The Case for Software-Defined Networking in Heterogeneous Networked Environments

Marc Mendonca, Katia Obraczka, Thierry Turletti

The Case for Software–Defined Networking in Heterogeneous Networked Environments

Marc Mendonca
University of California
Santa Cruz, USA
msm@soe.ucsc.edu

Katia Obraczka
University of California
Santa Cruz, USA
katia@soe.ucsc.edu

Thierry Turletti
INRIA
Sophia Antipolis, France
thierry.turletti@inria.fr

ABSTRACT

In this paper, motivated by the vision that future internets will comprise infrastructure–based and infrastructure–less networks, we explore the use of the Software–Defined Networking (SDN) paradigm in these so–called "heterogeneous" networked environments. To make the case for SDN in heterogeneous networks, or Heterogeneous SDN (H–SDN), we examine application scenarios in which H–SDN is a key enabling technology. We also identify the additional features and requirements imposed by the H–SDN paradigm and discuss the research challenges they raised.

1. INTRODUCTION

A critical enabling technology for future network services is support for a heterogeneous internet, which interconnects users and applications across networks ranging from wired, infrastructure-based wireless (e.g. cellular–based networks, wireless mesh networks), to infrastructure–less networks (e.g. mobile ad–hoc networks, vehicular networks). Mobile traffic has been increasing exponentially over the past several years, and is expected to increase 18–fold by 2016, with more mobile–connected devices than the world's population by the end of 2012[2]. As mobile devices with multiple network interfaces become commonplace, self–organizing networks may form to extend the range of infrastructure networks or handle episodic connectivity disruptions. Beyond traditional services, ad hoc networks also enable a variety of new applications such as vehicular communication, community services, healthcare delivery, emergency response, and environmental monitoring, to name a few.

Software–Defined Networking (SDN) has been proposed as a way to programmatically control networks, making it easier to deploy new applications and services, as well as tune network policy and performance. OpenFlow[9] is a notable example of a SDN architecture, based upon "programmable" switches which consist of: (1) a flow table containing an entry for each flow along with an action to be invoked; and (2) a protocol that allows communication between the switch and a controller, a process which typically runs on a remote machine and manages new flow table entries. SDN techniques to–date, such as OpenFlow, largely target infrastructure–based networks, especially those found in data centers.

As mobile devices become predominant, users will demand high–quality service regardless of location or type of network access. Efficient content delivery over wireless access networks will become essential, and self–organizing networks may form a prevalent part of the future hybrid Internet as wireless devices become ubiquitous. The goal of software–defined networking is to enable the rapid "development and delivery of new architectures, standards, software, and applications that will decrease costs, enable new innovation, and increase security, stability, and availability of networks around the globe"[1]. In addition to new applications, SDN deployment in hybrid environments would allow existing infrastructure connectivity and services to be offered to an expanded audience; service providers would be able to maintain policy, if not complete control, over self–organizing networks that form connections with their infrastructure. SDN may allow more efficient content delivery as the network would be able to choose the best interface and source to use for data delivery.

Software–defined networking has the potential to facilitate the deployment of new applications and services with greater efficiency; however, many of the current SDN architectures promote a centralized control mechanism that is ill–suited to the level of decentralization, disruption, and delay that is present in wireless environments.

Motivated by a vision of a fully connected world, we explore how SDN can be utilized to support both infrastructure–based wireless and infrastructure–less networks, as well as the research challenges involved with adapting the current SDN model to these challenging environments. While previous works have examined the use of SDN in wireless environments, their scope has primarily focused on wireless infrastructure deployments (e.g., WiMAX, Wi-Fi access points). For example, the idea of a flexible wireless infrastructure supported by OpenFlow was introduced by the OpenRoads project[14, 13], which envisioned a world in which users

could freely move between wireless infrastructures while also supporting the provider. Other works[4, 5] have examined OpenFlow in wireless mesh environments, but to our knowledge no one has explored the challenges and benefits offered by extending the SDN paradigm to heterogeneous networked environments.

