SDN Enabled Resiliency in LTE Assisted Small Cell mmWave Backhaul Networks

Jonathan Vestin and Andreas Kassler
Computer Science Department
Karlstad University
Karlstad, Sweden
Email: {jonathan.vestin, andreas.kassler}@kau.se

***Abstract*—As mobile data usage increases dramatically, new architectures and technologies for wireless communication are required. Next generation of mobile networks are expected to be augmented by a massive amount of small cells that will be densely deployed. In order to connect the small cells, new high capacity wireless backhauling technologies are required. A promising solution is to use frequencies in the mmWave band, which allows for much greater capacity due to the massive amount of free spectrum. However, the special characteristics of the mmWave bands such as high path loss and atmospheric absorption lead to unstable links. In this paper, we investigate using Software Defined Networking principles for the operation and control of wireless backhaul networks. We demonstrate how SDN resiliency mechanisms can be used to mitigate disruptive connectivity due to mmWave links frequently failing. For assisting the small cell backhauling, we propose to also use the LTE uplinks of the small cells as backup links, should the mmWave mesh forwarding link break. Our experiments using a network emulator show that using SDN-based local repair mechanisms can significantly reduce the packet loss rate inside the mmWave backhaul mesh, which can be further reduced with an LTE assisted Failover.**

I. Introduction

Mobile data usage has risen 4000-fold over the last 10 years according to the 2015-2020 CISCO Global Mobile Data Traffic Forecast [1], with global mobile data traffic expected to rise eightfold between 2015 and 2020. Mobile video traffic accounts for 55% of that data, and is expected to rise to 75% before 2020. The current standard cellular technology LTE has seen numerous recent physical layer technology improvements, including massive MIMO, better modulation and coding schemes, and coordinated multipoint transmission. However, in order to properly support the high throughput which future networks are predicted to require, a common assumption has been that using a higher frequency band, such as mmWave bands, where more spectrum is available and reducing the cell size in order to increase spatial reuse could provide the necessary throughput increase [2].

Compared to typical radio waves, mmWave frequencies, such as 60 GHz exhibit rather different propagation characteristics, such as increased attenuation due to high free space path loss and atmospheric absorption. Another difference is that exacerbated blockage and shadowing could lead to rapid variation in link quality. This could in turn lead to links changing their characteristics swiftly between line-of-sight (LOS), non-line-of-sight (NLOS) oroutage (OUT)) [3]. To compensate for this increase in path loss, phased antenna arrays can facilitate beamforming [2], achieving high directional gain. Thus, mmWave communication enables short range point-to-point (PtP) radio links that not only are suitable for access but also for novel backhaul solutions. Indeed, when a dense deployment of large cells is needed, the cost for cabling each small cell may be prohibitive. Consequently, network architects are considering to use mmWave to rely user data from one small cell to another, forming a mesh or tree based multi-hop wireless backhaul architecture. A large umbrella cell formed by the LTE eNodeB may provide coverage while the small cell mesh based backhaul may offer localized capacity on demand, depending on which small cells are activated. The backhaul traffic would then be routed multihop towards a local aggregation gateway (typically co-located at the LTE eNodeB).

Currently, the proprietary nature of many backhaul networks hinder innovation and the development of innovative and flexible wireless backhaul networks. In traditional networks, to counteract the closed nature of many network devices, technologies such as Software Defined Networking (SDN) [4] were introduced. SDN decouples the control from the data plane. A logically centralized control plane is facilitated by a so called SDN controller (such as e.g. NOX [5] or OpenDaylight [6]). The SDN controller flexibly installs forwarding rules that match certain packet header fields in the data plane using standardized protocols (such as OpenFlow [7]). Due to its increased flexibility, SDN has been proposed to be used for wireless backhaul networks [8], [9]. However, how to cope with the inherently unstable nature of mmWave links for small cell wireless backhaul networks has not yet been adequately addressed before, especially considering the context of multihop wireless backhauling.

Recently, OpenFlow has been augmented with resiliency mechanisms such as fast failover through the group tables feature in OpenFlow v1.1 [10]. Group tables allow OpenFlow switches to perform fast local reaction to failure events without invoking the controller. We believe that this feature will make an SDN-based architecture a feasible way to operate and manage stationary mesh backhauls, because the controller can calculate pre-programmed backup paths, which are then used locally once the primary link is declared down. A link failure is detected locally using Bidirectional Forwarding Detection (BFD) [11] with a configurable monitoring interval. This link failure detection together with the local repair mechanism

978-1-5090-3672-1/17/$31.00 ©2017 IEEE

effectively eliminates a round trip time communication with the controller once the link is down. Consequently, a failover can be triggered very rapidly leading to very short reaction times [12]. Still, once the primary mmWave link fails, another small cell neighbor maybe used towards the aggregation gateway, which again may fail. An alternative may be to use an LTE assisted scheme, where the small cell sends the packets to the eNodeB, if the mmWave backhaul links fail.

