7 Conclusion and Outlook

Fueled by the increasing popularity of wireless enabled mobile end-devices and the advent of the Internet-of-Things networks, the demand for wireless access technology is growing rapidly. However, supporting the ever increasing number of WiFi capable devices across residential, public, and enterprise networks is non-trivial. In particular, the (last) wireless hop is often critical for network performance, as it can contribute a non-negligible delay and may constitute a bandwidth bottleneck. Moreover, non-enterprise WiFi networks are often deployed in an unplanned and uncoordinated manner: different parties in a house or neighborhood typically deploy and run their own dedicated infrastructure; neighboring access points as well as public access points cannot be leveraged—but rather interfere with each other, introducing unnecessary transmission delays, and reducing network capacity. Also mobility support is often very limited, depriving users from essential services. Thus, future WiFi architectures are challenged by optimized medium utilization, mobility support, and network management. The latter challenge is integrating wired and wireless networks seamlessly. While today point solutions exist for some of the WiFi-specific network challenges, commodity off-the-shelf hardware is outside the purview of such ossified, expensive, and vertically integrated solutions.

Decoupling data plane and control plane operations, à la Software-Defined Networking, can greatly simplify network management and improve the overall performance and utilization of wired networks. However, SDN and NFV have not yet received as much attention in the context of wireless networks, due to fundamental differences between wireless and wired networks. First and foremost, wireless networks feature many peculiarities and knobs that often do not exist in wired networks. For example, wireless networks need interfaces for flexible resource allocation, client mobility, client-based load balancing, and fine/grained traffic engineering is paramount. Furthermore, today's trend towards Bring-Your-Own-Device (BYOD), implies that the network has to accommodate a more diverse set of user device types of different generations. Moreover, today's home networks, unlike enterprise networks, typically suffer from a non-existing dedicated control channel, rendering fine-grained centralized control challenging.

In this thesis, we show that there can be a major benefit of introducing programmability and virtualization in *wireless networks*, i.e., following an *Software-Defined Wireless Networking (SDWN)* approach. With our SDWN approach we combine the benefits of SDN and NFV with wireless access technology. We present novel abstractions that hide the complexities of the IEEE 802.11 protocol stack and allow network operators to manage their wired and wireless portion of the network in unison. We decouple the control and data plane to consolidate and outsource the control over a set of network devices including WiFi APs, switches, and routers to a logically centralized software controller. This allows the control plane to evolve independently of the data plane, enabling faster innovations. Moreover, we make the case for a functional decomposition of WiFi into its building blocks by following an NFV approach to *virtualize* network functions as software components running on generic compute resources and on programmable network devices. The resulting orchestration flexibilities can be used for a faster and cheaper service deployment. SDN can be exploited to steer flows through the appropriate network functions.

With our SDWN approach we overcome the aforementioned challenges in today's WiFi networks. We now summarize the key contributions and take-aways of each chapter individually, then give directions for future work in the area.

7.1 Summary

The key contributions of each chapter are summarized below.

Odin: An SDN framework that provides programmability and virtualization of the upper IEEE 802.11 MAC functionality with off-the-shelf commodity hardware.

Chapter 3 proposes a Software-Defined Wireless Networking framework targeting WiFi networks. With Odin, we present our novel Light Virtual Access Point (LVAP) abstraction, that addresses well the complexities of the IEEE 802.11 protocol stack, and a control plane that allows the orchestration of WiFi and wired networks in unison, by leveraging OpenFlow for the wired portion of the network. We show the benefit via six common network services realized as SDN Applications. Odin runs on top of *today's* commodity access point hardware without requiring client-side modifications, whilst being well-suited by design to take advantage of upcoming trends in physical layer virtualization and hardware extensions. Thus, with Odin, we present an solution to uniformly manage both wired and WiFi networks given the requirements of today's network operators. However, Odin addresses only one aspect of the envisioned flexible and programmable WiFi architecture. In particular, Odin does not provide fine-grained control over the WiFi data path or control over middleboxes.

OpenSDWN: A joint SDN and NFV framework that provides programmability of the WiFi datapath and per-client virtual middleboxes, to render network functions more flexible and support mobility and seamless migration.

Chapter 4 presents a flexible, novel WiFi architecture for home and enterprise networks based on a joint SDN and NFV approach. OpenSDWN implements per-client virtual access points and per-client virtual middleboxes, to render network functions more flexible and support mobility and seamless migration, *e.g.*, migrating firewall state between hotspots when performing a client handover. Moreover, OpenSDWN introduces IEEE 802.11 lower MAC (datapath) programmability to enable service differentiation and fine-grained transmission control, facilitating the prioritization of critical applications. Since the user is often left out of scope of this optimizations, we out-sources the control over the network to the user, application, or an Internet Service Provider through a *participatory interface*.

funcionalidad interesante ← para diferenciación de tráfico.

AeroFlux: A 2-tiered control plane that addresses the scalability aspects of an SDWN towards enterprise and ISP networks by exploiting locality in SDN control plane operations for scalability reasons, to tackle the risk of overloading the global logically centralised control plane.

