

OpenFlow over Wireless Networks: Performance Analysis

G. Araniti⁺, J. Cosmas^{*}, A. Iera⁺, A. Molinaro⁺, R. Morabito⁺, A. Orsino⁺

⁺ ARTS Lab., DIIES Dep., University Mediterranea of Reggio Calabria, Italy

^{*} WNCC, Brunel University, London, UK

email: {araniti, antonio.iera, antonella.molinaro, antonino.orsino}@unirc.it,
john.cosmas@brunel.ac.uk

Abstract—Software Defined Networking (SDN) and OpenFlow represent the most commonly deployed approaches of the so called *Programmable Networks*. SDN is an emerging network architecture, which performs the subdivision of control plane and data plane and allows greater speed, greater scalability, and greater ductility in terms of routing and forwarding. OpenFlow, instead, is an SDN component that characterizes the communication between a controller and the devices within the SDN network. Although SDN and OpenFlow work efficiently in wired networks, the integration of these new paradigms over the wireless networks represents still an open issue. In fact, in a dense wireless scenario with a high number of base stations and a limited radio spectrum is still difficult to manage in a proper way operations such as radio spectrum allocation, interference management, handover management, and load balancing among the cells. In this paper, we present a performance analysis of the SDN and OpenFlow over wireless networks. The aim is to evaluate potential advantages introduced by the SDN architecture in terms of Quality of Service (QoS) metrics such as throughput, packet loss, end-to-end delay, and packet delivery ratio. An exhaustive simulation campaign has been conducted through the OMNeT++ network simulator framework by taking into account the new generation IP-based broadband wireless networks. The obtained results show that the SDN architecture with OpenFlow introduces benefits in the network in terms of resource management, transmission data-rate and end-to-end delay.

Index Terms—OpenFlow; Routing; Next generation of Broadcasting system; Monitoring; IP Networks; Performance evaluation.

I. INTRODUCTION

THE fast growing of large scale networks represented by the current fourth generation mobile networks (4G) and future fifth generation mobile networks (5G) poses new challenges and problems related to the efficient management of the radio spectrum, the packet routing and forwarding. In particular, current wireless infrastructures are becoming crowded of small BSs and devices with different power and coverage [1].

The dense deployment of these low power nodes increases the power consumption of the network with a consequent degradation of the system performance due to the inter-cell and intra-cell interference [2]. In addition, the routing protocols in order to deliver efficiently the data flows without increase the costs of the signaling messages and end-to-end delays represent still an open issue [1].

Although the traditional routing protocols are able to efficiently manage the packet transmission in large-scale networks, due to the small size of new base stations, other approaches and paradigms have to be considered. In fact, in current wireless networks the coordination of the control plane decisions have to be made across several neighboring base stations simultaneously. Traditional distributed algorithms, as a consequence, do not manage in a proper way the network interference and the traffic load in presence of a high number of base stations with a degradation of network performance, especially in terms of latency [3] [4] [5].

Software Defined Network (SDN) and its approach OpenFlow represent the new paradigms of the so called *Programmable Networks* and can be a possible solution to these problems [6]. SDN is an emerging network architecture, which splits the Control Plane (CP) and Data Plane (DP) functionalities to allow a higher data speed, scalability, and flexibility in terms of routing and forwarding. In particular, the data plane is a hardware component of the SDN architecture that implements the packet forwarding and switching.

The control plane, instead, is a software or logical entity, which processes all the routing decisions taken by the data plane. For example, when a conventional router or switch receives a packet, the path of the packet is decided in the CP, whereas the physical action of this decision is executed in the DP. Although in a classical router or switch, the fast packet forwarding (data path) and the high level routing decisions (control path) occur on the same device, an OpenFlow switch separates these two functions. The data path functionality still resides on the switch, while high-level routing decisions are located in a separate controller, typically a server.

