Software-Defined Mobile Cloud: Architecture, Services and Use Cases

Ian Ku, You Lu, Mario Gerla
Department of Computer Science
University of California Los Angeles
Los Angeles, USA
{ianku, youlu, gerla} @cs.ucla.edu

Abstract—**Software-Defined Networking (SDN) is an emerging technology which brings flexibility and programmability to networks and introduces new services and features. However, most SDN architectures have been designed for wired infrastructures, especially in the data center space, and primary trends for wireless and mobile SDN are on the access network and the wireless backhaul. In this paper, we propose several designs for SDN-based Mobile Cloud architectures, focusing on Ad hoc networks. We present the required core components to build SDN-based Mobile Cloud, including variations that are required to accommodate different wireless environments, such as mobility and unreliable wireless link conditions. We also introduce several instances of the proposed architectures based on frequency selection of wireless transmission that are designed around different use cases of SDN-based Mobile Cloud. We demonstrate the feasibility of our architecture by implementing SDN-based routing in the mobile cloud and comparing it with traditional Mobile Ad Hoc Network (MANET) routing. The feasibility of our architecture is shown by achieving high packet delivery ratio with acceptable overhead.**

Keywords-Software-Defined Neworking; Mobile Cloud; Ad hoc Networks; Architecture Design; Simulation

I. INTRODUCTION AND RELATED WORK

Software-Defined Networking (SDN) [1] has emerged as a flexible way to control the network in a systematic way, with OpenFlow [2] as the most commonly used SDN protocol. Although the core idea of SDN, which is the separation of the control plane and data plane, is not a complete new idea (other similar proposals predate it, such as ForCES [3] and SoftRouter [4]), advancements in computing and network capacity recently made it feasible and attractive to re-explore such separation.

There have been many deployments of SDN-based architectures and systems in recent years. The Open Networking Foundation (ONF) [7] was founded in 2011 to promote the use and deployment of SDN and OpenFlow-based networks. Founding members include Deutsche Telekom, Facebook, Google, Microsoft, Verizon, and Yahoo. Currently, the ONF has more than 95 members. However, most interest of SDN has been focused on the wired domain, especially in data center deployments, such as deployments by Google and VMware.

There are many potential benefits of SDN, many of which are already demonstrated or under study in wired-SDN systems, which can also be used in a wireless SDN system. Examples include content-aware routing, where routing is not based on only source and destination, and; traffic differentiation. Client side techniques such as WAN optimization and adaptive streaming can create potential unfairness if all traffic is treated equally. With the introduction of SDN, flow-based prioritization allows traffic to be treated differently at the forwarding plane to deliver the required QoS while maintaining fairness.

Another motivation for SDN-based wireless networks is the expected growth of mobile data traffic. In order to keep up with traffic growth, mobile networks must not only optimize the current resources but also add new components/technologies that increase the capacity. Having witnessed the phenomenal burst of research in cloud computing, the idea of mobile cloud is bringing together Mobile Cloud Computing (MCC) and wireless networks to interconnect mobile devices and become a cloud-like service provider. One example would be mobile cloud Intelligent Transport Services [19]. Mobile cloud will also stimulate research beyond the scopes of traditional Internet Clouds or Mobile Computing technologies. Enabling SDN in wireless networks can bring the programmability and flexibility that is lacking in today's distributed wireless substrate while simplifying network management and enabling new services.

While mobile and wireless deployment of SDN has recently, begun, its scope has been primarily focused on carrier backbones and access networks. OpenRoads [5] envisions that users will move between wireless infrastructures. CloudMAC [6] proposes virtualized access points. The Wireless & Mobile Working Group (WMWG) [8] in ONF focuses on wireless backhaul, cellular Evolved Packet Core (EPC), and unified access and management across enterprise wireless and fixed networks (e.g., campus Wi-Fi).

