Wireless Software Defined Networks: Challenges and Opportunities

Claude Chaudet
Telecom ParisTech, Institut Telecom
Paris, France
claude.chaudet@telecom-paristech.fr

Yoram Haddad
Jerusalem College of Technology
Jerusalem, Israel
haddad@jct.ac.il

***Abstract*—Software Defined Networking (SDN) is a network paradigm that relies on the separation of the control and forwarding planes in IP networks. The interconnection devices take forwarding decisions solely based on a set of multi-criteria policy rules defined by external applications called *controllers*. It is possible to let multiple controllers manage each element of a given network, which allows to create independent networks on the same physical infrastructure. If the implementation of SDN in wired networks is relatively easy, it poses a lot of radio-specific problems in the wireless domain, related to link isolation or to channel estimation. Nevertheless, the wireless domain is also where SDN bears the highest potential, as it provides functions that could foster a better collaboration between access points to reduce interferences or to enhance security. This paper reviews some benefits of wireless SDN and exposes related challenges.**

I. Introduction

Our society becomes more and more connected everyday. At the end of 2011, about a third of the world population was connected to the Internet (Source: Internet World Stats) and the total volume of data exchanged increases by 40 % to more than 100 % per year over the last decade. People upload more and more data on cloud services, produce and access larger and larger multimedia contents and synchronize replicas of their documents across fixed and mobile terminals through the network.

This evolution is possible thanks to the technology improvements in the network, especially at the access network, where optical fibers get closer and closer to the end user. Reciprocally, this demand also drives the networks evolutions. In wired networks, new links and devices are installed regularly to extend the network capacity. This re-dimensioning has a cost, as it sometimes requires digging new cables. It also has a limit, as it increases the network complexity and its diameter. However, it provides enough flexibility to wait for the next technology upgrade.

Limits are harder to overcome in the wireless domain and users often experience data rate limitations or services unavailability when using, for example, their cellular network access. Radio bandwidth for a given technology is a limited resource and is controlled by national regulations. As a result, the capacity of a wireless technology is generally one level below its wired equivalent. Globally, the wireless spectrum is underutilized, even though frequency bands are either reserved for current or future use or for potential tactical communications. Software defined radio (aka cognitive radio) could allow IP traffic to use empty or already allocated frequency bands opportunistically. However, the legitimate owner of the channel shall have an absolute access priority and secondary traffic will have to be diverted when primary traffic is detected. This somehow limits the usage of such techniques to elastic traffic.

The spectral efficiency of most of today's the technologies is high. LTE for example comes close to 80 % of the Shannon limit. However, as radio channels are often shared between concurrent users, increasing the capacity offered to wireless users is still possible. It generally consists in limiting the interferences between cells through an optimized network channels allocation or through the reduction of transmission ranges to form smaller cells. Both strategies are well-known and efficient algorithms exist to optimize the spatial reuse and coverage in one operator's network.

However, these techniques do not work anymore when multiple operators need to share the same wireless channel. The lack of coordination between close access points limits optimization possibilities, as an access point chooses its parameters based on partial information transmitted by its manager, or optimistically, communicating with close access points. The situation where terminals experience collisions between packets sent by mutually out-of-range access points remains unsolved.

Software Defined Networks (SDN) propose to separate the control plane from the forwarding plane in IP networks. SDN rely on the presence of at least one controller entity that defines the data paths based on

978-1-4673-5756-2/13/$31.00 ©2013 IEEE. From the *2013 IEEE International Conference on Microwaves, Communications, Antennas and Electronic Systems (COMCAS 2013), Tel Aviv, Israel, 21-23 October 2013.*

information acquired from a collection of network devices. From the performance optimization point of view, this control entity is the perfect location for deciding of access points operational parameters.

In this article, after a short presentation of SDN and OpenFlow, identify the key functions whose wireless adaptation pose challenges. We then expose how a fully functional wireless SDN can help solving some wireless-specific issues before concluding.