In this paper, a preliminary performance analysis of OpenFlow over wireless networks is presented with the aim to evaluate pro and cons in terms of Quality of Service (QoS). Therefore, we provide a comparison between the standard approach (represented by a normal switch) and the OpenFlow approach (represented by a switch that separates the control plane and the data plane). In particular, the analysis is conducted by taking into account the new generation broadband wireless networks that are designed to work over IP protocol and support IP-based applications such as web browsing, VoIP, and streaming video.

Although in literature, there are several works that present the benefits introduced by SDN-OpenFlow paradigm in wired

scenarios, few works focus on the usage of the SDN in wireless networks. In [7], a really interesting flow-based hybrid mechanism to improve the performance in wireless OpenFlow switched networks is presented. The main idea of this scheme is that when the transmission of wireless OpenFlow switch fails, one of neighbouring wireless OpenFlow switches with better channel quality retransmits the lost frame.

In [8], a comparison among the most known routing protocols and OpenFlow routing is conducted by exploiting a testbed for Wireless Mesh Networks (WMN). This work shows that an acceptable performance improvement can be obtained by using OpenFlow with respect to the 802.11s standard.

The novel concept of Software Defined Wireless Networks (SDWN) is considered in [9]. Li et al. provide challenges and open issues about the integration of the SDN architecture within the cellular networks. In particular, the OpenFlow controller is responsible of the forwarding procedures by allowing the network operators to direct the traffic, change the traffic paths and schedule the traffic according to QoS requirements.

The limits of these performance analysis, as also described in [10], are mainly due to the use of testbeds that allow to test a limited number of user applications, network architectures, and topologies. Therefore, the network scalability is not be evaluated in detail and only few network metrics are investigated. As a consequence, in our work we use a real-time simulator in order to provide a complete evaluation of different typical wireless scenarios by varying the type and the length of different multimedia applications.

The remainder of the paper is organized as follows. Section II briefly describes SDN architecture and the OpenFlow approach, whereas a comparison between traditional routing and SDN routing is introduced in Section III. The simulation results are presented in section IV and, finally, conclusive remarks and future works are given in Section V.

II. SDN AND OPENFLOW

A. SDN Architecture

SDN is an emerging technology architecture with the aim to divide the network control in logical and physical entities. In particular, the network infrastructure is abstracted to the application and network devices and can be treated as a logical or virtual entity. Fig. 1 shown a typical SDN architecture. The intelligence of the network is based on software-based SDN controller, which has a global view of the network.

The physical delivery of the flow data, instead, is represented by the SDN switch that follows the rules and constraints imposed by the SDN controller. In this way, the complexity of the network is low and the network operators manage the entire network for a single logical point. Indeed, the network begins entirely programmable and the complex operations of maintenance, configuration and installation are executed in a simple manner (i.e., generally, by line of code). On the other side, the network behavior can change in real-time thanks to the centralized software intelligence of the networks controllers.

The SDN controller, also, supports a number of drivers that control the behavior of the underlying network elements. In

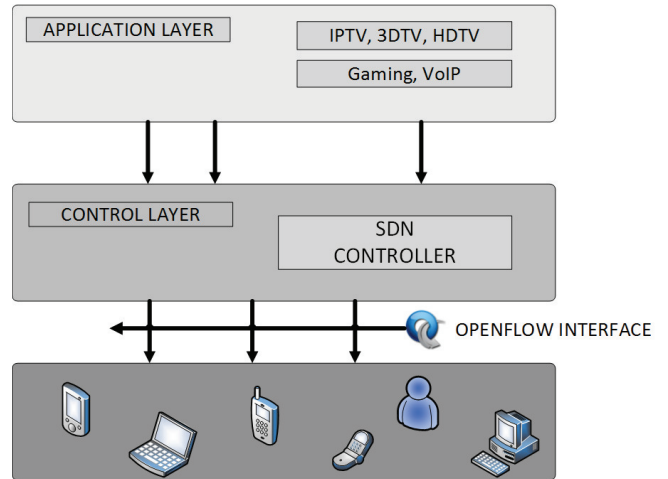


Fig. 1: SDN Architecture

this way, the network provides the desired network services. The controller provides management plane functionality, such as performance and fault management via Simple Networks Management Protocol (SNMP) and other standard protocols, and it typically handles configuration management of OpenFlow compliant devices in order to provide network topology, forwarding, QoS, and link management.