Other works on wireless SDN include OpenFlow in wireless mesh environments [9], OpenFlow in smartphone as an application [10], OpenFlow in wireless sensor networks [11], SDN in heterogeneous networked environments [12], and SDN for handover management in heterogeneous networks [21]. However, there remain many possibilities and challenges that have not yet been fully addressed. The goal of this paper is to explore these potentials, especially in the direction of building a SDN-based Mobile Cloud. In this paper, we focus on SDN in Mobile Ad Hoc Networks (MANETs) as potential wireless infrastructures that can support the SDN-based Mobile Cloud.

978-1-4799-0959-9/14/$31.00 ©2014 IEEE

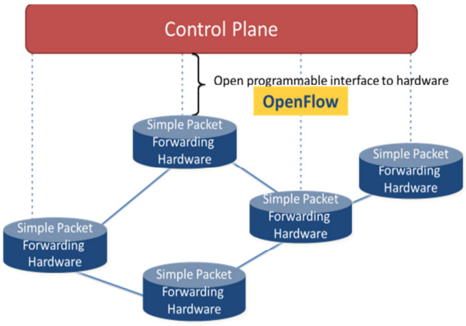


Figure 1. Software Defined Networking Concept

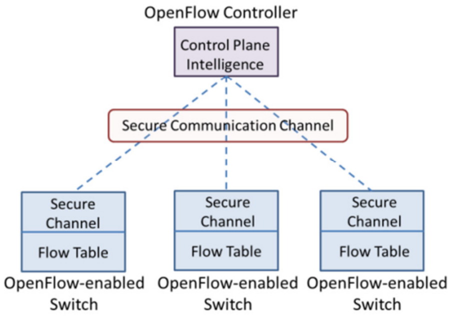


Figure 2. OpenFlow components

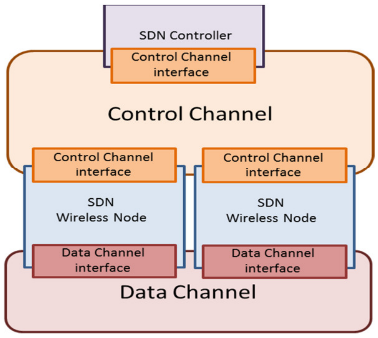


Figure 3. SDN-based mobile cloud overview

There are several challenges to build the Mobile Cloud on top of a MANET while applying SDN as follow: 1.) The SDN Controller-switch connection is no longer wired, instead it must be supported by the wireless medium. This introduces unreliability; 2.) The introduction of node mobility will add complexity. Both of these factors will require mechanisms not needed in wired-based SDNs.

In this paper, we propose two main instances of the SDN-based Mobile Cloud architecture in MANET environments. We present architecture and component designs, including an advanced architecture that is based on frequency selection that allows for many types of new services using a wireless SDN-based Mobile Cloud. The goal is to provide a framework for SDN-based Mobile Cloud that includes the components required by SDN in an Ad hoc environment. We discuss various use cases and services that require enhanced delivery ratio and latency for prioritized traffic (e.g., medical traffic or emergency service traffic).

We evaluate our architectures by using simulation. We implement SDN-based routing to verify the feasibility of our architecture design. We compare our system with traditional MANET routing to determine how SDN can help overcome unreliable wireless links and mobility in the SDN-based Mobile Cloud.

The rest of the paper is structured as follows. In Section II, we provide some background on SDN/OpenFlow. We then describe our architectures and component design for building an SDN-based Mobile Cloud, and discuss use cases. We describe an advanced SDN-based Mobile Cloud architecture based on frequency selection and its use case in Section III. We present our simulation and evaluation in Section IV, and conclude our paper in Section V.

II. Architecture and Conponents Design

A. *Background of SDN/OpenFlow*

The core concept of SDN [1] is the separation of the control plane and the data plane. SDN-based networks contain a control plane network which is used for control traffic, and a data plane network which is used for forwarding data.

OpenFlow [2] is the protocol for communication between the SDN control plane and data plane. Fig. 1 shows a high level concept of SDN. In our paper we extended OpenFlow for adapting it to Ad hoc wireless environments.

