaling about the load of each WiFi AP and eNodeB to each client. Similar to ATOM, we employ a log utility function of the throughput for each user in MOTA. MOTA is executed every second on each client and the eNodeB and WiFi APs broadcast the required information every 10 seconds. Each client computes the expected throughput on every interface based on the update received from the APs and the eNodeB every 10 seconds. However, clients always have accurate information about the throughput on the current interface, i.e., the interface on which they have an active flow. Although MOTA may be hard to deploy as it requires additional signaling overhead and standards support, it represents the ideal client-level solution for interface selection.

1) Performance: We setup a network of a single LTE basestation with 3 WiFi APs in its coverage and vary the number of user flows.

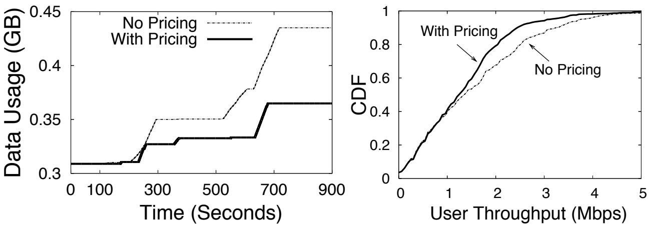


(a)Performance (b) Computation

Figure 12: Performance vs Computational efficiency.

The aggregate throughput obtained by all the clients for the 3 scenarios is shown in Figure 11(a). Figure 11(b) depicts the aggregate number of user flows that are mapped to the WiFi APs over time for a small period in the simulation. As a centralized technique, ATOM maps the appropriate number of user flows to WiFi APs and LTE resulting in better resource utilization and load balancing than MOTA. Although MOTA accounts for the load conditions of the eNodeB and the APs, it is not as efficient resulting in significantly lower throughput than ATOM. In this particular case, ATOM achieves an average aggregate throughput of almost 140 Mbps with a 5 percentile throughput in excess of 100 Mbps. On the other hand, WiFi-Default and MOTA achieve an average aggregate throughput of 90 and 110 Mbps respectively, with a 5th percentile throughput of about 70 and 80 Mbps respectively. A by-product of dynamic traffic management is that it leads to interface switching for user flows. As seen in Figure 11(c), ATOM is effective in achieving a higher throughput while keeping the average number of switches per user per session below 0.5, while MOTA causes an average of 2 switches per user per session. Switching the interface of a user flow causes additional signaling in the mobile network and hence, excessive switching may be undesirable for an operator. To investigate the fairness and QoE of user flows, we plot the aggregate utility obtained by each scheme. As seen in Figure 11(d), ATOM achieves better aggregate utility than MOTA with increasing number of WiFi APs. ATOM thus achieves significant gains not only over naive schemes such as WiFi-Default but is more efficient than distributed schemes such as MOTA.

2) Computational Efficiency: While ATOM is scalable operating at the granularity of a single LTE cell, we also investigate an approach to trade-off performance of ATOM for reducing its computational requirements further. At each epoch, the modified algorithm, namely eff-ATOM is executed only for the WiFi APs that have a change in state and the assignment of the user flows of the other WiFi APs is kept unchanged. Change of state for a WiFi AP occurs if there was at least a user flow that arrived or departed from the WiFi AP or there was a change in the average link layer rate of a user belonging to that WiFi AP in the previous epoch. We use a similar setup for this experiment as the previous one, with 5 WiFi APs. Figure 12(a) compares the performance of eff-ATOM with both ATOM and MOTA. Clearly, although there is a slight degradation in the



(a) Data Usage over LTE (b) User Throughput

Figure 13: Considering interface cost

aggregate throughput for eff-ATOM compared to ATOM, it still performs better than MOTA. Figure 12(b) depicts the percentage of cycles spent on computing the interface assignment for the number of WiFi APs. Specifically, eff-ATOM only computes the interface assignment for 1,2 and 3 WiFi APs 30, 40 and 20% of the times, while ATOM always computes the interface assignment for all the 5 WiFi APs. Hence, if we consider the computation expense of a WiFi AP as a single unit, eff-ATOM is 65% more efficient than ATOM.