II. Software Defined Networks and Openflow

Software Defined Networking (SDN) is often defined as the decoupling of the control and the forwarding planes. This means that the set of rules that defines how interconnection elements forward frames can be defined in any way. For example, paths can be defined by a remote centralized management entity or by a group of such entities, rather than by using a distributed routing protocol or by examining packets source and destination addresses locally in an interconnection device.

In 2008, a famous white paper [8] introduced *OpenFlow*, an Ethernet implementation of SDN. OpenFlow relies on a set of interconnection devices, switches and routers that only possess the capacity to classify incoming packets into *flows* and to take per-flow forwarding decisions. A flow is defined by a set of matching rules on 12 distinctive distinctive fields of the typical Ethernet/IP/UDP header (L2 and L3 addresses, VLAN information, ports, etc.).

The flows definition and the forwarding policies, i.e. the actions to take for each flow, are defined by a central entity called a *controller*. A single controller can theoretically control an unlimited number of devices and multiple controllers can share the management of a single device. The set of controllers form an execution environment for network programming, a *Network Operating System* (NOS) [6]. The most famous open source controller implementation was called NOX [2]. It was developed by the team that is at the origin of OpenFlow [6] to provide a platform to program and control switches through the OpenFlow API. It is now replaced by a Python-based evolution called POX, available from the same source.

When a packet enters a switch (or more generally any interconnection equipment), the switch tries to classify it into an existing flow by matching the 12-tuple against a local set of associations. When the matching succeeds, the switch executes the corresponding action (e.g. forward, drop, prioritize). If matching fails, the packet is forwarded to the controller that takes decision based on a client-based policy. [7] measured the time required to process a packet between 220 μs and 245 μs. [4] shows that, at the data plane level, OpenFlow introduces a drop of performance of about 11 % regarding latency and throughput compared to regular routing for small packets (64 bytes). The effect is negligible for larger packets.

As the granularity is not the packet anymore, but the flow, it is easier to give a higher priority to packets coming from a premium client and queue packets belonging to regular users, even if they belong to the same service and go to the same destination. Moreover, the controller can update the policies dynamically, depending on the state of the network. To this extent, the managed devices regularly send their own state and their perception of the network to the controller. With this information, the controller is able to acquire a global vision of the network that can span across multiple LANs.

Among the flows range, some flow IDs can be dedicated for experimental purposes. Provided that the interconnection devices are error-prone and powerful enough to handle a large number of flow IDs, these experimental flows can define a separate, virtual, network that does not interfere with the production network of the ISPs. This ability to separate the flows space in distinct subspaces is referred to as *slicing*. This is made possible by inserting an application between the physical devices and the controllers, called *FlowVisor*. FlowVisor gives the illusion to each controller that it controls a dedicated network. Each controller acts on a slice as if it were on a dedicated network and all slices are multiplexed over the same physical infrastructure (forwarding plane).

Software Defined Networks and OpenFlow were designed with infrastructure networks in mind, and more specifically for wired networks and the adaptation to the wireless context is not straightforward. For example, most of the current wireless SDN implementations only work well when slicing uses different channels. [5] implemented a wireless mesh network using SDN and their work showed serious issues regarding the time required to set up a new rule from the controller to the access points, the time needed to parse rules for an incoming packet, or the control traffic volume. As part of the Stanford ONRC OpenRoads project, Yap *et al.* [10] confirm that slicing is difficult on the same Wi-Fi channel. They also show that, if the control traffic is sent on a wired backbone, the corresponding load increase accounts for less than 0.05 %.

III. Wireless SDN challenges

Implementing SDN requires at least to be able to define slices and to limit interactions between these slices, and to let the network devices measure and report their status to the relevant controllers, which are non trivial operations with the wireless medium.

A. Slicing and Channels Isolation

Implementing slicing requires at least being able to isolate communication channels so that a FlowVisor application can present non-interfering networks to different coordinators. In a wired network, it is possible to isolate links to a certain extent by reserving, for example, different wavelengths in an optical fiber.