In OpenFlow, there are three main components, namely the controller, OpenFlow-enabled switches, and the secure communication channel between them. Fig. 2 shows an example OpenFlow network. The controller and OpenFlow enabled-switches use the secure channel to communication using the OpenFlow protocol. Although the latest protocol version is OpenFlow 1.4.0 [15], OpenFlow 1.0.0 [16] remains the most widely used specification and is the baseline spec version we use in our system.

B. *Design Overview*

The proposed architecture extends OpenFlow to operate in wireless mobile Ad hoc scenarios. In our architecture, we choose to use different wireless technologies for control and forwarding plane: Long range wireless connection i.e., LTE for control plane, and high bandwidth wireless connection i.e., Wi-Fi for data plane. We choose this wireless network configuration to evaluate the feasibility of a heterogeneous wireless solution for SDN in the Mobile Cloud. Another more practical reason is that in MANETs and VANETs not all nodes are easily reachable from the Infrastructure via WiFi Access Points. Since the inherent problem of wireless channel is its reliability/availability, there are always potential communication losses between mobile nodes. We account for this and propose architecture for various wireless environments. Fig 3 shows the overview of the components for our SDN-based mobile cloud.

C. *SDN Wireless Node*

One of the core components of SDN-based Mobile Cloud is the SDN wireless node, which is roughly equivalent to the OpenFlow-enabled switch in traditional OpenFlow networks. Fig. 4 shows the internal architecture design of a SDN wireless node.

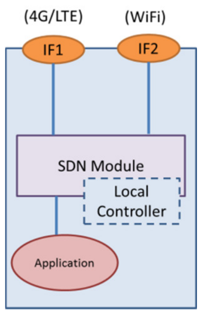


Figure 4. SDN Wireless Node

Each SDN wireless node has an optional local SDN Controller. This local SDN controller can either be the backup controller when connection to the global SDN controller is lost or the primary SDN controller when wireless communication to a global controller is not practical. Traditional Ad hoc routing protocols (e.g., AODV, DSDV, or OLSR) are supported as fallback mechanisms, to allow the SDN network to revert back to Ad hoc network operation even in the case where SDN controller communication is unavailable. In scenarios where the connection to a global SDN controller is stable, this device SDN controller can be removed.

One distinct characteristic of Ad hoc networks is that the nodes act both as Hosts (sending/receiving traffic) and Routers (forwarding traffic on behalf of other nodes). An SDN wireless node is therefore both an SDN data plane forwarding element and an end-point for data. Traffic from any wireless node will run through its own SDN module before being sent, which allows the SDN controller to determine the access of user traffic into the network.

D. Global SDN Controller

Different from the device SDN controller, which is used more as a backup in case of loss of communication, the global SDN controller is the primary intelligence in an SDN-based Mobile Cloud. The global SDN controller is responsible for populating the flow tables of the SDN wireless nodes with flow rules to control how traffic is moving in the network. There are two instances of the architecture depending on the connection conditions between this global SDN controller and SDN wireless nodes:

- *Constantly Connected Global SDN Controller*: a stable connection is maintained between global SDN controller and the SDN wireless node. An example would be the LTE control channel in an urban environment. This controller-node communication is similar to that of a wired SDN system, where flow rules are inserted reactively or proactively based on policy.
- *Intermittently Connected Global SDN Controller*: the connection between the global SDN controller and SDN wireless nodes is intermittent. It is assumed that a SDN wireless node will not always be able to establish a connection to the global controller. Flow rules are pushed by the global SDN controller during periods where connectivity is established; they are enforced by the local SDN controller on the wireless node with knowledge on how to treat traffic based on policy, or; they are created by a combination of the two above methods.

E. System Operations Overview

Fig. 5 shows the System operations overview on how an SDN-based Mobile Cloud will operate.

