Poster: Wireless Network Functions in the Era of Low-Power IoT

Akshay Gadre

agadre@andrew.cmu.edu

Carnegie Mellon University, PA

Swarun Kumar

swarun@cmu.edu

Carnegie Mellon University, PA

ABSTRACT

Network Function Virtualization has demonstrated how general purpose infrastructure in the cloud can match the performance of specialized hardware while providing greater flexibility and scalability. Yet, specialized hardware remains the preferred implementation choice for the wireless physical layer. This is primarily due to the tight latency requirements and high bandwidth demanded by operations at the wireless PHY, as opposed to the network-layer.

This poster argues that in the era of low-power Internet of Things, virtualization of physical layer functions deserves a revisit. We specifically study low-power Internet of Things technologies that present a ripe opportunity for cloudification of wireless network functions, with their lower bandwidths and lax latency constraints. We present our vision of how physical layer function virtualization should be structured to maximize benefits and performance. We present a feasibility study on the latency and scalability limits of such virtualization for various physical layer functions that could leverage such an architecture.

ACM Reference Format:

Akshay Gadre and Swarun Kumar. 2019. Poster: Wireless Network Functions in the Era of Low-Power IoT. In *The 25th Annual International Conference on Mobile Computing and Networking (MobiCom '19)*, October 21–25, 2019, Los Cabos, Mexico. ACM, New York, NY, USA, 3 pages. <https://doi.org/10.1145/3300061.3343387>

1 INTRODUCTION

Network Function Virtualization (NFV) aims to achieve flexibility of deployment of network services by leveraging general purpose hardware in cloud to receive IP packets and process them in software. Prior to NFV, network functions relied on expensive and specialized hardware to perform network functions. Through the use of general purpose hardware

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

MobiCom '19, October 21–25, 2019, Los Cabos, Mexico

© 2019 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-6169-9/19/10.

<https://doi.org/10.1145/3300061.3343387>

Figure 1: Our approach seeks to push physical layer functions to the edge and cloud.

and the cloud, NFV greatly reduced the cost and improved flexibility and scalability of network functions. Today, the wireless physical layer in the IoT era is witnessing a similar fragmentation with specialized hardware developed for PHY-layer functions: modulation and coding, localization, PHY-layer security, to name a few. Indeed, while many cloud-based architectures have been proposed for next-generation wireless (e.g. SoftAir [1], OpenRadio [2], CloudRAN [3]), these tend to focus on management and orchestration in the wireless network. Complete cloudification of wireless PHY functionality is traditionally seen as a non-starter, given their stringent latency constraints and bandwidth requirements, often an order of magnitude tighter than network functions.

This poster argues for PHY-layer function virtualization in the context of low-power IoT technologies. We particularly focus on technologies that support low-data rates under stringent battery-constraints, such as LoRaWAN, Sigfox, Z-Wave, to name a few. We argue that these technologies have several desirable properties ripe for PHY-layer virtualization. First, low-power IoT technologies operate at small bandwidths which reduces the amount of information that needs to be backhauled. Second, the clients transmit rarely and have relatively lax latency and timing requirements.

Motivated by these observations, this poster presents an architecture for PHY-layer function virtualization for low-power IoT that encompasses the clients, gateways, edge and cloud. At a high level, our architecture uses software radio platforms equipped with FPGAs and limited local computation that collect and process wireless signals which are then selectively shipped to the cloud for additional processing. Much like traditional NFV, our approach relies on cloud

Figure 2: Proposed architecture detects packets at the SDR. These packets are forwarded to local and cloud services based on service constraints. SDR performs appropriate action as notified.

infrastructure to implement the PHY-layer, providing flexibility to upgrade software and scalability based on demand. However, different from NFV, we propose mechanisms to conserve backhaul bandwidth and minimize latency by seeking to performing as much computation locally as possible. In particular, we leverage local compute resources on the FPGA to dissect individual components of the packets and measure common parameters. This allows SDN routers to only communicate the required parameters to the cloud when collaboration across base stations can provide benefit to system performance, without compromising on latency. We further propose mechanisms to compress information shipped to the cloud and fetch additional information only when needed.

