SDN-based management of heterogeneous home networks

Reference:
Soetens Niels, Famaey Jeroen, Verstappen Matthias, Latré Steven.- SDN-based management of heterogeneous home networks
Proceedings of the 2nd International Workshop on Management of SDN and NFV Systems / DMTF Datacenter and Cloud Management Workshop (ManSDN/NFV-DCM), Barcelona, Spain - ISBN 978-3-901882-77-7 - S.l., 2015, p. 1-4

SDN-based management of heterogeneous home networks

Niels Soetens, Jeroen Famaey, Matthias Verstappen, Steven Latré

Department of Mathematics and Computer Science, University of Antwerp - iMinds, Belgium
Email: *niels.soetens@uantwerpen.be*

***Abstract*—In recent years a lot of new consumer devices have been introduced to the home network. Modern home networks usually consists of multiple heterogeneous communication technologies such as Ethernet, Wi-Fi and power-line communications. Today, the user has to manually decide which transmission technology to use as there is no automated optimization across technologies. Load balancing algorithms can improve overall throughput while redundant links also provide the opportunity to switch flows in case of link failures. Current standards either lack real implementation in consumer devices or do not have the flexibility to support all necessary functionality towards creating a convergent hybrid home network. Therefore, we propose an alternative way by using Software-Defined Networking techniques to manage a heterogeneous home network. In this paper we specifically evaluate the ability of OpenFlow-enabled switches to perform link switching both under normal conditions and in case of link failures. Our results show that SDN-based management can be used to improve heterogeneous home networks by utilising redundant links for flow rerouting. However, they also show that improvements are still needed to reduce downtime during link failure or rerouting in case of TCP traffic.**

***Index Terms*—Heterogeneous Home Networks, Link Switching, Load Balancing, Software-Defined Networking, OpenFlow.**

I. Introduction

Future home networks are expected to contain a multitude of consumer devices which connect using different wired and wireless communication technologies. Home network architectures already consist of a mix of Wi-Fi, Ethernet and power-line communication. This implies that there can be multiple paths of communication between the home gateway and home network devices. Current in-home network protocols will always use a single default connection (e.g. wired if available), therefore neglecting the possibility of optimizing the network to exploit its full potential.

Video on-demand services such as YouTube or Netflix are very popular. Consequently, in-home devices require more bandwidth. Current state of the art of in-home network management does not show much possibility to use the full potential of all technologies. Efforts are being made with IEEE 1905.1, which enables load balancing across heterogeneous technologies [1]. However, an efficient and autonomous load balancing algorithm is still needed to utilise the full potential of this standard. Such an algorithm can be used to reroute flows, in case of link degradation or failure, and load balancing among all available paths in the home network. Nevertheless, only a handful commercial products have already implemented this technology standard. Therefore, in this paper we propose an alternative way for creating a fast and resilient hybrid home network by using Software-Defined Networking (SDN) techniques. SDN provides an alternative way to control the network and is already more commonly used inside networks today. Our research looks at traffic redirection in case of link failure or congestion. Introducing an SDN controller in the home environment that has an overview of the network with all the available devices, links and metrics which can be used to take automated actions on the network and redirect traffic flows if necessary.

The contributions of this paper are twofold. First, we present an architecture for the hybrid home network based on SDN techniques. By creating a heterogeneous network with the use of OpenFlow we enable link switching between wired and wireless technologies. Second, we perform experiments on a real set-up based on the SDN architecture. We look at packet loss and throughput when performing link switching on heterogeneous links. This could be during active link switching due to congestion or reactive switching in case of link failure.

The remainder of this paper is organized as follows: in Section II the works related to this paper are summarized. In Section III we give our view of the future heterogeneous home network and describe our test set-up in more detail. Section IV describes the experiments and gives an overview of the results. Finally, Section V presents the conclusion of this work.