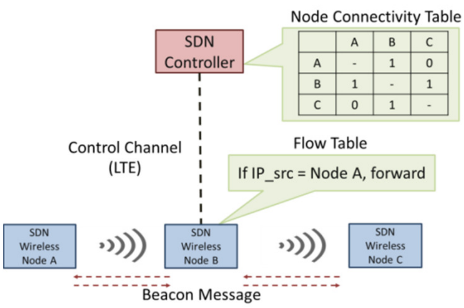


Figure 5. System Operations Overview

We use a modified OpenFlow protocol with some additional components added to make the design viable under Ad hoc network environments. However, The basic operation is the same, each SDN wireless node, when deciding how to handle traffic, will base the decision on flow rules inserted by the SDN controller. If a packet arrives that is not matched in the flow table, an OpenFlow-like behavior is invoked, and the SDN controller can insert new flow rules in corresponding flow table.

One difference compared to standard OpenFlow is that each SDN wireless node will also be exchanging beacon messages to learn about immediate neighbors, a common feature in MANET systems. In our evaluation we use this information to build a node connectivity graph, which is then used by the SDN controller to make intelligent decisions on choosing paths to route packets throughout the network. This allows for good response to mobility in MANET/VANET scenarios, as we will show in our evaluation. We choose to use this beacon method over Link Layer Discovery Protocol (LLDP) used in wired SDN systems.

F. Use Cases

We present Flow-based prioritization use cases where our design of SDN-based Mobile Cloud performs better than traditional SDN architectures. The SDN controller maintains a node connectivity table and has global knowledge of what traffic flows are more important (e.g., emergency messages). Thus, it can selectively prioritize traffic for better performance.

- *Global map-based Path selection:* Based on the node connectivity graph, the SDN controller has a better view on paths from any source to any destination. With this information, the SDN controller has several options. As standard functionality, it allows all traffic to be routed based on a common algorithm (e.g., any flavor of shortest paths). Alternately the SDN controller can decide paths based on the importance of flows. This means that less important traffic can be actively routed using a less optimal path. Benefits of doing so include avoiding contention with traffic classified as more important.
- *Multipath transmission:* Based on a global connectivity graph, the SDN controller allows certain traffic classes to be actively transmitted from source to destination using multiple paths. This increases the possibility of packets reaching the destination, with the trade-off of adding redundant packets in the network.
- *User preference feedback:* Users are given limited control on how to utilize given resources. For example, users can decide to increase their email traffic priority; the SDN controller can then take that requirement and use it as a guideline when setting up flow table rules.

III. SDN-BASED FREQUENCY SELECTION ARCHITECTURE

In this section we describe an instance of the SDN based Mobile Cloud architecture where the SDN controller selects frequencies for wireless node transmissions. In wired SDN/OpenFlow, the action field of a flow table rule is usually the output port. When moving SDN to MANET, the output port becomes the specific wireless interface and the specific frequency within that interface. A wireless node typically has multiple wireless interfaces; moreover the channel frequency of each interface (say, WiFi or LTE) is software reconfigurable.

We propose two alternative architectures using SDN-based frequency selection, based on the number of wireless interfaces for each wireless node, and on their capability.

A. Multiple Wireless Interface Wireless Nodes

Fig. 6-a shows the operation on a wireless node that has multiple wireless interfaces that can be used for data plane.

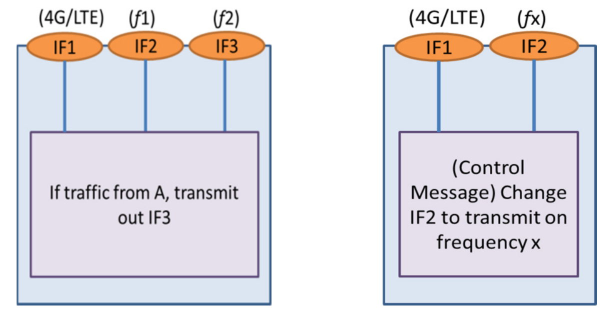
In this setup, wireless interface on nodes are preconfigured to specific frequencies. The SDN controller does not directly control frequency, but instead chooses appropriate interface for data to transmit. The SDN controller knows which radio is using which frequency, and chooses the appropriate one for each flow according to its policy.

B. *Configurable Wireless Interface Wireless Node*

Fig. 6-b shows the operation on a wireless node where wireless interfaces can be configured to transmit on a different frequency.