2 RELATED WORK

Related work on improving the wireless functionality in the cloud. **Cellular and 5G:** Management and Orchestration of Wireless Network[3][7]. **LP-WANs:** Cross Technology IoT Gateways[6][8][5]. **Hardware:** Radio hardware to support PHY-layer processing[2][4][6]. We present an architecture that enables virtualization of PHY-layer functionality building upon SDN/NFV concepts at the higher layers.

3 PHY-LAYER VIRTUALIZATION

Our architecture encompasses base stations, edge infrastructure and the cloud. Client devices are not modified to ensure backward compatibility. Our base stations are software-defined radios(SDRs) equipped with an FPGA and local compute resources for further processing. These base stations would detect packets from the wireless signals and generating parameterized packets for processing. These packets are then forwarded to a local SDN switch which will send them to appropriate decoder based on signal quality and expected service. Typically, services that span across base stations will

be handled in the cloud while others could be hosted locally. Upon decoding the packet, the local switch will task the SDR to perform appropriate action. In the following sections, we detail our design at the edge and the cloud.

3.1 Base Station and the Edge

Our base stations will be software-defined radios connected with local compute and storage resources. Each base station will be designed to perform packet detection, to avoid unnecessarily streaming I/Q samples to the cloud that do not carry any data. We further piggybank metadata such as RSSI, channel and timestamps along with the packet header. Note that since LP-WAN technologies have long preambles, the SDR FPGAs can compute such metadata with little overhead.

Our approach deliberately attaches metadata (e.g. RSSI) to the header to reuse the P4 framework at the physical layer. This will allow us to route packets at SDN switches based on PHY-layer parameters to appropriate modules that perform PHY-layer processing at the edge or cloud. For example, a frequency offset correction service requires data and the phase while a rate adaptation service may only be forwarded based on RSSI. These wireless network functions will typically be implemented in higher level languages such python and Matlab taking in packets and pushing out action tasks.

3.2 Cloud Services

The SDN switch in the cloud would forward the packets to the appropriate virtual machines based on the services required for the packet. Existing approach of NFV will allow virtual machines to provide software-isolation between different PHY-layer functions. Decoded packets undergo IP and higher-layer processing in the cloud. We note that several existing low-power architectures such as Sigfox and LoRaWAN already implement much of their network layer in the cloud.

An important question that remains is to determine which functions are at the edge and which are performed at the cloud as well as who makes this decision. Our approach is to perform as much PHY-layer computation locally as possible and provide decoded data to the cloud for higher-layer processing. The SDN switch would get forwarding rules from the controller based on metadata in the PHY-layer headers (RSSI, phase). A packet with good RSSI may be processed locally while one with worse may need to be collaboratively decoded in the cloud. These packets will be time-stamped at the base station to ensure correctness of services. If the packet needs to be served by the cloud services, it would be forwarded via an IP tunnel to the appropriate switch.

4 FEASIBILITY STUDY

Feasibility of the above architecture is explored with focus on three LP-WAN technologies: LoRa, Sigfox and NB-IoT and analyzes the latency and scaling requirements of different PHY-layer functions, should they be virtualized.

4.1 Latency Limits

Low-Power IoT typically have lax latency limits to allow for long packets sizes, long preambles and processing time. This avails more processing time at the cloud for our architecture before an acknowledgement needs to be sent for received packets. Below table highlights typical upper bounds for computation time available for different low-power IoT protocols. Note **80% of unlicensed LP-WAN clients are uplink only** negating requirement of latency constraints.