This paper explores the use of software–defined networking in heterogeneous networked environments: In Section 2 we expand on our motivation by examining use cases that would benefit from the SDN paradigm. In Section 3, we identify the requirements of SDN in heterogeneous networked environments, and how they differ from current approaches. Finally, we consider some of the research challenges involved with implementing a H–SDN architecture in Section 4.

2. USE CASES

The hybrid networks examined in the following use cases consist of mobile units that have limited or intermittent connection to infrastructure (e.g. wired, cellular or 802.11 access points), but are able to form ad-hoc connections with other nearby units. Additionally, some of the mobile units have multiple network interfaces (e.g. wired/802.11 or 802.11/cellular). In such environment, users may desire communication or content from two primary sources:

- Internet content; and data from the "cloud". This would be the case for traditional web applications.
- Local content; and data generated within the immediate area. This would probably be the case for emerging applications such as urban sensing, emergency ops, P2P gaming, etc.

For each use case, we will examine two scenarios: a 'traditional' case, and one in which the network is enabled with SDN technology. In the SDN use case scenario, we will assume that the mobile units have agreed to some form of external control insofar as routing decisions are concerned; although this raises several issues (which we will discuss in Sections 3 and 4), we identify in each of the cases the possible benefits to both the users and any infrastructure providers.

Internet content.

First, we examine several use cases that desire content or communication from Internet locations. For the following use cases, we assume a network topology similar to that of Figure 1. In the base case, a user "Alice" wishes to connect to the Internet and access the World Wide Web; unfortunately, she is unable to connect to infrastructure so she joins an ad hoc network instead. Another user "Bob" is connected to both the ad hoc network and a mobile data network.

- Traditional: assuming the ad hoc network learns to route to Bob as a gateway, and Bob allows his device to be used as a NAT router by strangers, the mobile data service provider is not aware of the existence of Alice. Bob's connection is not assigned additional bandwidth, possibly harming performance; furthermore, Bob will be held responsible for Alice's traffic by the service provider for any possible data overages or illegal activity.

- SDN: The service provider is made aware when Alice joins the ad hoc network; therefore, it may decide to offer service to Alice via Bob and provision Bob's connection accordingly. The service provider may decide to sell Alice a temporary connection plan on the spot, or Alice may have an existing contract on another device; it can also factor in available bandwidth, past user behavior, or any number of factors in deciding whether to offer service to Alice. The service provider is happy, as it maintains control of its network policy while being granted an opportunity for additional business. Alice is happy, as she is able to quickly connect to the Internet using her existing service plan. Bob is happy, as he may be offered incentives by the service provider, while avoiding performance loss or being held liable for Alice's traffic.

As an addendum to the base case, we look at a situation with multiple gateways. Shortly after Alice joins, a user "Charlie" with access to wired infrastructure also connects to the ad hoc network. In the traditional scenario, traffic will be routed depending on how the MANET protocol handles multiple gateways. In the SDN scenario, traffic will be routed based on policy. For example, Alice's traffic may continue to flow through the slower mobile data network instead of the wired network, because she has a service plan with the mobile data provider; alternatively, the mobile data provider may have an agreement with the wired network such that even Bob's traffic will flow through Charlie to either increase Bob's performance or reduce the load on the mobile data network.

Next, we look at examples of service optimizations. In one possible situation, a group of users in the ad hoc network may be viewing the same content simultaneously (e.g., live streaming of a sport event). Using the base case from above, Bob is the link to the Internet from which the content originates. In the traditional scenario, any optimizations such as caches or CDN are performed either in the provider network or in the cloud; leaving Bob's link to the provider saturated with duplicate content. SDN enables routing policies to evolve and promotes the creation of new services; for example, it may be possible to reduce the strain on the limited infrastructure connectivity by caching and retrieving common content locally, or by creating multicast streams on-the-fly for live content.