3) Pricing: To study the effect of different pricing models, we conducted simulations with a similar setup but added a cost to the LTE interface. The scenario mimics the case in today's networks, where the users have a data cap and pay for usage on the LTE network, while the usage on WiFi networks is free. We plot the CDF of the throughput obtained by one of the users in Figure 13(b) with and without incorporating pricing. With pricing, there is a deterrence for ATOM to move the user flow to the LTE network due to the associated cost. This results in a slightly lower throughput for the user. However, the data usage of the user over LTE is lower over time as shown in Figure 13(a) with the pricing function since the flows of the user are kept over WiFi more often than the case with no pricing. Hence, ATOM allows an operator to balance the utility for additional throughput (QoE) in an interface with its associated cost or data usage on a per user and/or flow basis.

8. RELATED WORK

Commercial Solutions: Technologies from Qualcomm [29] and Interdigital [30] provide WiFi offloading with intelligence mainly at the mobile devices. They claim to manage user flows across WiFi and 3G/LTE networks based on throughput and delay measurements. They rely on integration with the I-WLAN architecture to provide seamless connectivity. Although not widely deployed, the existence of such solutions indicates the importance of managed WiFi offloading. Since these technologies incorporate context in their solution and provide management across 3rd party WLANs that are not managed by the operator, they can be used as complementary solutions to ATOM.

Client-side Optimizations: Recent studies [7, 8, 9, 10, 31] have proposed distributed algorithms for interface selection. Most of these solutions either require additional signaling from the network or leverage P2P to disseminate the current network. As shown in our evaluations, even with network load information, such solutions are not as efficient as a centralized solution such as ATOM. Some studies also argued the feasibility of using public WiFi for offloading 3G traffic [11, 12]. However, the scope is limited to delay tolerant traffic and may require changes to the applications to support the framework. There is also a recent study [32] that proposes seamless switching of user traffic but it does not provide any intelligence for optimal interface selection.

Network Solutions: Several studies have proposed network-driven algorithms for interface selection or user association to optimally balance the load across heterogeneous networks. A few of these works [6, 33, 34, 35] use utility-based optimization to maximize the throughput obtained by the users. However, these works (i) assume idealized settings with little or no consideration of practical constraints (ii) are tightly integrated with the basestation schedulers hindering their deployability and (iii) do not provide an end-to-end solution that dynamically selects interfaces adapting to changes in the mobile network. ATOM on the other hand is a comprehensive solution for traffic management for heterogeneous WiFi-LTE networks that is adaptive, light-weight, scalable and deployable in today's mobile networks.

9. DISCUSSIONS

Mobility: While ATOM works well for static and mobile clients at moderate speeds, clients with vehicular mobility may require additional support to use context information (e.g., speed) to ensure that such users are treated as LTE-only users to avoid unnecessary switching due to limited WiFi coverage.

Scalability: Since ATOM is designed to execute for every LTE cell independently, it can be scaled-up easily as LTE cells are added to the network. Typically in current LTE deployments, the coverage of macro, metro and small cells is greater than that of WiFi APs. Hence, considering each LTE cell in isolation ensures scalability without much compromise in performance. However, we envision that future cellular networks may be significantly more denser with smaller cell sizes resulting in overlapping coverage across several cells. It is an interesting avenue for future work to design a scalable system for such dense deployments to manages flows belonging to several LTE cells and WiFi APs.

Cellular networks have large number of cells and user flows, provisioning both CPU and storage (for state information) resources for ATOM instances to manage each LTE cell can be expensive. In practical deployments, operators could execute ATOM instances to manage only the subset of LTE cells that are loaded beyond a threshold. Moreover, the epoch time T can be increased to reduce processing overhead (say 1 minute) and ATOM can be executed as and when user flows arrive and/or depart.

Signaling Overhead: ATOM relies on feedback messages from the basestations and WiFi access points which introduces extra overhead on the network. In order to provide quantitative insights into the traffic overhead introduced by ATOM, we performed back of the envelope calculations. The feedback message contains the flows current IP address/port numbers, average transmission rate for that user on the LTE basestation and on the WiFi AP(s). Assuming two WiFi APs on average, these values can be composed in 10 bytes and an average of 50 active flows per LTE basestation, the feedback packet would be about 500 bytes per basestation. If we assume that a particular data center manages a network of about 1000 basestations and also an epoch time (T) of 10 seconds, the total data rate for feedback messages is around 400 Kbps. In the same network assuming an average user data traffic rate of 10 Mbps per basestation (typically peak rate of LTE basestations is about 60 Mbps), the total average user data traffic would be around 10 Gbps. Thus, the network overhead for ATOM under the above assumptions would amount to less than 0.05% of aggregate user data traffic.