Chapter 5 presents a 2-tiered approach for the design of a scalable wireless SDN control plane targeting large scale WiFi deployments. AeroFlux handles frequent, localized events close to where they originate, i.e., close to the data plane, by relying on *Near-Sighted Controllers*. Global events, which require a broad picture of the network's state, are handled by the *Global Controller* (`GC`). More specifically, `GC` takes care of network functions that require global visibility, such as mobility management and load balancing, whereas `NSC`s control per-client or per-flow transmission settings such as rate and power based on transmission status feedback information exported by the Access Points (AP). Put differently, we enable the global controller to offload latency-critical or high-load tasks from the tier-1 control plane to the `NSC`s. This reduces the load on the `GC` and lowers the latency of critical control plane operations. As a result, with AeroFlux, we realize a scalable wireless SDN architecture which can support large enterprise and carrier WiFi deployments with low-latency programmatic control of fine-grained WiFi-specific transmission settings. Another key feature of AeroFlux is that it does not require modifications to today's hardware and works on top of commodity WiFi equipment.

LegoFi: A functional decomposition of the WiFi architecture where WiFi function blocks are allocated where (and when) they are most useful.

Chapter 6 presents a modularized SDWN approach that is designed along the lines of SDN and NFV for WiFi networks. However, while the aforementioned architectures are important components of a future-proof WiFi architecture, there are additional opportunities to render the deployment of WiFi networks even more flexible. In particular, WiFi networks today provide a wide range of functionality related to performance and security. Today, these functionalities are often integrated and located at a single location, *i.e.*, either implemented on the AP or the control plane. Given the different requirements and characteristics of these functions, this is suboptimal. A functional decomposition of the WiFi building blocks can aid to improve the overall networks performance With LegoFi, WiFi function blocks are realized as virtualized and programmable Wireless Virtual Network Functions (WVNFs), and are allocated (and composed) where and when they are most useful. Specifically, through WVNFs, we achieve a functional decomposition of the WiFi architecture, allowing to overcome inflexibilities found in today's monolithic, vertically integrated and expensive WiFi architectures.

To wrap up, by orchestrating and modularizing WiFi along the lines of SDN and NFV we can overcome today's ossified WiFi architectures. Moreover, we present the necessary abstractions to introduce common features of enterprise networks to residential and hotspot deployments, *i.e.*, for WiFi networks based on off-the-shelf commodity hardware. The practicality of our approaches has been successfully demonstrated at several international conferences and are currently deployed and running in two WiFi access networks, *i.e.*, one university enterprise and one larger (30 household) residential network. Moreover, our contributions have gained commercial interest by network vendors and operators. Therefore, we believe that the contributions in this thesis constitute a relevant step forward to modern and future-proof WiFi networks.

7.2 Future Directions

This thesis encourages a multitude of directions as future work, both in terms of research as well as in terms of open source code development.

Long Term Research Directions: With LegoFi, we propose a radical and consequent approach to further flexibilize the WiFi architecture. While we have made a first step towards this vision, the topic deserves an in-depth study. In particular, as a first step towards such a study, an implementation and evaluation of an orchestrator may provide important insights into the design space. The questions in the scope of LegoFi are: What are the right interfaces between the virtual wireless function blocks? Which abstraction can be used for the functions blocks? How and where should the function blocks be placed? What are the design requirements to achieve a scalable logically centralized control plane? How and where should the control logic be placed, *e.g.*, on an intermediate node close to the datapath or in the cloud? Is it beneficial to realize a modularized control plane where control logic can be deployed in a distributed fashion?

In the direction of 5G there is a plethora of open research directions: Can we leverage the modularized concept of LegoFi for *5th generation mobile networks (also known as 5G)*? For instance, can we apply the `LVAP` concept to 5G networks? In other words, how can we leverage the `LVAP` abstraction for other wireless access technology? Thus, can we use `LVAPs` as a generic abstraction in SDWNs. Is the `LVAP` abstraction suitable for vertical handover, *i.e.*, can we leverage the `LVAP` abstraction for fixed-mobile convergence to perform state migration between cellular and WiFi? Can participatory networking help in cases such as QoS provisioning or mobile-edge cloud, *i.e.*, can the network provide resources such as computing and storage blocks?

`LVAPs` provide fast handover which enables seamless mobility. This seamless mobility is desirable for a couple of different network applications such as seamless WiFi access on highways, underground service, trains etc. Thus, further research in this direction can be to investigate the design requirements of an architecture that supports frequent handover at a large scale. In particular, how control traffic can be minimized in such environments?

The systems in this thesis do not fully support multi-tenancy. For instance, they do not support enough isolation and airtime fairness at the PHY layer. However, WiFi networks at scale should also allow to operate multiple tenants on top of the infrastructure. FlowVisor [126] or OpenVirteX [29] enable operators to create and manage virtual SDNs of multiple tenants on top of their own infrastructure. Specifically, OpenVirteX and FlowVisor act as OpenFlow proxies between an operator's network and the tenants' network controller to perform slicing etc. With these approaches tenants can specify their own topology and addressing schemes with performance isolation between slices. This challenge was not addressed yet in the scope of Software Defined Wireless Networking. Specifically, how to host a multitude of different tenants on top of a shared wireless infrastructure. How can we achieve