In this paper, we propose to use SDN-based fast failover approaches for mmWave small cell wireless backhaul links. Our approach effectively reduces the packet loss caused by frequent topology changes due to link outage events commonly seen in mmWave communication. In our approach, the SDN controller optimizes the routing inside the forwarding mesh and calculates for each link a backup link. By using BFD, a link failure is detected and the SDN-enabled small cell node switches locally to the backup link. Alternatively, the small cell backhaul node may use LTE assist mode, where packets are forwarded directly over an LTE uplink to the eNodeB in case the mmWave links fail. By using an extensive set of experiments with a network emulator, we demonstrate that our approach significantly improves the performance for small cell mmWave backhauling when fast local topology changes are present due to the unstable nature of mmWave links, thus increasing network robustness.

II. Experimental Setup and Results

A. Experimental Setup

In order to evaluate the SDN-based resiliency mechanisms for small cell wireless backhaul operation, we modified[1] the CORE [13] 4.8 (b86881aba967..) emulator to emulate mmWave links. The CORE emulator uses the Linux Containers (LXC) feature called network namespaces. We integrated Open vSwitch (OVS) v2.3.0 into CORE, which is a SDN-enabled virtual switch that provides OpenFlow functionality and supports BFD and fast failover group tables.

We also implemented a probability based link state model [14] to emulate the characteristics of mmWave links. The model defines the probability for each link to be in a LOS, NLOS or OUT state depending on the link distance d between the small cell sender and recipient. Essentially, the higher d, the larger the probability for the link to be in a NLOS or OUT state. We configured the evaluation scenario so that each link enters a new randomly selected state every $500\,\mathrm{ms}$ and used state transition probabilities as given in [14]. In the LOS state, the capacity of the link is set to $300\,\mathrm{Mbps}$, when the link is in the NLOS state the link capacity is $100\,\mathrm{Mbps}$ and if the link is in a OUT state, the link is configured with a 100% packet loss. We assume that during LOS or NLOS, any lower layer packet loss is recovered at the link layer due to e.g. HARQ. Those state transitions emulate the behavior of mmWave links because due to blockage, a mmWave link may be blocked leading to no throughput for some time [15]. Note, that throughput values are scaled down by a factor of 10 to decrease the load on the emulator. Another extension we made

[1]Available here: https://github.com/tohojo/core

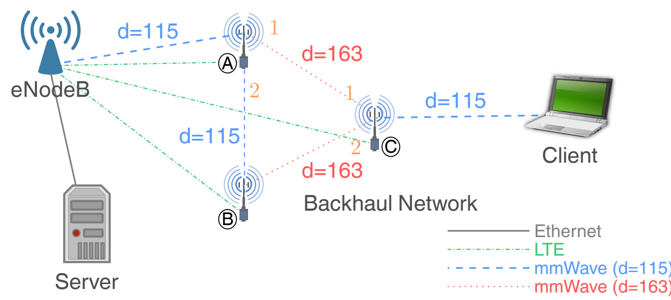

Figure 1. mmWave small cell backhaul emulation scenario.

to CORE uses Network Emulator (netem) instead of Token Bucket Filter (TBF) to perform rate limiting, which allows us to control the buffer size of the links to not exceed $1\,\mathrm{ms}$.

Figure 1 shows our scenario setup. Client and server represent the traffic source and sink respectively. The server is connected to the eNodeB node with an emulated $1000\,\mathrm{Mbps}$ Ethernet link at a delay of $5\,\mathrm{ms}$. The eNodeB and the client nodes are connected to the small cell backhaul nodes, however they are also using OVS using OpenFlow version 1.3 to be controlled by SDN and having fast failover functionality. The numbers next to the backhaul nodes show which link is the primary and backup link. The primary link is indicated with a 1 and the backup link with a 2. Also shown in Figure 1 is the distance d of each link in meters. As can be seen, the eNodeB is connected to small cell A ($d = 115$), A is connected to C as its primary link ($d = 163$) using B as its backup link ($d = 115$). Node B is connected to node A ($d = 115$) and node C ($d = 163$). This setup causes the link $A \rightarrow C$ to intermittently fail to link $A \rightarrow B$, and should that link also fail, the packets are dropped. The state change events are pregenerated and stored in an events file. The same events file is used for all tests. Each link in the small cell backhaul network is configured with a latency of $1\,\mathrm{ms}$.