Excessive Switching: Although the execution time for ATOM will be in the order of several seconds in practice to ensure stability of the system, in certain highly dynamic scenarios, ATOM may cause certain flows to switch frequently between interfaces. However, excessive interface switching for a flow can be avoided by using a deterrence for flows based on the history of switches performed for that flow in the recent past.

Energy Consumption: Although we do not explicitly consider interface energy consumption in ATOM, the energy consumption could be easily incorporated in the utility framework. Depending on the interface, the energy cost can be added as a deterrent for using a particular interface. This can capture scenarios where clients with critical battery levels are assigned to a more energy-efficient interface even though that may not be the best interface in terms of throughput. However, since energy models are complex and depend on the energy consumption of other system components, we exclude it in the design of ATOM to ensure simplicity.

10. CONCLUSION

To summarize, we designed and implemented a standards compatible framework, ATOM that enables an operator to effectively manage traffic flows across a heterogeneous network of LTE and WiFi APs. ATOM consists of two novel components: (i) NIA dynamically assigns interfaces to user flows and (ii) ISS provides seamless interface switching for HTTP-based flows to enable dynamic traffic management, while saving significant backhaul costs for the operators for HTTP-based video streaming and Web browsing, making it an attractive solution for current networks.

Acknowledgments: This paper benefited significantly from MobiCom 2014 reviewer comments, for which we are very thankful. We would also like to thank Narayan Prasad, NEC Labs for various inputs related to the theoretical aspects of the algorithm.

11. REFERENCES

[1] Cisco visual networking index: Global mobile data traffic forecast update, 2009-2014. http://tinyurl.com/klotw6g.

[2] China Mobile WiFi services. http://tinyurl.com/m7lj366.

[3] KDDI WiFi services. http://tinyurl.com/mzpelfv.

[4] AT&T WiFi services. http://www.att.com/gen/general?pid=5949.

[5] AT&T connection manager. *http://tinyurl.com/msgo4ou*.

[6] Ye et. al. User Association for Load Balancing in Heterogeneous Cellular Networks. *arXiv:1205.2833v2*, 2012.

[7] Aryafar et. al. RAT Selection Games in HetNets. *IEEE Infocom*, 2012.

[8] Deb et. al. MOTA: Engineering an Operator Agnostic Mobile Service. ACM Mobicom, 2011.

[9] Coucheney et. al. Fair and Efficient User-Network Association Algorithm for Multi-Technology Wireless Networks. *IEEE Infocom*, 2009.

[10] Zhu et. al. Network Selection in Heterogeneous Wireless Networks: Evolution with Incomplete Information. *IEEE WCNC*, 2010.

[11] Lee et. al. Mobile data offloading: How much can wifi deliver? *IEEE/ACM Transactions on Networking*, 21(2), 2013.

[12] Balasubramanian et. al. Augmenting Mobile 3G Using WiFi. ACM MobiSys '10, 2010.

[13] Ericsson Traffic Report. http://tinyurl.com/l4sg5td.

[14] Aptilo SIM-based auth. sols. http://tinyurl.com/o5pqqvb.

[15] 3GPP Spec for I-WLAN. http://www.3gpp.org/DynaReport/24327.htm.

[16] Chandra et. al. MultiNet: Connecting to Multiple IEEE 802.11 Networks Using a Single Wireless Card. In *IEEE Infocom*, 2004.

[17] Mpeg dash standard. http://dashif.org/mpeg-dash.

[18] F. Kelly. Charging and rate control for elastic traffic. In *European Transactions on Telecommunications*, 1997.

[19] Andrews et. al. Optimal Utility Based Multi-User Throughput Allocation subject to Throughput Contraints. In *IEEE Infocom*, 2005.

[20] Prasad et. al. Exploiting Cell dormancy and Load Balancing in LTE HetNets: Optimizing the proportional fairness utility. In *ICC*, 2014.

[21] Sandvine traffic report. http://tinyurl.com/ntaddrh.

[22] Erman et. al. To Cache or not to Cache: The 3G case. In *IEEE Internet Computing*, 2011.

[23] OpenEPC. http://www.openepc.net.

[24] Pantech LTE dongle. http://www.pantechusa.com/modems.

[25] Squid HTTP Proxy. http://www.squid-cache.org/.