Finally, with the usage of the SDN the complexity of the devices or base stations within the networks is low due to the fact that they no longer need to understand and process thousands of protocol standards but merely accept instructions from the SDN controllers.

B. OpenFlow Specification

The OpenFlow paradigm represents a protocol in order to allow software application to configure and manage the flow tables of different network switches. An OpenFlow network is formed basically (i) by one or more OpenFlow switches which contains a Flow-table with the defined flow entries and the actions to be performed, (ii) one or more Controller which are the responsible of adding and dropping flow entries, and (iii) a Secure-Channel that interconnects the switch with the controller and the OpenFlow Protocol for signaling the whole architecture

The OpenFlow switch that is connected to the controller through the secure channel is a simple network component that forward the data flow from a client to a server and viceversa. Based on its functionalities, OpenFlow switches are divided into two main categories: *dedicated OpenFlow switches* and *OpenFlow enabled switches*. In the first category, the OpenFlow switches are characterized by a simple hardware component with the only task to deliver the data flow packets, whereas in the second category, the commercial switches are enhanced with the OpenFlow standard.

The OpenFlow controller is the network element that is in charge for managing the OpenFlow switches and making decision about the requirement and the behavior of the different packets present in the network. The Controller can be an active or a passive component depending on the function

implemented or performed [11]. The Controller is passive when add or remove flow entries from the Flow Table in a statically manner, while is active when in addition to its basic functions (for example, delivery packet decision or particular rules for a packet flow) performs additional activities such as routing and network access. In addition, the number of the controllers and their positions affect the performance of the network and the fault tolerance capabilities.

III. TRADITIONAL ROUTING VS SDN ROUTING

In order to better understand how the OpenFlow technology works, it is necessary to remind what happens in a classical router or switch where the packet forwarding (also called data path) and the high level routing decision (control path) occur on the same device. Fig. 2 shows the difference between traditional switch design and OpenFlow design.

An OpenFlow network is formed by (i) one or more OpenFlow-switches which contain a Flow-table with the defined flow entries and the actions to be performed; (ii) one or more Controller which are the responsible of adding and dropping flow entries, (iii) a Secure-Channel that interconnects the switch with the controller and the OpenFlow Protocol for signaling the whole architecture.

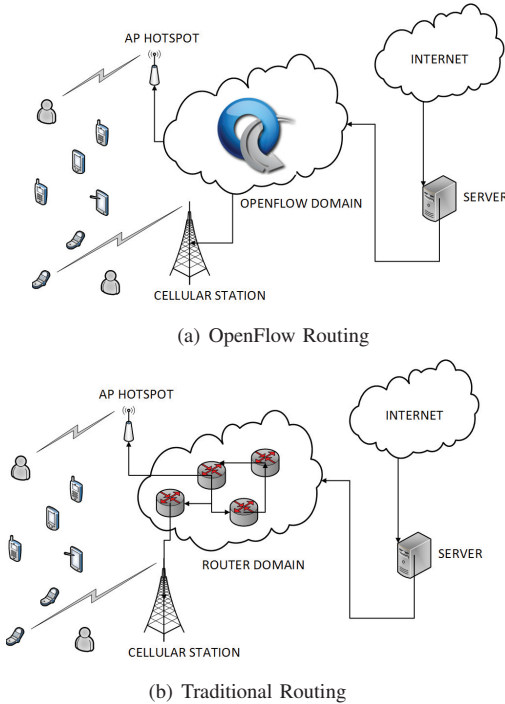


Fig. 2: OpenFlow Routing vs Traditional Routing

OpenFlow Switch and Controller communicate via the OpenFlow protocol that is a complex and versatile model in which several messages and procedures are defined in order to allow the right management of the network [12]. The data path operations of an OpenFlow Switch are based according to a clean flow table abstraction. In fact, each flow table entry contains a set of different packet fields to match and an action (i.e. send-out-port, modify-field, or drop) that has