II. Related work

Optimising the home network to fully use the multipath environment can strongly improve the overall network performance [2]. Creating a unified high bandwidth home environment has become more important in recent years. In 2008, Javaudin et al. described a convergent gigabit home network using heterogeneous transmission technologies [3]. The goal was to create an Inter-MAC architecture for home network devices [4]. This means creating an abstract MAC layer on top of the current data link layer (OSI layer 2) to transparently combine all the heterogeneous MAC interfaces on a network device [5]. Eventually, their contribution formed the basis of the IEEE 1905.1 standard towards a convergent digital home network. The benefits that come with IEEE

978-3-901882-77-7

1905.1 are simple set-up, configuration and operation of network devices with heterogeneous technologies [6].

Several load balancing algorithms have already been proposed each with a different focus. Sahaly and Christin defined an algorithm that uses a per-flow decentralized load balancing technique within the heterogeneous home network [7]. With a per-flow approach it is capable of reactively distributing incoming flows on the available links. Macone et al. proposed a per-packet load balancing technique [8]. This can theoretically provide a better result because of the lower granularity. However, per-packet load balancing in combination with TCP can result in unnecessary retransmissions because the nature of TCP wanting to deliver packets in order, resulting in lower throughput of the system. Olvera-Irigoyen et al. showed that bandwidth probing for path selection in heterogeneous home networks can provide significantly better results in distributing the flows [9].

Summarized, current research mostly focusses on developing theoretical models for technology switching and load balancing in heterogeneous home networks. In contrast, the focus of this paper is to use existing and commonly available techniques such as SDN to evaluate the effects of technology switching in a real environment.

III. SDN-based Hybrid Home Network Architecture

Devices such as desktops, laptops or set set-top boxes usually have multiple network interfaces. Moreover, in densely populated areas, e.g. apartment buildings, several Wi-Fi connections may co-exist. The whole topology with all interconnected links will result in a multipath environment. This creates new opportunities to increase the network's maximum throughput and make it resilient against link failures. As the current implementation of standards for creating a convergent network are very limited, we propose an alternative approach to autonomously and efficiently manage heterogeneous multipath home networks, based on readily available standards and technologies. Specifically, our architecture employed SDN concepts to implement multipath optimization, resilience and load balancing.

A. SDN managed home network

The key element to SDN is to differentiate the control plane from the data plane. By introducing an SDN controller managing the networks control plane, other network devices only act as forwarders in the data plane and do not need any intelligence. Figure 1 shows an example of a heterogeneous home network managed by an SDN controller which monitors the SDN-enabled devices. The controller could be a standalone device monitoring the network or integrated in the home gateway. Consumer devices such as PCs, tablets, smart-phones or set-top boxes are connected by Ethernet, Wi-Fi, power-line communication or a combination of these technologies which results to multiple paths in between network devices. SDN-enabled devices communicate their links and statistics using an SDN communication protocol.

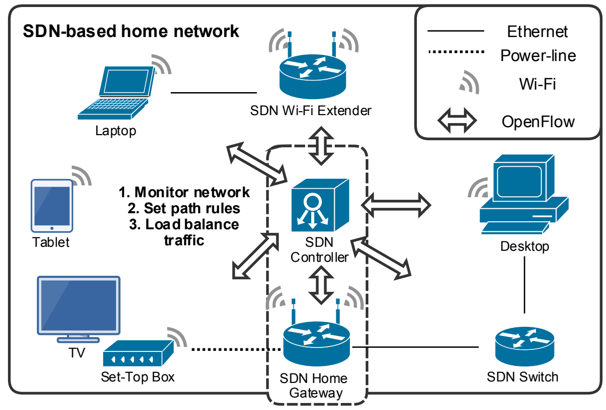

Fig. 1: SDN-based hybrid home network architecture.

B. OpenFlow

OpenFlow is an open SDN standard to send information between network devices and the SDN controller. This protocol plays a crucial role for creating SDN networks where the control plane is separated from the forwarding plane on network devices.

1) Monitor the network: Available links between devices are communicated towards the SDN controller using OpenFlow. Link status monitoring is achieved by regularly sending Link Layer Discovery Protocol (LLDP) messages through the network. Furthermore, extra information from interfaces is gathered, including throughput and packet loss. With this information the controller can create a topology of the home environment.