In this setup, the SDN controller can directly control the transmission frequency of a wireless interface. This flexibility is provided by a more advanced architecture which requires radio frequency to be part of the SDN control message component, as shown in Fig. 6-b. The radios themselves must also be able to accept external commands to change the frequency, such as cognitive radios [13, 14].

If advanced cognitive radios are used as the wireless interface to transmit data, the SDN controller then must gather information from the radios to coordinate spectrum management in order to make decisions.



(a) Multi-Interface Wireless Node   (b) Configurable Wireless Interface Wireless Node

Figure 6. Frequency selection-based architectures

C. *Use cases for SDN-based frequency selection*

There are many potential use cases of using SDN-based frequency selection in a wireless situation; here we describe some of the potential applications:

- *Wireless Network Virtualization:* We can let different flows from different networks choose different radios/interfaces using different frequencies. If the radio frequencies used by each individual network is different, individual network's traffic are isolated from each other and we have thus effectively sliced the networks and created virtual wireless networks. The control of which network uses which radio interface/frequency for which time period is done by the SDN controller, which makes the allocation of network traffic a programmable fashion. Time slicing for efficient OFDM spectrum allocation used for LTE networks can be applied in the SDN Mobile Cloud to support one virtual wireless network per time slot. If multiple radio interfaces are available, multiple virtual networks can be supported in the same time slot. For instance, in one time slot, video content may be streamed from the Internet to the mobiles via the LTE interfaces; ITS traffic is exchanged on WiFi f_1 channel; P2P MPEG DASH video is transmitted on f_2. Note that the video packet broadcast on channel f_2 is picked up by all neighbors tuned on f_2. The node that will receive and forward the video packet is determined by SDN controller intelligence. Additionally, the SDN controller can set filters on node inputs so that some nodes, say, may reject certain traffic classes. This could be used, for example, to restrict the propagation of video surveillance traffic to law enforcement vehicles. This input filtering is an SDN feature unique of wireless networks where broadcast is used.

- *Privilege Traffic Reservation:* We can reserve or limit specific frequencies so that emergency, security, or otherwise privileged traffic uses this reserved path. The difference between this and traditional emergency channels is that reservation in our architecture is configurable dynamically. The SDN controller can assign flows to these channels or remove them based on current traffic conditions. This can also be used to offer different level of services based on policy. The way this can be done is by changing rules during the emergency period. Emergency traffic gets priority over the remaining traffic.

- *Frequency Hopping:* We can protect a flow from jamming or eavesdropping by putting it on a frequency hopped mode, where SDN wireless nodes are coordinated and hop across channels according to a prescribed sequence in the wireless virtual network defined by the time slot schedule.

IV. Simulation and Evaluation

In this section we describe our simulation setups, configurations, and results. We model the architecture using the NS-3 simulator [17]. The goal is to demonstrate the ability to implement services in an SDN-based mobile cloud. In our evaluation, we implemented SDN routing. Using node connectivity information, the SDN controller uploads flow table rules that deliver the required routing functionality. In our study we use a proactive flow table entry insertion model, assuming that the SDN controller knows the topology and has the knowledge of which channels and to which neighbor each packet should be transmitted.

A. *Simulation Setup*

The simulation is performed over a SUMO [18] generated road network that spans an area of 1000 x 1000 m^2. The road network is a grid type network with each road segment = 200m. Node density varies from 30 to 50 nodes in the simulations. The SDN controller LTE access is placed in the center of the simulation area where it is in wireless range of all SDN wireless nodes. Each SDN wireless node has multiple wireless interfaces; short range using 802.11with the Friis propagation loss model to limit the transmission range to 250m, and long range using LTE. In this study we assume a single frequency for each channel. Each simulation run features a pair of random nodes in the topology running a NS3 echo client-server streaming session, with a packet generation rate of 4 packets/s and packet size of 1024 byte. Beacon message interval is 500ms. SDN wireless nodes will update neighbor information to the SDN controller at intervals of 1s. Simulation parameters were chosen based on MANET comparison studies [20]. Each set of simulations is averaged over 10 runs each running for 5 minutes.