We also included another scenario which allows for additional redundancy equipping each small cell with a separate LTE connection to the eNodeB. Consequently, for the uplink transmission, if node C has no available mmWave links, it will forward the packet to the eNodeB using fast failover. Moreover, during downlink transmission if node B has no available mmWave channel, it will also forward the packet to the eNodeB, which will in turn forward it to the last hop (node C). This increased redundancy however comes at a cost; the LTE link is configured with a latency of $12\,\mathrm{ms}$, and a jitter of $5\,\mathrm{ms}$ with a paretonormal distribution emulating an LTE link[16] and a throughput of $2\,\mathrm{Mbps}$ (scaled down from $20\,\mathrm{Mbps}$).

Figure 2 shows the state change events in each of the test scenarios for two particular links: $A \rightarrow C$ ($163\,\mathrm{m}$) and $A \rightarrow B$ ($115\,\mathrm{m}$). For readability reasons, only the first 30 seconds of the test is shown. The y-axis shows the current state which the particular link is in, while the x-axis shows the time.

B. Results and Analysis

In this section we present the evaluation results from the described scenario. Each scenario is divided into two different

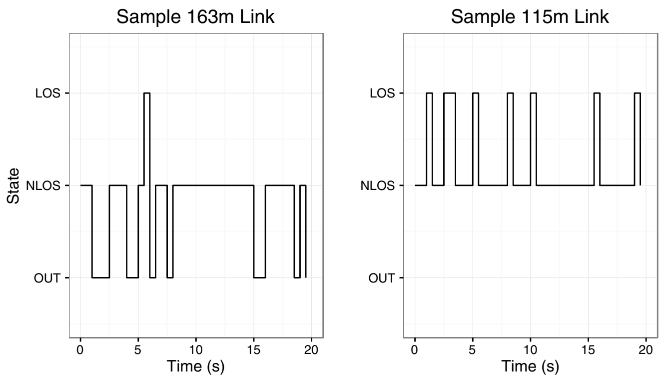

Figure 2. State change events between 5 and 30 seconds elapsed test time for links $A \rightarrow C$ (163 m) and $A \rightarrow B$ (115 m).

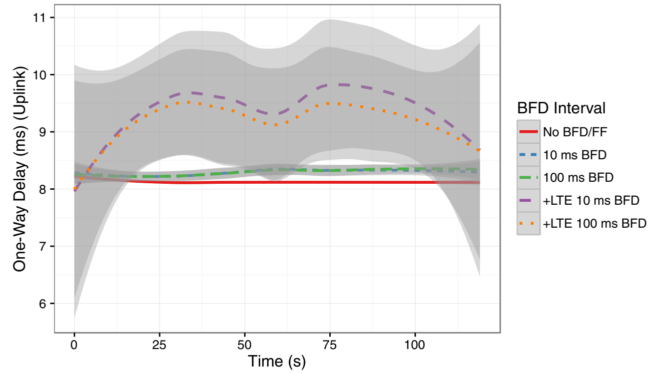

Figure 4. Upink delay using various BFD intervals, with/without the LTE backup link.

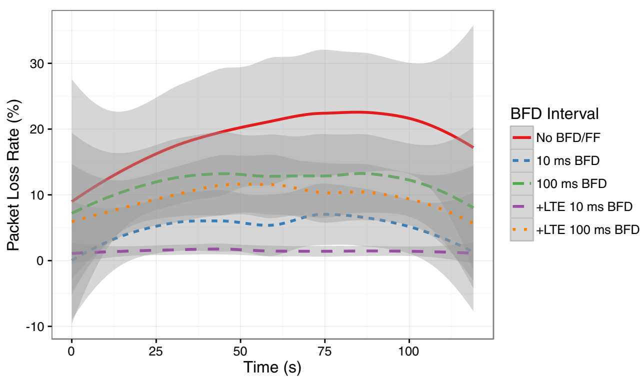

Figure 3. PLR using various BFD intervals, with/without the LTE backup link.

tests, one using only the mmWave links together with fast failover, and another using additionally the LTE assisted Failover connection to the eNodeB (marked with +LTE).