2) Forwarding rules: If the controller has the topology information together with the monitoring statistics, it can make decisions on how to route traffic over different network transmission technologies. A set of rules are created by the controller and then sent to the corresponding network devices using the OpenFlow protocol. Therefore, each device gets a dynamic set of rules as flows enters and leave the network.

C. SDN-based flow management

When a new traffic flow enters the network, the header of the first packet is communicated to the SDN controller. The controller evaluates which route to take within the heterogeneous environment. Once calculated the controller sets appropriate forwarding rules on the network devices. As this is a layer 2 approach, MAC address translation is used in the rules to forward packets on the correct interface. This approach makes switching of flows transparent to the transport and application layers.

When available, the SDN controller can decide to switch flows to redundant paths in case of packet loss or link degradation. Therefore, allowing to reroute existing flows or distribute flows over different paths towards the same destination. Monitoring statistics, such as packet loss and throughput, on each SDN managed interface will help the controller to take appropriate measures in case of traffic loss or congestion.

IV. Evaluation and discussion

The main goal of our experiments is to determine how SDN techniques can contribute towards a convergent heterogeneous home network. Therefore, an experimental set-up is created according to the architectural approach outlined in Figure 1. Experiments have been conducted on how the network behaves during link switching under normal conditions and when introducing link failures. Furthermore, we look at more detail on the downtime of UDP and TCP flows during link switching.

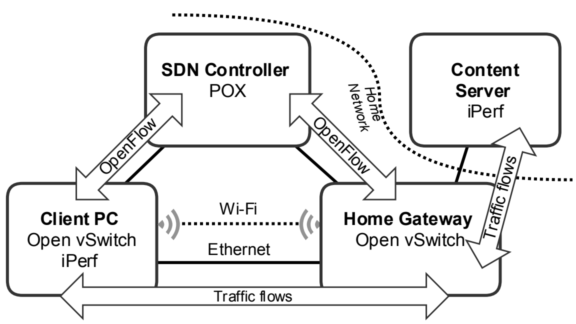

Fig. 2: Experimental set-up using SDN techniques.

A. Experimental Set-up

Figure 2 shows our experimental set-up consisting of four Linux devices used to simulate a small heterogeneous home environment. On this figure we can differentiate a client PC, the home gateway, a network SDN controller and an content server. A simple multipath connection has been established between the PC and home gateway in the form of a wired (Fast Ethernet at 100 Mbps) and wireless (Wi-Fi 802.11g at maximum 24 Mbps) connection.

POX is used as a software platform to act as an SDN controller. It is used to develop and prototype network control functionality using Python. Within the POX controller modules can be added or changed to create custom forwarding rules based on monitored statistics of the network. The experimental set-up consists of two Open vSwitch Linux devices located on the client pc and the home gateway. Open vSwitch is an open source implementation of a virtual switch. These Open vSwitches can be remotely managed by the SDN controller using OpenFlow. Furthermore, iPerf is used to create UDP and TCP traffic flows between the client PC and the content server.

B. Link switching on normal conditions

When multiple paths are available, flows can be switched between heterogeneous interfaces. In this first experiment we look at the impact of rerouting flows on different types of links within our SDN managed network. Following results were averaged over 10 iterations, but for presentation purposes only 1 of these iterations is shown in the graphs for both UDP and TCP.

An experiment with running UDP traffic at 10 Mbps from content server to client PC is shown in Figure 3, where the throughput is plotted over time measured on the wired and wireless interface of the client. At first, only the Wi-Fi link is available and the POX controller generates rules for both home gateway and client to use the wireless connection. After approximately 10 seconds the Ethernet interfaces comes up between the home gateway and client PC. The controller is notified with the new link and changes its topology. We configured the POX controller to prefer this wired link instead of the wireless link. Therefore, it decides to set new rules for the existing flow to use the wired link. These rules are communicated with OpenFlow towards the home gateway and client PC.