B. Evaluation

We first evaluate our system under different mobility scenarios. Fig 7 shows the packet delivery ratio under different node speeds.

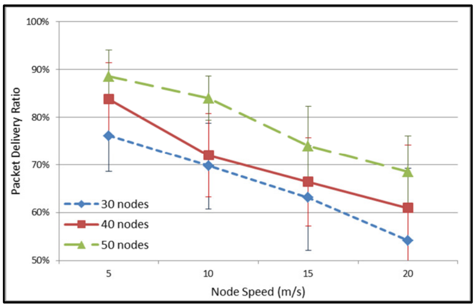


Figure 7. PDR under different node speed

We can see that packet delivery ratio drops from both the increase in node mobility and decrease in total node count. This is expected as routing will fail when there is no path between sender and receiver, and both of the factors will increase the chance of not finding a valid path.

To verify the feasibility of our system, we compare performance with more traditional Ad hoc routing protocols. Figure 8 shows the results. The scenario we use is the 50 node scenario.

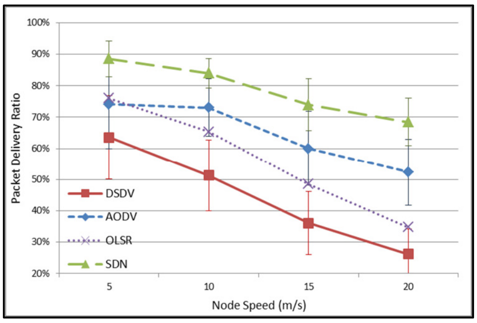


Figure 8. PDR comparison: SDN vs Traditional Ad hoc routing

We can see that our SDN-based routing outperforms the other traditional Ad hoc routing protocols. The major reason is that our SDN-based system responds much faster to topology change. As SDN wireless nodes update the SDN controller about neighbor information, the SDN controller immediately detects that there is topology change and sends out control messages as needed.

The fast response to topology change requires control messages between SDN controller and SDN wireless nodes. We evaluated the amount of overhead caused by this exchange. More precisely, we account for the SDN wireless nodes sending neighbor information to the SDN controller, and SDN sending rules to SDN wireless nodes. We measure the amount of control traffic from both and compare in Fig 9.

We can see that the majority of control traffic is generated by the SDN controller. This is because while SDN wireless nodes send neighbor information to the SDN controller in a constant rate, the SDN neighbor must send out flow table modification messages when node connectivity information changes. Also, we can see that the ratio of control traffic generated by the SDN controller increases as speed increases.

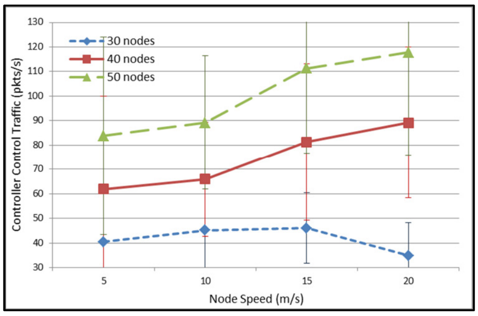
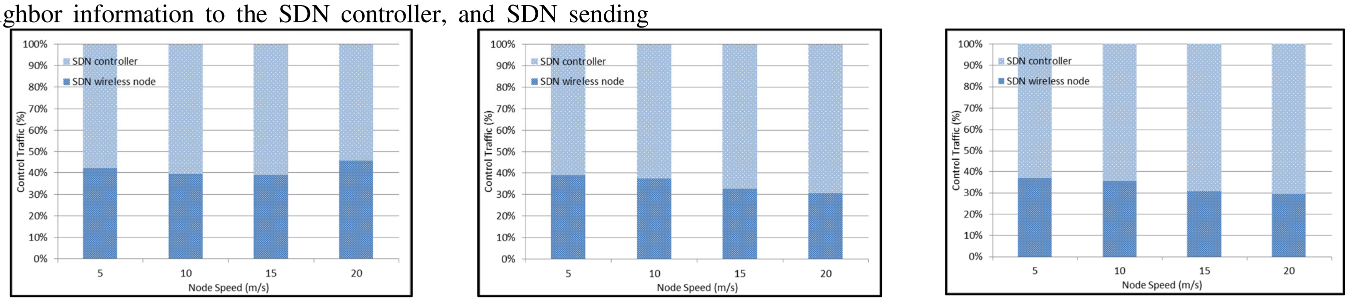