Figure 3 shows the PLR of the two test scenarios using five separate configurations. The first test, (“No BFD/FF”) is configured to not use BFD or fast failover. Here, when the mmWave link goes into outage state, packets will be lost. The second and third test (“10 ms BFD” and “100 ms BFD”) are configured using fast failover and BFD with a heartbeat interval of 10 ms and 100 ms respectively. Here, when the primary mmWave link goes into outage state, fast failover is triggered locally and the other mmWave link is used. The fourth test is using the LTE eNodeB link to the fast failover group. Each link is configured using a BFD heartbeat interval of 10 ms. The fifth test is also using the LTE eNodeB connection as Failover candidate, but instead uses a BFD interval of 100 ms. The test ran for 2 minutes and was performed by sending 12 byte UDP pings every 20 ms from the server to the client. The client encodes the transmission time of each packet into the data field. The server then records the time difference between the timestamp and the current time. The timestamp is encoded in microseconds using a 64-bit integer. The clock on the client and server are automatically synchronized as they are both running on the same machine (both run inside the Core emulator). Figure 3 shows the PLR over time. The lines are smoothed using Locally Weighted Scatterplot Smoothing (LOESS) (`stat_smooth` in R) in order to increase readability.

As we can see from the Figure, not using fast failover leads to an average PLR of around 19%. Introducing fast failover with a 10 ms BFD heartbeat interval using only mmWave links leads to a much lower variability in the PLR with average PLR of 5.1%. This is because when the primary link goes into outage, BFD packets are lost. Once the BFD times out, local fast failover is triggered and subsequent packets are sent over the secondary mmWave link. Increasing the BFD interval to 100 ms results in average PLR of around12%. This is because the larger BFD interval leads to a slower outage detection which triggers the fast failover significantly later increasing the PLR. In contrast, when using the LTE assisted fast failover with a BFD interval of 10 ms, the PLR is significantly reduced to in average 1.5%. Using a BFD interval of 100 ms instead, the average PLR increases to 9.6%. This confirms that adding a LTE assisted fast failover to the eNodeB can additionally help reduce the packet loss because we assume that we do not lose any packets on the LTE link. Still, the overall PLR for using LTE assisted fast failover is not zero because it takes some time until the failover is triggered once the primary link is in outage state. All packets during that duration are lost. Consequently, a smaller BFD interval reduces the PLR.

Figure 4 shows the one-way uplink delay over time for the same scenario. The lines are again smoothed using LOESS (`stat_smooth` in R) in order to increase readability. As we can see, not using fast failover achieves an average delay of 8.13 ms. Using fast failover with a BFD interval of 10 ms and 100 ms the average delay is 8.29 ms and 8.3 ms respectively. Finally in the +LTE test configuration, we see even higher average delay of 9.41 ms using a BFD interval of 10 ms and 9.21 ms using an interval of 100 ms. The increase in delay when using a lower BFD interval can be attributed to the longer path length when node C or node A are in a failover state ($S \leftrightarrow A \leftrightarrow C \leftrightarrow R$ vs $S \leftrightarrow A \leftrightarrow B \leftrightarrow C \leftrightarrow R$). The smaller the BFD interval, the longer the network will stay in a disadvantageous state. The additional LTE backup path further increases the latency as the LTE connection between the eNodeB and the nodes have an average one-way latency of 8.19 ms.

III. Conclusion

In next generation mobile networks, a dense deployment of small cells can help to improve the capacity. For such networks, the backhaul operation is crucial in order to not be a bottleneck. Consequently, high capacity backhaul connections are required which not always can be based on fiber technology due to excessive costs. Consequently, high capacity wireless technologies need to be deployed as backhaul solutions. A mesh-based small cell backhaul network based on mmWave links has been considered because of the large portion of bandwidth available at those frequency bands. In such networks, it is important to cope with the intrinsic characteristics of mmWave links, which may lead to unstable operation. Therefore, we have proposed to use a Software Defined Networking based approach to provide a robust forwarding in the small cell mesh backhaul, following the HetNets concept. We investigate different alternatives based on SDN resiliency mechanisms such as fast failover groups. In addition, we proposed a LTE eNodeB assisted operation, where a small cell may offload traffic to the LTE umbrella cell should the mmWave mesh backhaul forwarding link goes into outage state. This can help to reduce the impact of the fast channel attenuation due to the instability of mmWave communications. Our evaluation shows that SDN-based fast failover can significantly decrease the packet loss ratio with small additional latency overhead.