A wireless FlowVisor has to manage a limited number $\mathcal{N}$ of independent channels (e.g. the three non-overlapping Wi-Fi channels in the 2.4 GHz band). The easiest solution, at least in appearance, consists in defining at most $\mathcal{N}$ slices, which still can raise planning problems, as close links interfere. FlowVisor therefore needs to have the vision of a whole area and to plan geographical channels reuse in a large network.

Defining more slices than the number of channels defined by the technology is possible, but poses interesting challenges. Using time division multiplexing, for example, requires a fine coordination of close access points and a very fine-grained time synchronization so that time slots do not overlap. Frequency division multiplexing requires reserving a guard interval between adjacent channels, which wastes bandwidth when the number of sub-channels increases. Random access protocols can statistically provide to users the same throughput, but the fairness of the access is impossible to guarantee in a multi-emitter scenario (ad-hoc, mesh, ...) without fine-grained centralized control.

B. Monitoring and Status Report

SDN requires that network elements are able to report their status, and this reporting is one of the key components allowing controllers to take decisions. Besides classical measures such as devices CPU load or available memory, environment assessment in wireless networks essentially consists in two non-trivial aspects: estimating the different wireless channels status (e.g. loads), up to the links characterization (delay, loss rate, stability, etc.) and topology discovery, including close access points identification. The CAPWAP (Control And Provisioning of Wireless Access Points) protocol [3] specifies how to acquire statistics on the communication between wireless stations and access points. At layer 2, the IEEE 802.11v extension has been included in the standard in 2011 and includes some considerations about channel measurement and topology discovery.

The channels load is quite difficult to estimate, as the wireless channel state changes frequently, especially in an indoor scenario where fading, shadowing or multiple paths affect links quality. A channel status can be affected by close access points transmissions, but also by small events in the environment such as people passing, doors closing, etc. Not all channel variations are relevant for network management and short-term variations can be smoothed using various filters, which need to be properly tuned. Too few smoothing does not remove random channel variations and potentially induces a lot of status repots. Too much smoothing hides or delays the detection of more permanent changes in the environment.

As transmissions quality is greatly affected by congestion and interferences, identifying the local topology surrounding a node is important. An access point may be interested in knowing its neighbors identities, their operational channel(s) or their output power, e.g.. Knowing an access point's 1-hop neighborhood is relatively easy and can rely on classical beacons. However, interferers may be located outside the transmission range. Multihop topology discovery techniques usually rely on broadcasting hello packets that include one-hop neighborhood description, which allows nodes to know their 2-hop neighborhood. In a network of access points, the network density may be insufficient to identify all interferers and it may be necessary to implicate the terminals. Adding links characteristics and quality estimation to topology discovery further complicates the problem.

C. Handoffs

A fully functional implementation of SDN should be able to manage multiple handoff situations. Users may migrate from access point to access point when moving, but also for load balancing or topology control. SDN also eases multihoming, mobiles also may have to switch technologies, passing from Wi-Fi to 3G, e.g.. Both situations are fairly well-known and, for example, amendments such as IEEE 802.11f, 802.11k and 802.11r deal with homogeneous handover situations and 802.21 deals with heterogeneous handovers. Such handovers can be realized with almost no service interruption if all cases but a sudden loss of connectivity.

However, SDN introduce a new type of handoff that is trickier: seamless handoff between service providers. Ideally, a user should be able to attach to any access point and IP considerations as well as rerouting or duplication of the traffic should be performed within the network. Mobile IP has proposed solutions to this problem a few years ago, SDN could provide a good support for mobile IP implementations, as cross-operators handover could use in-network duplication of data frames so that information reaches the current mobile location and all its potential destinations. If no off-the-shelf implementation exists today, this scenario demonstrates that even if SDN introduce specific issues, especially at layer 2, they may also help upper layers.

IV. Wireless SDN opportunities

SDN have received a great support in the wired world and numerous examples show that the flexibility it brings enables numerous innovative applications that also exist in wireless. However, the wireless world could benefit from the SDN for its specific needs too.