Figure 10. SDN Controller control traffic

Fig. 10 shows the control traffic rate generated by the SDN controller to modify flow rules. First, we can see that as node density increases, the SDN controller needs to handle more SDN wireless nodes and must send out more control messages. Also, we see that the control rate increases with speed because the SDN controller must keep up with topology change. The exception is in the 30 node and 20 m/s case where the number of control sent by the SDN controller decreases. This is happening because with only 30 nodes it is more likely that the path does not exist. No rules can be inserted, and the total control traffic actually decreases. Overall, since control traffic does not carry large payloads, even at 100 packets/s the overhead traffic is still manageable. Nevertheless, as with all centralized control systems, scalability is a concern that should be addressed, the amount of control message will only increase as more SDN wireless nodes are under the SDN controller's control. Also, topology change, which is common in VANETs, will also increase control messages. Methods to reduce these messages for scalability and freshness should therefore be investigated, such as delegated some functionality to local SDN controllers on each individual node.

As we mentioned previously, controller-node communication using the LTE medium is not as stable as wired SDN, we thus investigate the case where there is a SDN controller link failure. We choose a 50 node scenario, as shown in Fig. 11. The dashed line shows the time of SDN controller failure.



(a) 30 Nodes

(b) 40 Nodes

(c) 50 Nodes

Figure 9. Control traffic breakdown

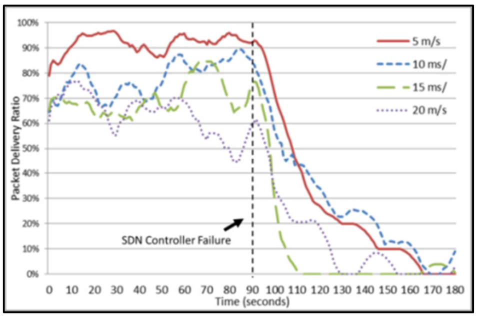


Figure 11. SDN controller failure

We can immediately observe that packet delivery drops as no more rules are inserted to SDN wireless nodes by the SDN controller. Packet delivery entirely relies on the last flow rule being still valid. In wired SDN and low node mobility scenarios, the path will stay valid longer and has less impact on system performance. However, we can see from Fig. 10 that higher mobility will quickly change the topology and make the rule obsolete. Successful packet delivery is then entirely dependent of the next nodes in the last rule being within transmission range. This makes a valid recovery mechanism such as the one provided by a local SDN controller a critical function in wireless SDN and will be an important issue to investigate in future work.

V. Conclusion and Future Work

In this paper, we propose several instances of the SDN-based Mobile Cloud architecture. The architectures capture the components and strategies needed to deploy SDN in a Mobile Cloud. We demonstrate feasibility by building our system in the NS-3 simulator and implementing SDN routing as a Cloud service and compared with traditional MANET routing schemes. Simulation results show that the SDN-based Mobile Cloud routing service can achieve good packet delivery ratio with acceptable overhead.

For future work, there are several directions that we intend to explore. First, we wish to explore the recovery mechanisms when SDN controller is lost, specifically using a local routing fall back we described in our design. Likewise, locally updates rules may be used to scale to large systems. There are also cases of partial controller connectivity loss. For example, a subset of mobile nodes lost communication to the controller. In this case, isolated nodes should form their own SDN cluster. The best choice will depend on node density, mobility pattern, and possibly other factors.