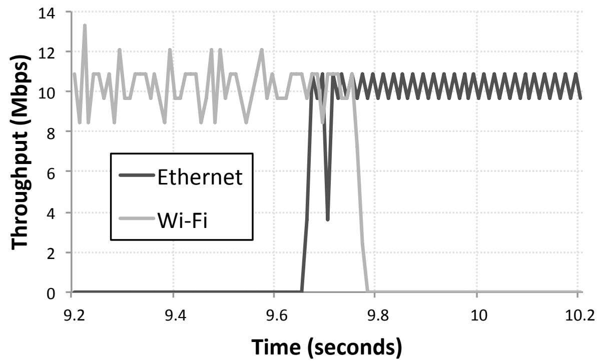

Fig. 3: Link switching with UDP traffic at 10 Mbps during normal conditions from Wi-Fi to Ethernet link.

The rerouting of flows is performed almost seamlessly between heterogeneous links. As can be seen on Figure 3, packets already start arriving on the wired link before the packets on the wireless fade out. However, this could result in packets arriving out of order. During a short interval, on average 20 milliseconds, after link switching we observed an average of 70% packet loss over 10 iterations. This observed packet loss, although very short, is most likely the result of delay between setting the rule on the source and destination node of the link. Similar results can be obtained from switching from wired to wireless connection, though these results were omitted due to space constraints.

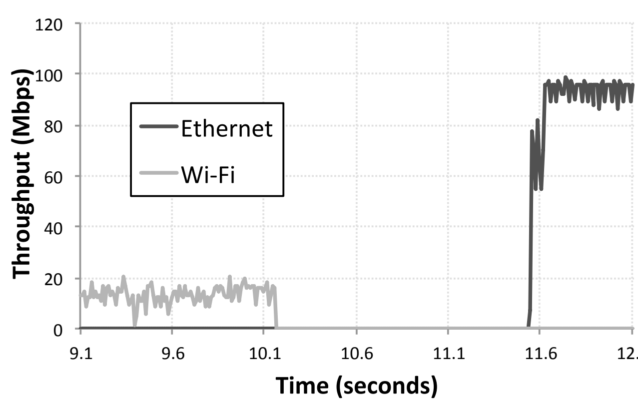

Fig. 4: Link switching from Wi-Fi to Ethernet link with TCP traffic during normal conditions.

Figure 4 shows a similar experiment with a TCP flow running through the network from the content server towards the client PC. Again starting from only the wireless link we see that TCP fills up the available bandwidth. However, when the wired link is brought up after 10 seconds in the experiment and rules are applied by the POX controller, the TCP connection is reduced to 0 Mbps throughput for on average 1.4 seconds over 10 iterations before it switches to the Ethernet connection. This is most likely caused by TCP trying to adjust itself when packet loss is observed or by packets being received out of order. We conclude that the standard OpenFlow solution shows some shortcomings that need to be overcome in case of TCP traffic and further study of this issue is necessary.

C. Link switching on link failure

The second experiment looks at how our SDN-based home network reacts to link failure in a system with redundant paths. An important factor is to know as fast as possible when a link failure occurs. This information needs to be communicated towards the POX controller so it can decide to reroute existing flows on redundant paths.

Within the POX controller it is possible to change the link timeout parameter. This value specifies the time for the POX controller to wait before deciding a link failures has occurred. As mentioned earlier, POX uses LLDP to gather information from available links. Lowering the link timeout parameter will increase the LLDP send cycle. Therefore, if no LLDP information is gathered for a specific interval, the link is considered to be down.

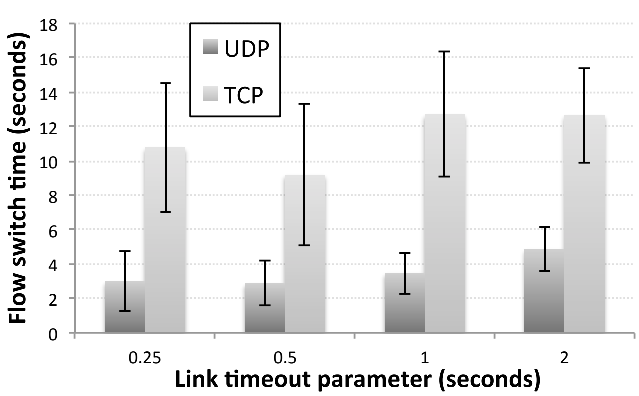

Fig. 5: Traffic downtime during link failure shown for both UDP and TCP traffic with changing link timeout values.