Acknowledgement

Parts of this work has been funded by Knowledge Foundation through the project SOCRA.

References

[1] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update", Tech. Rep., Feb. 2016. [Online]. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html.

[2] W. Roh, J. Y. Seol, J. Park, B. Lee, J. Lee, Y. Kim, J. Cho, K. Cheun, and F. Aryanfar, "Millimeter-wave beamforming as an enabling technology for 5g cellular communications: Theoretical feasibility and prototype results", *IEEE Communications Magazine*, vol. 52, no. 2, pp. 106–113, Feb. 2014, ISSN: 0163-6804. DOI: 10.1109/MCOM.2014.6736750.

[3] S. Sun and T. S. Rappaport, "Wideband mmwave channels: Implications for design and implementation of adaptive beam antennas", in *Microwave Symposium (IMS), 2014 IEEE MTT-S International*, Jun. 2014, pp. 1–4. DOI: 10.1109/MWSYM.2014.6848596.

[4] Open Networking Foundation, "Software-defined networking: The new norm for networks", Tech. Rep., Apr. 2012.

[5] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker, "Nox: Towards an operating system for networks", *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 3, pp. 105–110, Jul. 2008, ISSN: 0146-4833. DOI: 10.1145/1384609.1384625. [Online]. Available: http://doi.acm.org/10.1145/1384609.1384625.

[6] The OpenDaylight Project. (n.d.). Opendaylight: Java-based openflow controller, [Online]. Available: https://www.opendaylight.org.

[7] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: Enabling innovation in campus networks", Tech. Rep., Mar. 2008.

[8] K. Seppänen, J. Kilpi, and T. Suihko, "Integrating wmn based mobile backhaul with sdn control", *Mob. Netw. Appl.*, vol. 20, no. 1, pp. 32–39, Feb. 2015, ISSN: 1383-469X. DOI: 10.1007/s11036-015-0574-7. [Online]. Available: http://dx.doi.org/10.1007/s11036-015-0574-7.

[9] J. Núñez-Martínez, J. Baranda, and J. Mangues-Bafalluy, "A service-based model for the hybrid software defined wireless mesh backhaul of small cells", in *Network and Service Management (CNSM), 2015 11th International Conference on*, Nov. 2015, pp. 390–393. DOI: 10.1109/CNSM.2015.7367388.

[10] Open Networking Foundation. (Feb. 2011). Openflow switch specication: Version 1.1.0 implemented (wire protocol 0x02), [Online]. Available: http://archive.openflow.org/documents/openflow-spec-v1.1.0.pdf.

[11] D. Katz and D. Ward, *Bidirectional Forwarding Detection (BFD)*, RFC 5880 (Proposed Standard), Updated by RFC 7419, Internet Engineering Task Force, Jun. 2010. [Online]. Available: http://www.ietf.org/rfc/rfc5880.txt.

[12] N. van Adrichem, B. Van Asten, and F. Kuipers, "Fast recovery in software-defined networks", in *Software Defined Networks (EWSDN), 2014 Third European Workshop on*, Sep. 2014, pp. 61–66. DOI: 10.1109/EWSDN.2014.13.

[13] J. Ahrenholz, C. Danilov, T. R. Henderson, and J. H. Kim, "Core: A real-time network emulator", in *MILCOM 2008 - 2008 IEEE Military Communications Conference*, Nov. 2008, pp. 1–7. DOI: 10.1109/MILCOM.2008.4753614.

[14] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation", *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1164–1179, Jun. 2014, ISSN: 0733-8716. DOI: 10.1109/JSAC.2014.2328154.

[15] M. Zhang, M. Mezzavilla, R. Ford, S. Rangan, S. S. Panwar, E. Mellios, D. Kong, A. R. Nix, and M. Zorzi, "Transport layer performance in 5g mmwave cellular", *CoRR*, vol. abs/1603.02701, 2016. [Online]. Available: http://arxiv.org/abs/1603.02701.

[16] S. Andreev, O. Galinina, A. Pyattaev, J. Hosek, P. Masek, H. Yanikomeroglu, and Y. Koucheryavy, "Exploring synergy between communications, caching, and computing in 5g-grade deployments", *IEEE Communications Magazine*, vol. 54, no. 8, pp. 60–69, Aug. 2016, ISSN: 0163-6804. DOI: 10.1109/MCOM.2016.7537178.