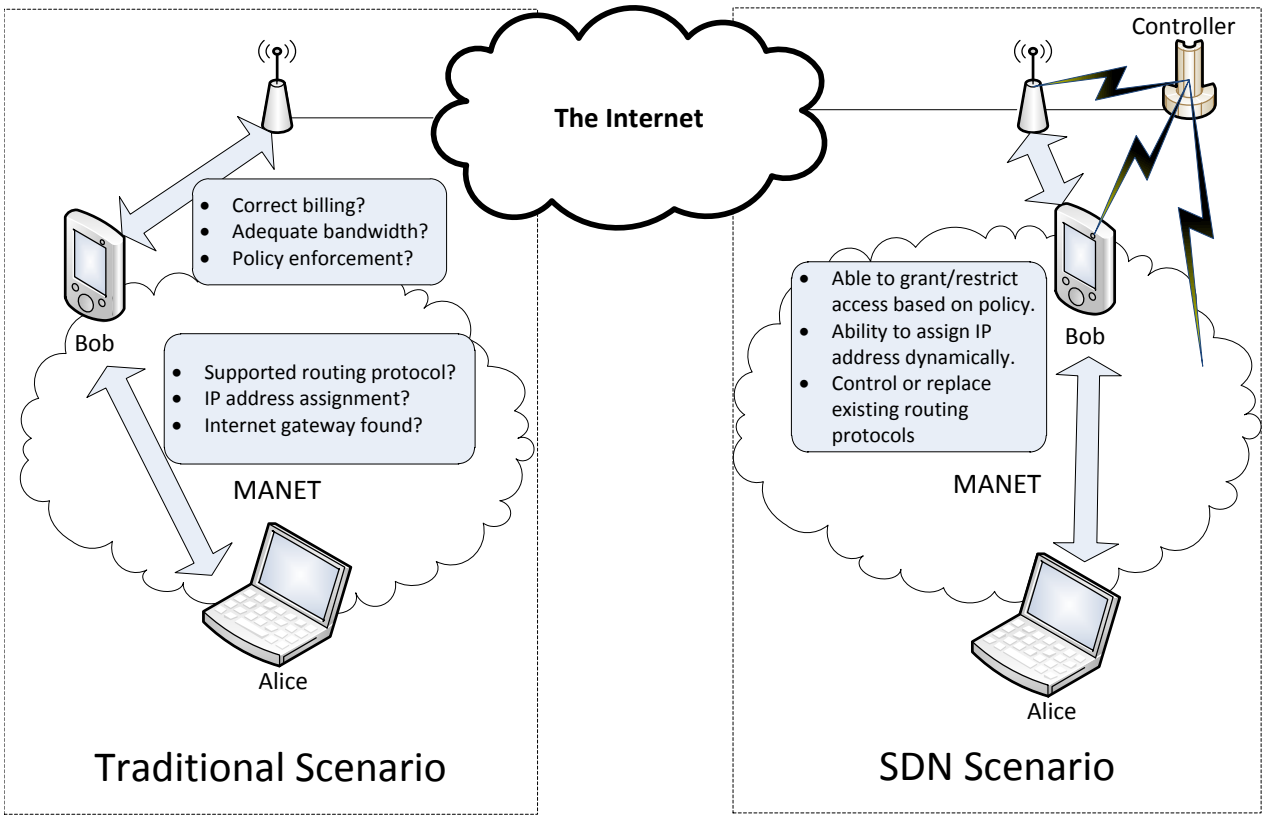

Figure 1: A possible Internet use case topology

Another possibility is support for mobile data offloading, in which other network interfaces are used for delivering packets originally destined for a cellular data network. Traditionally, this tends to be device initiated through periodic scanning for known complementary networks, such as a 802.11 Wi-Fi access point previously visited by the user. With SDN, a mobile provider may be able to remotely initiate offloading for heavy flows based on user proximity to complementary networks. Additionally, it may be able to seamlessly initiate the connection with a secured Wi-Fi or Femtocell network that the user would have otherwise been unable to access.

Local content.

Now we will look at some possible use cases from emerging applications in self-organizing and ad hoc networks. While a connection to the Internet may exist on these networks, the users in the following scenarios are primarily interested in locally generated information.