[26] shrpx HTTP/SPDY Proxy. https://github.com/tatsuhiro-t/spdylay.

[27] SPDY web protocol. http://www.chromium.org/spdy.

[28] Atom project page. http://tinyurl.com/pousloq.

[29] Qualcomm CnE solution. http://tinyurl.com/ky29ogc.

[30] Interdigital SAM solution. http://tinyurl.com/q94s674.

[31] Pedrasa et. al. Information Exchange for Enhanced Network Selection. In *WCNC*, 2010.

[32] Rahmati et. al. Seamless TCP Migration on Smartphones without Network Support. In *Transactions on Mobile Computing*, 2014.

[33] Wang et. al. Mathematical Modeling for Network Selection in Heterogeneous Wireless Networks. *IEEE Comm. Survey and Tutorial*, 2012.

[34] Shen et. al. Cost-Function-based Network Selection Strategy in Integrated Wireless and Mobile Networks. *IEEE AINAW*, 2007.

[35] Chan et. al. A Utility-Based Network Selection Scheme for Multiple Services in Heterogeneous Networks. *IEEE WNCMC*, 2005.

Appendix: To prove that Problem 1 is NP-Hard, we consider a simple instance of the problem: Given a set of $\mathcal{S}$ user flows, find a solution to split the flows among a LTE basestation L and a WiFi AP W so as to maximize the overall utility (given by equation 4). Each flow belongs to a user whose PHY rates for LTE basestation L are $r_{Wi} = 1$ and WiFi AP W are $r_{Li} = \frac{1}{w_i}$ resp.

Proof: The throughput for a particular user on WiFi AP with $\mathcal{S}_W$ user flows is given by (Equation 3):

$$t_{Wi} = \frac{1}{\sum_{i \in \mathcal{S}_W} \frac{w_i}{r_{Wi}}} \implies t_{Wi} = \frac{1}{\sum_{i \in \mathcal{S}_W} w_i} \quad \forall i \in \mathcal{S}_W$$

Hence the utility U_W for all the flows assigned to WiFi is given as:

$$\begin{aligned} U_W &= \sum_{i \in \mathcal{S}_W} w_i \times \log(t_{Wi}) \\ U_W &= \sum_{i \in \mathcal{S}_W} w_i \times \log(\frac{1}{\sum_{i \in \mathcal{S}_W} w_i}) \end{aligned} \tag{5}$$

Similarly, the throughput for a particular user on the LTE basestation with $\mathcal{S}_L$ user flows is given by (Equation 2):

$$t_{Li} = \frac{w_i \times r_{Li}}{\sum_{i \in \mathcal{S}_L} w_i} \implies t_{Li} = \frac{1}{\sum_{i \in \mathcal{S}_L} w_i} \quad \forall i \in \mathcal{S}_L$$

Hence the utility U_L for all the flows assigned to LTE is given as:

$$\begin{aligned} U_L &= \sum_{i \in \mathcal{S}_L} w_i \times \log(t_{Li}) \\ U_L &= \sum_{i \in \mathcal{S}_L} w_i \times \log(\frac{1}{\sum_{i \in \mathcal{S}_L} w_i}) \end{aligned} \tag{6}$$

Let $X = \sum_{i \in \mathcal{S}_L} w_i$. Applying normalized weights, without loss of generality: $1 - X = \sum_{i \in \mathcal{S}_W} w_i$
Hence, the overall utility of the system U is given by

$$\begin{aligned} U &= U_W + U_L \\ U &= X \log(\frac{1}{X}) + (1 - X) \log(\frac{1}{1 - X}) \end{aligned} \tag{7}$$

The solution that maximizes the above utility function is $X = \frac{1}{2}$. Hence, Problem 1 can be defined as: Given a set of flows $\mathcal{S}$, the solution should return a set of flows $\mathcal{S}_W$ and a set of flows $\mathcal{S}_L$ such that the sum of the weights of the flows belonging to the two sets are equal. This is an instance of the subset sum problem (partition problem): Given a set of k integers, the solution should return two subsets such that the sum of the integers of the first set is equal to that of the second set. Our problem can be mapped to a subset sum problem where the input is the set S with elements that have a weight w_i, and the output will be two sets such that the sum of the weights of the elements of each set are equal. Since, subset sum problem is proven to be NP Complete, the proof is sufficient to show that Problem 1 is NP Hard.