to be executed. In addition, the OpenFlow switch performs two different actions when a packet arrives. If the packet is unknown, the OpenFlow switch (this implies that it has no matching flow entries in the flow table) sends this packet to the controller. In particular, the communication between switch and controller is guaranteed by a secure channel that is encrypted using the Secure Socket Layer (SSL) or Transport Layer Security (TLS) [13].

Then, the controller decides how to manage this packet or which task has to performs such as the packet can be dropped or can be added a new flow entry in order to forward similar packets in the future [14]. If the packet is within the own flow table, an appropriate action is made by the OpenFlow switch (such as forward the packet into a specific port of the switch).

Therefore, the main difference between the design of the traditional distributed network and an OpenFlow network lies in the structure of the control plane network. For traditional distributed architectures like the Border Gateway Protocol (BGP), Open Short Path First (OSPF) and the Intermediate System to Intermediate System (IS-IS), the control topology is peer to peer. It means that each forwarding device listens for events from its peers and makes independent decisions based on the local view.

In contrast to the distributed architectures, the control planes in the decoupled networks are more like client-server networks. In these networks the packet forwarding devices have a very limited decision making capabilities and have to implement the control decisions made by the controller.

IV. SIMULATION RESULTS

In order to evaluate the impact of the OpenFlow routing within a wireless scenario, an exhaustive simulation campaign has been conducted through the network simulator OMNeT++ [15]. OMNeT++ is a discrete events framework simulator based on C++ library that is extendible, modular, and able to simulate the most used wireless standards. In addition, an Eclipse-based IDE is provided in order to have a graphical runtime environment and to host other module developed by the community research.

The scenario investigated takes into account situations and/or places in which the provision of connectivity is needed such as computer labs, classrooms, large conference rooms, hotspot area, airports, campus university, convention centers, and so on. In particular, we consider a server connected to the internet that deliver data-flows to end-users, through an access point WiFi. The application traffic model used in this study is derived from the modelling carried out in [16].

The routing and forwarding of the delivered packets is managed with the usage of traditional routing protocols at first and next with the OpenFlow routing protocol. Furthermore, the traditional routing protocol used during the simulation is represented by the Open Shortest Path First (OSPF) [17]. The OSPF is an interior gateway protocol that routes Internet Protocol (IP) packets solely within a single routing domain (autonomous system). It gathers link state information from available routers and constructs a topology map of the network. The topology determines the routing table presented to

the Internet Layer which makes routing decisions based solely on the destination IP address found in IP packets.

The OpenFlow protocol, instead, consists of a controller that keeps a general overview of the interfaced networks and is directly connected to two switches. The first one is connected to the access point (since the OpenFlow switch and the access point are combined into a single entity, forming an OpenFlow Access Point), while the second one is connected to the remaining part of the network.

The others parameters used in the simulation campaign are shown in Table I:

Parameter	Value
Wireless Standard	802.11g
Frequency	2.4 GHz
Channel	11 22 MHz channels
Data-Rate	54 Mbps
Range	100 m
Transmission Power	0.005 W
Number of UEs	5, 10, 20, 30
Number of AP	1
Application Type 1	Interactive Gaming
Application Type 2	VoIP
Application Type 3	Streaming Audio
Application Type 4	IPTV
Session length	3, 5, 10, 20 minutes

TABLE I: Main System Parameters

The Jitter by considering different types of applications is shown in Fig. 3. The obtained results remarks that OpenFlow allows to obtain better performance levels with respect to traditional routing because least congested routes are used.