Emergency Operations: In an emergency, such as a city in the aftermath of a disaster, there may be limited communications connectivity available; moreover, any remaining infrastructure may be overwhelmed as victims, concerned relatives, and responders all vie for the remaining bandwidth. The primary technological challenge is the rapid deployment of communication systems to supplement or replace existing infrastructure. User access to network resources should be constrained to ensure critical communication continues uninterrupted, yet periodic information updates are also important for the emotional stability of the population. In addition, while emergency response organizations are initially faced with a scarcity of information, they are later faced with a deluge of imprecise information that strains both the management system and communication infrastructure[8].

In such an environment, cognitive radio[3, 10] devices may be employed by emergency response workers to maximize spectrum efficiency based on detected conditions and policy rules. They may also be used to bridge a connectivity gap between different communication systems that would otherwise be unable to interact.

- Traditional: As identified by previous work[8], rapid supplemental communication deployment is a complicated issue in areas with pre-existing infrastructure due to interoperability, interference, and user dependence on prior networks. It is also difficult to share and disseminate important information while determining trust and conserving limited resources in an emergency situation. While some of the above issues may be more social than technological, access to a common network is a critical

step towards reliable emergency communications. If cognitive radio devices are employed, it may still be difficult to establish connections between multiple responder organizations due to incompatible policies that were not coordinated a priori[7, 6], such as operating modes, technical control, frequency bands, etc.

- SDN: As discussed above in the Internet use case, SDN can be used to extend the range of existing infrastructure via ad hoc networks; additionally, the network can be programmed with a fallback emergency policy that activates during crises to limit bandwidth consumption while prioritizing emergency and informational traffic. As supplemental disaster communication points are added, they can provide structure to the ad hoc networks that have formed and route emergency requests to an appropriate destination, such as a new mobile command post.
  With cognitive radios, a SDN controller may be used to closely coordinate spectrum management with topology, policy, and routing decisions. As SDN devices are designed to communicate with a controller, policies may be updated on the fly, maximizing agency interoperability.

Other Local Applications: Many other emerging applications (e.g., participatory sensing, peer-to-peer gaming, vehicular communication) are centered around local content or communication with nearby users. Traditionally, many local applications require access to a central server or cloud service to retrieve this information, even if content is ultimately generated by nearby users. SDN may help facilitate local service discovery and P2P communication in an ad hoc network without requiring indirect transmission or lookup through an external host.

3. REQUIREMENTS

In the previous section, we discussed possible use cases in which SDN can be beneficially employed in heterogeneous environments without going into detail about how such a system would operate. While current specifications such as OpenFlow support a number of capabilities, there are also features lacking that would be necessary to enable the scenarios discussed previously. We identify the elements necessary for a heterogeneous SDN, then consider the research challenges involved in Section 4.

End device deployment.

The concept of SDN is centered around controlling forwarding policies, and has been thus far put into practice within the network infrastructure. However, in infrastructure-less environments such as mobile ad hoc networks (MANETs) or vehicular networks (VANETs), the devices involved with forwarding are also end devices themselves. To offer the greatest benefit in these networks, the end devices should be able to communicate with controllers and understand how to handle rules. The deployment should be lightweight in terms of device and network overhead.

Multiple controller domains.

Unlike most infrastructure deployments, a heterogeneous network may span multiple domains of control. As an example from the last section, an ad hoc network may have gateways to two different infrastructure networks. While previous work[11] considered using a transparent proxy to allow multiple controllers, devices in an infrastructure-less network must be able to discover, differentiate between, and support multiple controllers on their own as they may not be able to rely on an outside proxy due to the unpredictable nature of such networks.

Flexible rules and actions.

Current specifications targeted at infrastructure networks often limit the types of matches and actions that can be performed on flows, perhaps due to performance or hardware constraints. Although the latest OpenFlow 1.2 specification adds support for experimenter-specified flow match fields, switches do not have to support this feature and are only required to match a small, pre-defined set of fields. As the expected throughput in a wireless infrastructure-less network is already much lower due to physical constraints, and the specification will be implemented in end device software, the benefit of supporting flexible rules (e.g. flow matching on custom headers) and actions would likely outweigh any drawbacks.