A. Improving end-user connectivity and QoS

Users of wireless technologies often have to deal with unpredictable quality of service, for example because they are located at the limit of their base station coverage, or because the channel they selected is loaded. Nevertheless there often exists a better access point, not too far away, that operates on a moderately loaded channel, but that belongs to another operator.

In an SDN-enabled network, the multiple controllers in a given area communicated together can exchange enough information to allow users to connect to any access point around, regardless of the operator it belongs to. An access point receiving an incoming packet with a source IP address that belongs to a partner's network could forward it to the home network with a single rule. If load balancing, security and access control problems still need to be addressed, such a cooperation between operators in an area would greatly improve users QoS.

Moreover, a controller could be defined in each geographic area, gathering statistics on the wireless channels utilization from all the access points, regardless of their possessor, and send channel load statistics to the terminal who could, in turn, select which access point to connect to, or even which wireless interface to choose for a given application. Indeed, most mobile terminals are equipped with various wireless interfaces such as Wi-Fi, 3G or Bluetooth, which all have their own characteristics in terms of delay, stability, throughput and coverage. Without accurate statistics, a terminal has a hard time choosing automatically between these technologies and simple algorithms are often implemented (e.g. always prefer Wi-Fi over cellular). Nevertheless, with appropriate statistics sent by multiple controllers, a terminal could make more clever choices per application, as evoked by [9].

B. Multi-network planning

The network also can benefit from using SDN, especially when it operates over a crowded channel. If we take back the Wi-Fi example, there are only 3 non-overlapping channels in the 2.4 GHz ISM band. In dense urban areas, it is common to see tens of Wi-Fi networks at a given place, all interfering together. Network planning is not possible, as the access points locations are uncontrolled by any operator. If IEEE 802.11f (IAPP) is a first step towards this goal, recognizing at least that collaboration between close access points is fruitful.

SDN allows going one step further in the access points cooperation. It allows to create zone-specific controllers that transcend operators, suggesting channel selections and power control to participating access points, based on sensed mutual interferences levels. Channel allocation is a well-known problem that falls back to coloring the graph formed by modeling the access points interdependencies. k-coloring is known to belong to NP-hard and localized algorithms require at least neighbor nodes to be able to exchange color information.

Power control, on the other hand, helps reducing interferences by attaching users to the closest access point and by reducing the transmission power of both parties. The more access points are available, the more efficient this process will be, and the capacity of wireless SDN to aggregate in a single virtual network multiple base stations belonging to multiple ISPs eases the problem.

Access points usually dispose of a limited number of transmission power levels. Let us consider a set of access points distributed across an area. SDN makes it possible, for each access point, to easily identify the neighbors reachable at various transmission levels simply by sending beacons. All receiving access points can forward the beacons to an area controller that can correlate this information with the transmission power of the emitter. Once all the possible transmission powers have been tested, the controller can create locally all the possible interferences graphs and select the one that maximizes coverage while minimizing interferences. If each of the n access point disposes of k power levels, a communication round requires $n \cdot k$ messages emissions that can be performed in parallel. The controller needs to compare k^n situations but can use branch and bound strategies to reduce complexity.

Within the network, operators also share communication links. For example, when facing a sudden capacity increase, an ISP can offload traffic to another collaborating ISP's network or to another technology. Mobile operators sometimes offload traffic from a 3G network to a partner Wi-Fi network when in range, but fail to preserve connections across technologies. With SDN, a controller provoke packets duplication within the core so that they are transmitted to the mobile using two networks with a simple rule update. Such soft handover across technologies has been demonstrated in the OpenRoads project using Wi-Fi and WiMax.

Aside from the handovers, if we look at the effect on the network itself, it is possible to build on the experience of power grids, where offloading is a common

practice. In power grids, lack of coordination when offloading has been the cause of severe issues such as the Nov. 2006 blackout in Western Europe.If such a consequence is unlikely in communication networks, offloading without coordination could easily increase congestion. The presence of a cross-operators controller monitoring network status and triggering offloading accordingly should prevent most situations.