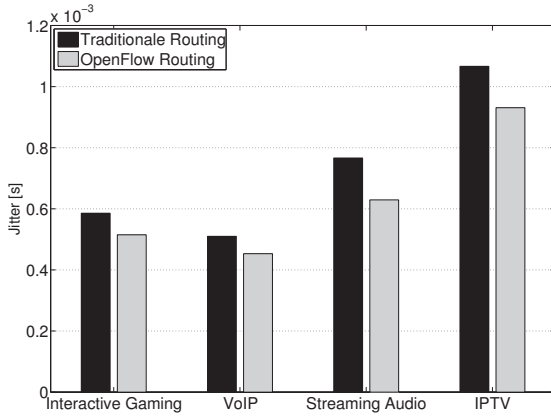


Fig. 3: Jitter for Different Application Type

The End-to-End Delay in presence of a fixed number of users equal to 5 is depicted in Fig. 4. In case of VoIP applications there is no relevant difference between two routing protocols, whereas in case of Interactive Gaming and Streaming Audio the OpenFlow has an acceptable loss of performance. The highest gain is achieved when an IPTV application is considered. In fact, OpenFlow routing enhances the traditional routing protocol and achieves a gain of about 47%.

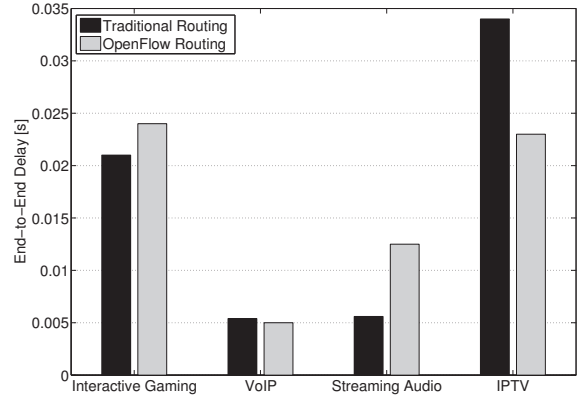


Fig. 4: End-to-End Delay for Different Application Type

In order to evaluate the robustness and the accuracy of the considered routing protocol, the Packet Delivery Ratio (PDR) is shown in Fig. 5. In all the considered applications, the traditional routing achieves better results than the OpenFlow routing. The gap is in any case acceptable due the fact that the highest lost is of about 8%. It means that the OpenFlow routing enhances the performance in term of End-to-End delay without a high degradation of the system metrics such as the PDR.

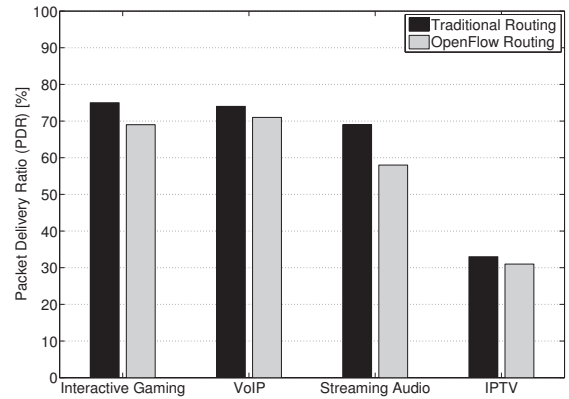


Fig. 5: PDR for Different Application Type

The behavior in terms of End to End delay (Fig. 6) shows how OpenFlow routing introduces considerable performance improvements. Indeed, for short sessions, the end-to-end delay is almost halved by exploiting this new technology. However, for longer sessions (i.e. 20 minutes), the performance of the two strategies coincides. On the other hand, the analysis of the throughput performance shows that OpenFlow routing performs worse when more than 20 users are considered, as shown in Fig. 7.

