Software Defined Networking in Wireless Networks

Abstract

Wireless networks share a common channel that is spatially reused. This gives rise to interesting and complex interference phenomena that have substantial effects on performance and fairness. The shared channel makes wireless networks a strongly coupled system where decisions by one node can have a substantial and cascading effect on other nodes in the system. Developing effective distributed protocols in such an environment is challenging since local decisions can have significant effect making it difficult to converge to effective operating points.

Software defined networking (SDN) [1, 2, 3] has recently emerged as a compelling model to manage networking infrastructure. In particular, SDN decouples the control plane – the component of a network that makes decisions on how the packets should be handled by each router – from the data plane – the component of the network that implements these decisions and forwards the packets. This decoupling allows centralized (or semi-centralized) control of the network, to avoid the complexity, inefficiency and inconsistencies that often arise from distributed control.

SDN is establishing itself as a game changing technology in managing network infrastructure in a number of networking domains. SDNs makes it possible to carry out decisions that are globally effective and introduce the level of coordination not possible with distributed protocols. In addition, an SDN framework offers advantages with respect to service deployment, network monitoring and instrumentation as well as security.

Wireless mesh network (WMN) is created by several wireless Access Points creating an adhoc network in a mesh topology. WMNs allow Internet access to communities, neighborhoods, and parks. Wireless mesh networks have been around for a while but some of the fundamental issues with these networks in terms of interference, channel assignments, scheduling, etc. are still active area of research.

In this research project, we investigate the feasibility of using Raspberry PIs – a lowcost credit card-sized single-board computer, as an SDN enabled wireless routers for a Wireless Mesh Network.

Research question and Significance

Software Defined Networking (SDN), a new architecture that brings software programmability to networks, is transforming the networking industry [4][5]. It aims to make networks easier to operate and manage, and better able to respond to changes in network conditions and application demands. SDN separates the control plane (e.g., routing) from the data plane (e.g., packet forwarding) in routers, centralizes the control plan logic in (typically) dedicated devices (called controllers or SDN controllers), and provides interfaces for the direct programmability of networking devices [6]. This allows for ease of network management, adaptation, and quick experimentation of new protocols and policies. SDN is rapidly being commercialized in the networking industry and is being increasingly deployed [4][5]. For instance, Google recently announced that it has completely redesigned its inter-datacenter WAN to use SDN and have reported substantial performance improvements.

In traditional IP routers/switches, data plane operations (e.g., packet forwarding) and control plane operations (e.g., routing decisions) are performed on the same device. SDN [7] decouples the control plane and the data planes similar to other prior works [7], makes the network state and intelligence logically centralized, and provides interfaces for programming networking devices. Therefore, in the SDN architecture, the data plane still resides on a network router/switch, however, control plane operations are moved to a separate SDN controller(s), which is typically a standard server. The network state (or intelligence) is logically centralized in software-based SDN controllers, which maintain a global view of the network. The network state is built by the traffic statistics sent periodically by routers/switches in the network.

OpenFlow is the first standard interface for communication between SDN controllers and the network devices (i.e., routers/switches) [8]. OpenFlow provides basic primitives that can be used by an external software application to directly program the data plane of networking devices. In particular, an OpenFlow switch provides a flow table abstraction, in which each flow table entry contains a set of packet fields to match and a corresponding action (e.g., drop, update). When an OpenFlow switch receives a packet from a flow for which there is no matching entry in the flow table, the packet is sent to the SDN controller. The controller then decides how to handle the new flow.

There has been some work done in using SDNs in wireless networks. Researchers in Stanford have developed a Wireless extension of their OpenFlow based testbed. This network is deployed on Stanford University campus and enables researchers to run their algorithms and protocols on a production network [9]. OpenRadio is another initiative from Stanford University that provides a programmable wireless data plane [10]. OpenRadio divides each wireless protocols into two main components; the processing component that is responsible for carrying out specific actions while the decision component determines which action to perform. This architecture allows for flexibility in modifying the physical and MAC layers of commodity Access Points.

This research proposal focuses on using Software Defined Networking to address the challenges posed by Wireless Mesh Networks. We seek to answer the following questions: Is it feasible to use Raspberry PIs as open-flow enabled wireless access points in order to provide low cost wireless mesh networks to localized communities.