C. Security

In wired as well as in wireless networks, the monitoring capacity of SDN can provide a clear vision of a network status to an entity in charge of detecting intrusions or abnormal behavior. The network load and the packets distribution per protocol can be compared to statistics and a process could decide if the current traffic matches the expected values for this date and time. Suspicious situations may indicate an intrusion or the presence of inside computers participating to a Botnet.

In wireless networks, detecting attacks such as the presence of a rogue access point require cooperation between access points that is facilitated by SDN monitoring and control capacities. Similarly, in mesh or collaborative networks, sharing status information allows detecting misbehaving or selfish users. Reaction can be uploaded in the form of a simple rule by the controller in all managed nodes.

D. Localization

Users localization has become a key function for several location-aware services. The experience of smartphones opportunistic localization has shown that a terminal looking at its wireless environment could achieve fair localization accuracy by looking at a database containing access points locations, even in an indoor situation where GPS is blind. If a terminal can easily locate itself in a multi-network operators infrastructure, the infrastructure has a harder time locating users and this is where SDN can help. A controller collecting information from several access points provides the infrastructure enough information to obtain a coarse localization of the user, sufficient for example to predict handovers and to offer localized services.

V. Conclusion

In this article, we presented challenges and opportunities behind the adaptation of the SDN paradigm to the wireless context. For illustration, we used several examples involving wireless LANs and IEEE 802.11, however the major issues such as data channels separation, slicing, channel estimation, topology discovery and interferences management are not specific to a given technology. Nevertheless solving these issues, even partially, should allow a better management of the wireless spectrum, to enhance QoS offered to users and even provide new tools to security analysts.

Several problems were left out of this paper however and the non-technical problems may be the harder to solve, though. Letting operators or ISPs collaborate and exchange status information would help but is not straightforward when considering commercial issues. Legislation often requires an ISP to be able to identify users upon request, which means that logging volume will increase drastically. From the user's point of view, SDN facilitate enables transparent roaming between operators, they may also introduces serious privacy issues. In roaming situations, an operator can acquire information on people localization and traffic, even if it has no contract with the user and, subsequently, if the user never agreed explicitly to transmit personal data to this operator. It poses, more generally, the question of the user involvement in the decision process.

Acknowledgement

This work was supported in part by COST Action IC0905 TERRA "Techno-Economic Regulatory framework for Radio spectrum Access for Cognitive Radio/Software Defined Radio" [1].

References

[1] http://www.cost-terra.org.
[2] http://www.noxrepo.org/.
[3] Control and provisioning of wireless access points (capwap) protocol specification. RFC 5415, Mar. 2009.
[4] A. Bianco, R. Birke, L. Giraudo, and M. Palacin. Openflow switching: Data plane performance. In *IEEE International Conference on Communications (ICC)*, may 2010.
[5] P. Dely, A. Kassler, and N. Bayer. Openflow for wireless mesh networks. In *20th International Conference on Computer Communications and Networks (ICCCN)*, Aug. 2011.
[6] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker. Nox: towards an operating system for networks. *SIGCOMM Computer Communication Review*, 38(3), July 2008.
[7] M. Jarschel, S. Oechsner, D. Schlosser, R. Pries, S. Goll, and P. Tran-Gia. Modeling and performance evaluation of an openflow architecture. In *23rd International Teletraffic Congress (ITC)*, sept. 2011.
[8] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. Openflow: enabling innovation in campus networks. *SIGCOMM Comput. Commun. Rev.*, 38(2), Mar. 2008.
[9] K.-K. Yap, S. Katti, G. Parulkar, and N. McKeown. Delivering capacity for the mobile internet by stitching together networks. In *ACM workshop on Wireless of the students, by the students, for the students*, 2010.
[10] K.-K. Yap, M. Kobayashi, D. Underhill, S. Seetharaman, P. Kazemian, and N. McKeown. The stanford openroads deployment. In *4th ACM international workshop on Experimental evaluation and characterization (WINTECH '09)*, Beijing, China, Sept. 2009.