This is due to the signaling overhead. Also when traditional routing achieves a better performance with respect to OpenFlow routing, the difference is not significant. Anyway, for larger networks and applications exploiting a high network

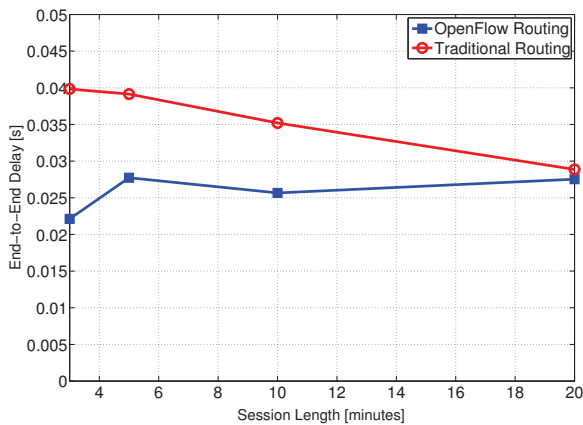


Fig. 6: End-to-End delay vs Session length

capacity we expect an Open Flow percentage overhead much smaller and therefore a greater throughput performance.

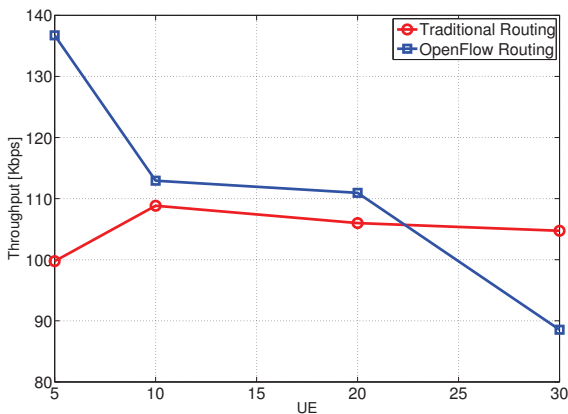


Fig. 7: Throughput vs Number of Users

V. CONCLUSION AND FUTURE WORKS

In this paper the impact of OpenFlow over wireless networks has been investigated in order to evaluate the enhancements introduced by the usage of the SDN architecture.

The simulation campaign has been conducted by the usage of the OMNeT++ simulator by considering different types of multimedia applications (Interactive Gaming, VoIP, Streaming Audio and IPTV). A wireless scenario with a WiFi access point by varying the number of the attached users and the length of the application session is considered. The obtained results show that the usage of the SDN architecture and OpenFlow standard introduces benefits in terms of end-to-end delay, throughput, and jitter.

A possible extension of this work could be to consider an high number of WiFi access point with users that request different types of applications simultaneously. The integration of the current 4G networks in order to evaluate the cooperation of 3GPP and non-3GPP standard could be also taken into consideration. Finally, the effect of the OpenFlow routing

in machine type communication such as Machine-to-Machine (M2M), Device-to-Device (D2D), and Vehicular-to-Vehicular (V2V) could be investigated.