LP-WAN Technology	Compute Time Available
LoRaWAN(class C)	2s
Sigfox(both UL and DL)	45s
NB-IoT(IP ACK timeout)	1s

Another key latency constraint is the computation and communication delays required to perform various PHY-layer functions on commodity hardware. Typical cloud communication latency vary between **60-80 ms**. Finally, we implement a few of the publicly available functions in MATLAB on a server with the following configuration: Core i7 8th Gen, 32 GB RAM, GTX 1060Ti. Table below presents the typical computation times for PHY-layer functions (Code: <https://github.com/AkshayGadre/M2019>) implemented for LoRaWAN clients. Note some functions also require intensive GPU infrastructure which are typically available in the cloud.

Wireless Function	Computation Time(ms)
Matched Filter ¹	246
Phase Measurement	1.23
RSSI Measurement	1.25
DL based Security ²	50
Coherent Combining	2.52

4.2 Scalability Limits

We analyze the stress our architecture imposes on storage and network resources as the network scales. An important limiting factor that limits the scale of wireless function virtualization is the bandwidth required to carry these wireless signals to the virtual functions in the cloud. Fig. 3 shows the number of base stations that can be supported given the bandwidth of a base station and an off-the-shelf backhaul of 100 Mbps. Further, I/Q samples recorded from the base stations need to be stored in the memory of the VMs at the cloud for the duration of the wireless function. Thus, the maximum number of clients that can be served at a single cloud machine is a function of the number and size of packets. Fig. 3 shows the number of clients that can be stored on a machine with memory of 32 GB for a LoRaWAN packet of 55.25 chirps arriving at various configurations. Note that

¹This happens at the SDR in our architecture

²Authentication latency does not include training time.

Figure 3: Left: # client packets that VM with 32 GB memory can store. Right: # base stations supported on a 100 Mbps backhaul

real implementation will see upto ten-fold worse numbers due to memory usage by active virtualized functions.

While scaling is limited by the resources at cloud, recent studies[6] have exploited sparsity of signals to significantly improve this limit. Much of the spectrum in the 915 ISM band is reused by the various LP-WAN technologies where clients transmit every 15-20 mins most frequently. The presence of edge computing can reduce the necessary storage and bandwidth requirements at the cloud in large-scale networks.

5 CONCLUSION

This poster presents our vision of how a virtualized PHY-layer can greatly improve the flexibility and scalability of wireless networks. We particularly target low-power IoT networks where cloudification imposes minimal stress on the scalability and latency requirements of such networks. We present an architecture that leverages computing at the edge to improve system performance. We further provide a feasibility analysis of our architecture to target different PHY-layer functionalities across low-power IoT technologies.

ACKNOWLEDGMENTS

This work was supported by Kavčič-Moura award and National Science Foundation grants 1823235, 1718435 and 1837607.

REFERENCES

- [1] AKYILDIZ, ET AL. SoftAir: A software defined networking architecture for 5G wireless systems. *Computer Networks* (2015).
- [2] BANSAL, ET AL. OpenRadio: a programmable wireless dataplane. In *HotSDN* (2012), ACM.
- [3] CHECKO, ET AL. Cloud RAN for mobile networks – A technology overview. *IEEE Communications surveys & tutorials* (2014).
- [4] DONGARE, ET AL. Charm: Exploiting geographical diversity through coherent combining in Low-power Wide-area Networks. In *ACM/IEEE IPSN* (2018).
- [5] GADRE, ET AL. Frequency Configuration for Low-Power Wide-Area Networks in a Heartbeat. In *NSDI* (2020), USENIX.
- [6] KHAZRAEE, ET AL. SparSDR: Sparsity-proportional Backhaul and Compute for SDRs. In *ACM MobiSys* (2019).
- [7] LIANG, ET AL. Information-centric network function virtualization over 5G mobile wireless networks. *IEEE Network* (2015).
- [8] NARAYANAN, ET AL. Revisiting Software Defined Radios in the IoT Era. In *ACM HotNets* (2018).