Figure 5 shows the traffic downtime for both UDP and TCP traffic measured over time for different values of the link timeout parameter. The experiment consists of sending traffic over Ethernet and bringing the link down after 10 seconds. The downtime is measured until traffic is flowing again on the redundant Wi-Fi link. For each value of the link timeout, 10 runs were conducted to calculate the average downtime and the error bars represent the standard deviation.

Results show UDP traffic can be rerouted on average within 3 seconds when the link timeout is set to 0.5 seconds. Lowering this value does not result in lower downtime and it also increases the chance of observing false link failure detections. These false detections happen when the POX controller falsely assumes a link went down because it did not receive information over a certain time interval, although the link was still up. TCP traffic has a overall longer downtime on link failure and link rerouting. At best, an average downtime is seen of approximately 9 seconds with a link timeout value of 0.5 seconds. It is observed that TCP stops the flow for some time before trying to retransmit, resulting in a higher downtime compared to UDP. This experiment shows that the POX controller is capable of detecting link failure and reroute flows accordingly. However, high downtimes are still observed, especially for TCP traffic.

V. Conclusion

In this paper, an SDN-based architecture to manage heterogeneous home networks is presented. In general, our results show that the home environment can benefit from having SDN managed heterogeneous interfaces on network devices which enables flow rerouting in case of link degradation or complete failure. The standard OpenFlow solution still needs to be improved for TCP as the traffic downtime is too long between link switching.

Acknowledgment

Part of this work has been funded by the SHIFT-TV project, co-funded by iMinds, a digital research institute founded by the Flemish Government. Project partners are Alcatel-Lucent, Technicolor, Proximus and OpenTelly, with project support from IWT.

References

[1] "IEEE Standard for Convergent Digital Home Network for Heterogeneous Technologies," *IEEE Std. 1905.1-2013*, 2013.

[2] B. Almási and A. Harman, "An overview of the multipath communication technologies," in *Proc. Conf. on Advances in Wireless Sensor Networks*, 2013, pp. 7–11.

[3] J.-P. Javaudin, M. Bellec, D. Varoutas, and V. Suraci, "OMEGA ICT project: Towards convergent Gigabit home networks," in *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*. IEEE, 2008, pp. 1–5.

[4] T. Meyer, P. Langendörfer, M. Bahr, V. Suraci, S. Nowak, and R. Jennen, "An inter-mac architecture for heterogeneous gigabit home networks," in *Personal, Indoor and Mobile Radio Communications, 2009 IEEE 20th International Symposium on*. IEEE, 2009, pp. 1–5.

[5] J.-P. Javaudin, M. Bellec, P. Jaffré, O. Hoffmann, A. Foglar, and O. Isson, "Inter-mac concept for gigabit home networks," in *Personal, Indoor and Mobile Radio Communications, 2009 IEEE 20th International Symposium on*. IEEE, 2009, pp. 1–5.

[6] E. Cohen, D. Ho, B. Mohanty, and P. Rajkotia, *IEEE 1905.1*. CRC Press, 2015/07/22 2014, pp. 427–454. [Online]. Available: http://dx.doi.org/10.1201/b16540-19

[7] S. Sahaly and P. Christin, "Inter-MAC forwarding and load balancing per flow," in *Personal, Indoor and Mobile Radio Communications, 2009 IEEE 20th International Symposium on*. IEEE, 2009, pp. 1–4.

[8] D. Macone, G. Oddi, A. Palo, and V. Suraci, "A dynamic load balancing algorithm for Quality of Service and mobility management in next generation home networks," *Telecommunication systems*, vol. 53, no. 3, pp. 265–283, 2013.

[9] O. Olvera-Irigoyen, A. Kortebi, and L. Toutain, "Available bandwidth probing for path selection in heterogeneous home networks," in *Globecom Workshops (GC Wkshps), 2012 IEEE*. IEEE, 2012, pp. 492–497.