Capability discovery.

While infrastructure devices and links are relatively homogeneous in terms of performance, security, capabilities, and status; the devices and wireless links found in infrastructure-less networks are completely heterogeneous. There are many additional factors to consider, such as power, device policy, wireless interference, and trust to name a few. Clearly, a controller that learns this information would be better equipped to make decisions.

Delay and disruption tolerance.

As the OpenFlow specification centers around a single point of control, it is susceptible to performance and functionality issues in lossy wireless environments. While this may be reduced by proactively partitioning and distributing rules[15], the network should be able to adapt if it is severed from an infrastructure-based

controller.

Hybrid SDN.

Infrastructure-less networks may contain a variety of independently-operated devices, some of which may be unable to communicate with a SDN controller due to reasons of support, capability, or user choice. While this drawback would seem to preclude those devices from participating as a forwarding node in a SDN, it is likely that such a device would be capable of receiving instructions on some other control plane, such as a standard routing protocol.

A hybrid solution should be able to integrate with other protocols to extend functionality and connectivity to non-compatible devices as much as possible. Taking the basic Internet content case from Section 2 with the network topology identified in Figure 1, Alice should still be able to connect to the Internet even if Bob is the only SDN-enabled device in the infrastructure-less network, which may be utilizing a MANET routing protocol such as OLSR.

4. RESEARCH CHALLENGES

The requirements listed in the previous section highlight several outstanding challenges that SDN architectures for heterogeneous networked environments would face.

Security.

Though SDN can be used to strengthen network policy enforcement, the underlying architecture must be secure and maintain basic principles such as confidentiality, integrity, and availability. In an infrastructure-less network with independently-owned end devices also acting as forwarding nodes, it may be difficult to establish trust and ensure a secure channel from end-to-end. As the possible issues range all the way from jamming at the physical layer to worms at the application layer, any solution will likely need to take a multi-layered approach.

Although the problem of security has been explored in the MANET community[12], it is exacerbated by the existence of independent controllers. While a switch in an infrastructure-based network may easily be configured to securely connect to a pre-determined controller, devices and controllers in infrastructure-less networks must discover each other without prior knowledge of the network setup. Furthermore, it is not enough that control messages successfully and securely reach their destination; both endpoints must be able to trust each other to act properly. Forwarding nodes need to be able to trust that the discovered controller is not malicious before accepting control. Likewise, the controller must be able to trust that forwarding nodes that have accepted control are correctly following instructions. For this trust to exist, mechanisms must be in place to ensure the legitimacy of nodes/controllers, the authenticity of the control traffic, and verify that devices act as expected in response to instructions.

End device deployment.

As end devices often act as forwarding nodes in ad hoc networks, they should be able to communicate with the controller, either directly or via a relay, to participate in the SDN. The deployment of an SDN framework on these devices brings up a number of technical issues, including performance and security, as well as some social challenges.

Unlike the core infrastructure, software switching is likely sufficient for the throughput of infrastructure-less networks, but user device capability and availability as forwarders vary and are an important factor. For example, a desktop machine will likely be able to handle more traffic longer than a battery-powered mobile phone. Regardless, it is important to use lightweight software that minimizes device resource usage while maximizing compatibility. In addition, the communication overhead should be limited to preserve the restricted bandwidth often found in infrastructure-less networks.

Incentives are necessary to ensure collaboration between nodes, as forwarding traffic consumes resources such as power, bandwidth, CPU time, and memory. Before a device joins a control domain, it may be necessary for a controller to promise certain benefits, such as connection to services on other devices, reciprocity in the form of network access credits (e.g., time or bandwidth), or monetary compensation.

Network control and flexibility.

Due to the nature of a heterogeneous network, which may span multiple infrastructure/infrastructure-less networks and their associated devices, the question of control is complex. The goal is to maintain the control and flexibility of a system such as OpenFlow, while acknowledging that tradeoffs will be made for an environment with disruptions, diverse device capabilities, multiple domains of control, and independent forwarding devices.