REFERENCES

- [1] I. Bisio, A. Delfino, G. Luzzati, F. Lavagetto, M. Marchese, C. Fra, M. Valla, "Opportunistic estimation of television audience through smart-phones," *Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, 2012 International Symposium on , vol., no., pp.1,5, 8-11 July 2012.
- [2] C. Sacchi, F. Granelli, C. Schlegel, "A QoE-Oriented Strategy for OFDMA Radio Resource Allocation Based on Min-MOS Maximization," *Communications Letters, IEEE* , vol.15, no.5, pp.494,496, May 2011.
- [3] G. Araniti, V. Scordamaglia, A. Molinaro, A. Iera, G. Interdonato, F. Spano, "Optimizing point-to-multipoint transmissions in High Speed Packet Access networks," *Broadband Multimedia Systems and Broadcasting (BMSB)*, 2011 IEEE International Symposium on , vol., no., pp.1,5, 8-10 June 2011.
- [4] G. Araniti, V. Scordamaglia, M. Condoluci, A. Molinaro, A. Iera, "Efficient Frequency Domain Packet scheduler for Point-to-Multipoint transmissions in LTE networks," *Communications (ICC)*, 2012 IEEE International Conference on , vol., no., pp.4405,4409, 10-15 June 2012.
- [5] M. Condoluci, G. Araniti, A. Molinaro, A. Iera, J. Cosmas, "On the impact of frequency selectivity on multicast subgroup formation in 4G networks," *Broadband Multimedia Systems and Broadcasting (BMSB)*, 2013 IEEE International Symposium on , vol., no., pp.1,6, 5-7 June 2013.
- [6] Nick McKeown , Tom Anderson , Hari Balakrishnan , Guru Parulkar , Larry Peterson , Jennifer Rexford , Scott Shenker , Jonathan Turner, OpenFlow: enabling innovation in campus networks, *ACM SIGCOMM Computer Communication Review*, v.38 n.2, April 2008.
- [7] Oh Hayoung, Lee Junjie, Kim Chongkwon, "A Flow-Based Hybrid Mechanism to Improve Performance in NOX and Wireless OpenFlow Switch Networks," *Vehicular Technology Conference (VTC Fall)*, 2011 IEEE , vol., no., pp.1,4, 5-8 Sept. 2011.
- [8] Chung Joaquin, G. Gonzalez, I. Armuelles, T. Robles, R. Alcarria, A. Morales, "Characterizing the multimedia service capacity of wireless mesh networks for rural communities," *Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2012 IEEE 8th International Conference on , vol., no., pp.628,635, 8-10 Oct. 2012.
- [9] Li, L.E.; Mao, Z.M.; Rexford, J., "Toward Software-Defined Cellular Networks," *Software Defined Networking (EWSN)*, 2012 European Workshop on , vol., no., pp.7,12, 25-26 Oct. 2012.
- [10] P. Dely, A. Kassler, N. Bayer, "OpenFlow for Wireless Mesh Networks," *Computer Communications and Networks (ICCCN)*, 2011 Proceedings of 20th International Conference on , vol., no., pp.1,6, July 31 2011-Aug. 4 2011.
- [11] H. Ali-Ahmad, C. Cicconetti, A. de la Oliva, V. Mancuso, M. Reddy Sama, P. Seite, S. Shanmugalingam, "An SDN-Based Network Architecture for Extremely Dense Wireless Networks," *Future Networks and Services (SDN4FNS)*, 2013 IEEE SDN for , vol., no., pp.1,7, 11-13 Nov. 2013.
- [12] Technical Solution Guide HP: "Open-Flow Protocol Overview" . Available at: <http://h17007.www1.hp.com/docs/networking/solutions/sdn/devcenter/03-HP-OpenFlow-Technical-Overview-TSG-v1-2013-10-01.pdf>
- [13] Open Networking Foundation: "OpenFlow Switch Errata". Available at: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.0.2.pdf>.
- [14] HP: "OpenFlow FAQ. Question Answer: OpenFlow and Software Defined Networking". Available at: <http://h17007.www1.hp.com/docs/openflow/openflow-faq.pdf>.
- [15] OMNeT++ Simulation Network. Available at: <http://www.omnetpp.org/>.
- [16] M. Needham, J. Harris, "Traffic and network modeling for next generation applications," *Broadband Multimedia Systems and Broadcasting*, 2008 IEEE International Symposium on , vol., no., pp.1,18, March 31 2008-April 2 2008.
- [17] W.V. Wollman, Y. Barsoum, "Overview of open shortest path first, version 2 (OSPF V2) routing in the tactical environment," *Military Communications Conference, 1995. MILCOM '95, Conference Record, IEEE* , vol.3, no., pp.925,930 vol.3, 8 Nov 1995