Second, we intend to investigate alternate SDN wireless architectures that we did not include in this study, for example the case where the SDN controller transmits control traffic in P2P mode using WiFi channels. While this allows to build a wireless SDN system that is completely distributed and thus does not need infrastructure support, the communication with the SDN controller can be delayed and even interrupted causing new complications.

Third, we intend to investigate frequency selection use cases. Transmitting traffic on different frequencies and in different virtual subnets will allow us to deploy new services with new market potentials.

References

[1] N. Mckeown, "Software-defined networking." INFOCOM keynote talk, Apr, 2009

[2] 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., vol. 38, no. 2, pp. 69–74, 2008.

[3] A. Doria, J. Hadi Salim, R. Haas, H. Khosravi, W. Wang, L. Dong, R. Gopal, and J. Halpern, "Forwarding and Control Element Separation (ForCES) Protocol Specification", RFC 5810, March 2010

[4] T. V. Lakshman, T. Nandagopal, R. Ramjee, K. Sabnani, and T. Woo, "The SoftRouter Architecture," in Proc. ACM SIGCOMM Workshop on Hot Topics in Networking, 2004.

[5] K.-K. Yap, M. Kobayashi, R. Sherwood, T.-Y. Huang, M. Chan, N. Handigol, and N. McKeown, "OpenRoads: empowering research in mobile networks," SIGCOMM Comput. Commun. Rev., vol. 40, no. 1, pp. 125–126, January 2010.

[6] J. Vestin, P. Dely, A. Kassler, N. Bayer, H. Einsiedler, and C. Peylo, " CloudMAC: towards software defined WLANs," ACM SIGMOBILE Mobile Computing and Communications Review, vol. 16, no. 4, pp. 42–45, 2013.

[7] Open Networking Foundation. [Online]. Available: https://www.opennetworking.org/

[8] Wireless & Mobile Working Group (WMWG). [Online]. Available: https://www.opennetworking.org/images/stories/downloads/working-groups/charter-wireless-mobile.pdf

[9] P. Dely, A. Kassler, and N. Bayer. "Openflow for wireless mesh networks." In Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN), pages 1{6. IEEE, 2011.

[10] P. Baskett, Y. Shang, W. Zeng, and B. Guttersohn. "SDNAN: Software-Defined Networking in Ad hoc Networks of Smartphones", Consumer Communications and Networking Conference (CCNC), 2013 IEEE

[11] T. Luo, H. Tan, and T. Quek, "Sensor openflow: Enabling sofware-defined wireless sensor networks," Communications Letters, IEEE, Vol. 16 , Issue: 11, 2012

[12] M Mendonca, K Obraczka, and T Turletti, "The Case for Software–Defined Networking in Heterogeneous Networked Environments", ACM converence on CoNEXT student workshop, 2012.

[13] P. Pawelczak, R. Venkatesha Prasad, L. Xia, and I. Niemegeers. "Cognitive radio emergency networks-requirements and design." In the First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN), pages 601{606. IEEE, 2005.

[14] I. Akyildiz, W. Lee, and K. Chowdhury. "Crahns: Cognitive radio ad hoc networks." Ad Hoc Networks, 7(5):810{836, 2009.

[15] OpenFlow Switch Specification, Version 1.4.0 (Wire Protocol 0x05). [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.4.0.pdf

[16] OpenFlow Switch Specification, Version 1.0.0 (Wire Protocol 0x01). [Online]. Available: http://www.openflow.org/documents/openflow spec-v1.0.0.pdf

[17] NS-3, http://www.nsnam.org/

[18] Simulation of Urban Mobility (SUMO), http://sumo-sim.org/

[19] M. Gerla. "Vehicular Cloud Computing", VCA 2012 Proceedings, Cyprus, June 2012

[20] J. Broch, D. Maltz, D. Johnson, Y.-C. Hu, and J. Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98), pages 85–97, Dallas, TX, October 1998. ACM

[21] C. Guimarães, D. Corujo, R. L. Aguiar, F. Silva, and P. Rosa, "Empowering Software Defined Wireless Networks Through Media Independent Handover Management", Proc. 2013 IEEE Globecom, Atlanta, USA, Dec 2013