As forwarding devices may be independent end user devices, they have the ability to decide whether to accept instructions upon discovering a new controller, perhaps on the identity of the control authority and/or the user policy on how their device may be used.

Likewise, a controller must decide whether to use a newly discovered device to forward traffic; the decision may be based on criteria such as hardware capabilities, performance, and level of trust. Additionally, a controller may decide to delegate certain rules and authority to other devices based on an eligibility function, as a fully centralized control architecture may suffer in a

lossy environment.

Compatibility.

As previously identified in Section 3, a heterogeneous network may span a variety of devices and networks, not all of which may be able to directly interface with a SDN controller due to reasons of capability or choice. While a controller would not be able to establish direct oversight of non-compatible entities via a SDN control channel, it may still be possible to utilize those entities with a hybrid approach that considers using other control mechanisms, such as standard routing or configuration protocols, to achieve policy objectives. Such a solution should be extensible to the control plane of other protocols, allowing seamless communication, policy enforcement, and information exchange, with and across non-compliant devices and networks to the extent that is possible. While primarily an engineering challenge, the creation of such a framework would extend the control range of a SDN to allow service to be offered to a greater number of users.

5. CONCLUDING REMARKS

As wireless devices rise along with user demand for universal connectivity, software-defined networking may be utilized to extend the range of existing services as well as support emerging applications in infrastructureless networks.

To support such heterogeneous scenarios, software-defined networking solutions would have to go beyond the current infrastructure-based SDN model and address several challenges, including deploying a framework on end devices, handling multiple domains of control, supporting a flexible set of rules and actions, recognizing diverse device capabilities, tolerating delay and disruption, and integrating with other control planes.

The existence of independent forwarding devices and controllers raises other challenges, namely security, compatibility, and maintaining the balance of control and flexibility when devices cooperate to form a SDN. The formulation of a hybrid SDN framework to support heterogeneous networks would allow service providers to expand coverage of existing infrastructure while enabling innovation and interoperability in self-organizing networks.

6. REFERENCES

[1] Open networking foundation. https://www.opennetworking.org/about.

[2] Cisco visual networking index: Global mobile data traffic forecast update, 2011–2016. Technical report, Cisco, February 2012.

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

[4] A. Coyle and H. Nguyen. A frequency control algorithm for a mobile adhoc network. In *Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, November 2010.

[5] 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.

[6] N. Jesuale and B. Eydt. Spectrum paradigm shift. *Radio Resource Mission Critical Communications Magazine*, 23(3):83–91, 2008.

[7] B. Lane. Cognitive radio potential for public safety. http://transition.fcc.gov/pshs/techtopics/techtopic9.html.

[8] B. Manoj and A. Baker. Communication challenges in emergency response. *Communications of the ACM*, 50(3):51–53, 2007.

[9] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. Openflow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2):69–74, 2008.

[10] 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.

[11] R. Sherwood, M. Chan, A. Covington, G. Gibb, M. Flajslik, N. Handigol, T. Huang, P. Kazemian, M. Kobayashi, J. Naous, et al. Carving research slices out of your production networks with openflow. *ACM SIGCOMM Computer Communication Review*, 40(1):129–130, 2010.

[12] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang. Security in mobile ad hoc networks: challenges and solutions. *Wireless Communications, IEEE*, 11(1):38–47, 2004.

[13] K. Yap, M. Kobayashi, R. Sherwood, T. Huang, M. Chan, N. Handigol, and N. McKeown. Openroads: Empowering research in mobile networks. *ACM SIGCOMM Computer Communication Review*, 40(1):125–126, 2010.

[14] K. Yap, R. Sherwood, M. Kobayashi, T. Huang, M. Chan, N. Handigol, N. McKeown, and G. Parulkar. Blueprint for introducing innovation into wireless mobile networks. In *Proceedings of the second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures*, pages 25–32. ACM, 2010.

[15] M. Yu, J. Rexford, M. Freedman, and J. Wang. Scalable flow-based networking with difane. In *ACM SIGCOMM Computer Communication Review*, volume 41, pages 351–362. ACM, 2010.