Project Design and Plan

1. Explore the Raspberry PI boards and install OpenFlow on them.
2. Determine the maximum possible throughput supported by the raspberry Pi by stress testing it.
3. Set up a mesh network with a central controller and Raspberry Pi's as mesh routers.
4. There are many applications that can be implemented using SDNs. SDNs allow newer techniques in Traffic Engineering, Mobility, Measurement and Monitoring, and Security applications. We plan on investigating the handoff algorithms for mobile clients in a wireless network. Specifically we intend to:

    a. Determine mobility patterns of mobile units based on historical data in a network that is collected by the controller. As the controller determines patterns of user mobility and the chain of Access Points user connects whit as he moves, the controller can proactively created connections for the user with Access Points that are more likely to be in it path.
    b. Investigate the possibility of preemptive handoffs to minimize delay and packet loss

If successful, the efficient handoffs will open new doors for mobility management in such mesh networks. In situations where a client is moving very quickly from one access point to the next data transfer becomes very inefficient due to frequent packet drops however if the sequence of handoffs is identified beforehand this packet drops can be significantly reduced. Once possible application is the provisioning of wireless network access on the road.

Qualifications and Goals

Relevant Coursework and projects:

- SLATE: a network-based collaborative whiteboard app (functionally similar to Google Docs) developed for CarnegieApps Hackathon'14.
- Introduction to Computer Systems
  - HTTP PROXY: Fully Developed a HTTP v1.0 cached proxy server for a course homework
- Computer Networks
  - LISO Web Server: Developed an HTTP webserver with support for HTTPS and CGI
  - Bit-torrent: Developed a chunk based peer-to-peer file transfer application
  - Content Distribution Network: Developed an adaptive bitrate video streaming proxy with a DNS server for load balancing among servers.

Since it has been only 2 years that I have been exposed to the field of computer science I have not yet decided upon a career path for my self. However having taken system courses that I have found immensely interesting, I feel that I must explore this field more, especially networks, which is where the future of technology is going.

As mentioned above SDNs have potential to revolutionize computer networks and offer administrators unprecedented flexibility. I feel that this is a chance for me to explore this area of cutting edge research and hopefully contribute, in whatever way I can. Moreover if our experiments with using raspberry pi's as network infrastructure succeeds it will open the door for easy to deploy/manage, cost effective networks to be set up. This may improve network accessibility without the need for expensive infrastructure improvement. I also intend to explore how this may facilitate network deployment in less technologically developed regions.

Budget and Timeline

- Term: Summer 1
- Funding: Individual Fellowship
- We would also like to request funding to purchase 8 Raspberry PIs at $40 each. Total of $320.
- Schedule:
  - Week 1: Do background reading in Software Defined Networks
  - Week 2: Get familiar with hardware and OpenFlow software
  - Week 3 – 4: Evaluate the feasibility and effectiveness of Raspberry PI as wireless Access Points
  - Week 5 – 6: Develop measurement schemes to determine mobility patterns and investigate the possibility of predictive handoffs.
  - Week 7: Document research work.

References

1. Open networking foundation. 2011.
2. Software defined networking. https://www.opennetworking.org/sdn-resources/sdn-definition. Last visited on: 19/4/2015.
3. Ashikur Rahman and Nael B. Abu-Ghazaleh. On the expected size of minimum-energy path-preserving topologies for wireless multi-hop networks. In 2014 IEEE Conference on Computer Communications, INFOCOM 2014, Toronto, Canada, April 27 - May 2, 2014, pages 889–897, 2014.

4. Jain, Sushant, et al. "B4: Experience with a globally-deployed software defined WAN." ACM SIGCOMM Computer Communication Review. Vol. 43. No. 4. ACM, 2013.

5. Hp launches comprehensive sdn portfolio. http://www.datacenterknowledge.com/archives/2012/10/03/hp-launches-comprehensive-sdn-portfolio/. Last Accessed April 20, 2015

6. Barath Raghavan, Martín Casado, Teemu Koponen, Sylvia Ratnasamy, Ali Ghodsi, and Scott Shenker. Software-defined internet architecture: Decoupling architecture from infrastructure. In Proceedings of the 11th ACM Workshop on Hot Topics in Networks, HotNets-XI, pages 43–48, New York, NY, USA, 2012. ACM.

7. Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." proceedings of the IEEE 103.1 (2015): 14-76.

8. Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. Openflow: enabling innovation in campus networks. ACM SIGCOMM Computer Communication Review, 38(2):69–74, 2008.

9. Kok-Kiong Yap, Masayoshi Kobayashi, Rob Sherwood, Te-Yuan Huang, Michael Chan, Nikhil Handigol, and Nick McKeown. Openroads: Empowering research in mobile networks. SIGCOMM Comput. Commun. Rev., 40(1):125–126, January 2010.

10. Manu Bansal, Jeffrey Mehlman, Sachin Katti, and Philip Levis. Openradio: A programmable wireless dataplane. In Proceedings of the First Workshop on Hot Topics in Software Defined Networks, HotSDN ’12, pages 109–114, New York, NY, USA, 2012. ACM.