An Insight Into Software Defined Wireless Networks

Christian Niephaus, George Ghinea
Brunel University
London, United Kingdom
christian.niephaus@brunel.ac.uk
george.ghinea@brunel.ac.uk

Osianoh Glenn Aliu, Senka Hadzic, Mathias Kretschmer
Fraunhofer FOKUS
Sankt Augustin, Germany
osianoh.glenn.aliu@fokus.fraunhofer.de
senka.hadzic@fokus-extern.fraunhofer.de
mathias.kretschmer@fokus.fraunhofer.de

***Abstract*—The flexibility of future wireless networks architectures is aimed at allowing more innovation, reducing complexity and improving service offerings. Software Defined Networking (SDN) has been identified as an enabler for this adoption. In order to grasp a better understanding of the challenges faced as well as the potential uses cases, we identify the need for flexible software defined wireless network, its applications and challenges and propose an architectural framework.**

I. Introduction

Network operators are forced to embrace new services and revenue opportunities, but not necessarily new technologies that require overhauling their physical network. It is thus pertinent to give operators the freedom and flexibility to innovate and create new service offerings by providing them with a more flexible network architecture. This is the main driver for the evolution of today's communication networks towards being software based.

SDN was initially designed for infrastructure based wired networks. However, due to its huge potential, various use cases and testbeds for its application to wireless networks have been discussed. Having a programmable wireless network will be of benefit to developers, researchers and operators looking to use and provide applications of SDN in the wireless domain. We acknowledge that SDN may mean different things to different network operators and service providers from different technological domains. In the context of this paper, our reference to SDN is in the context of a network framework that allows network operators to intelligently manage and control their network in a flexible and simplified way using software based tools (high-level languages and APIs). This is achieved by a logical separation of application, control and data planes as well as abstraction of the open interfaces between them.

SDN advocates a new way of thinking network architectures in order to make them flexible and open for innovation. In principle, it proposes for design of future network architectures be based on the following principles [1], [2], [3], [4]:

1) Physical separation of data plane from control plane with open interfaces between them.
2) Protocol independent forwarding (rules-action model).
3) Logically centralized network control and management.
4) Programmable software based networks.
5) Network Virtualization.
6) Simplified generic forwarding devices.
7) Technology abstraction.

The aim of this paper is to first introduce Software Defined Wireless Network (SDWN) as a guide for further research that will be done in the wireless domain. Second, to present a logical flow of describing the need for flexible, software based wireless networks, challenges and benefits as well as an architecture for implementation. The rest of this paper is organized as follows: Section II gives a summary of current efforts in SDWN for Wireless Local Area Network (WLAN) and mobile cellular networks. In section III, the use cases and challenges peculiar for wireless networks are described. In Section IV, we present our proposed framework with description of modules in each plane. We conclude by giving pointers for future work required for adoption of SDWN in Sections V and VI.

II. Literature Review: Software Defined Wireless Networks

There has been extensive work and surveys on SDN in wired networks, networking in data centers and OpenFlow specifications in particular ([5], [6] and [7]). However, it has also been acknowledged recently that the principles of SDN also benefit wireless networks.

One of the pioneering papers on making wireless networks software defined in accordance with OpenFlow specification was presented by Yap et al. in [4]. OpenFlow Wireless builds on top of OpenFlow and serves to separate control and data, decouples mobility from the physical network, and is able to slice/virtualize the network using FlowVisor. Using this virtualization, multiple service providers can control the underlying infrastructure. Thus service providers can handle mobility, authentication and billing for their users, regardless of the network they are connected with. The authors also present Openroads in [8] to serve as an enabler for future innovation in wireless networks. The long term objective is to trigger virtual network operators and virtual service providers whose service delivery will be independent of the underlying physical infrastructure thus allowing transparent handover between heterogeneous wireless technologies. It is claimed that the OpenFlow specification as currently described for wired networks can also be used for control of traffic forwarding in the wireless network. However, as wireless channels usually require more parameter to configure enhanced SDN concepts can bring additional benefits, as we later describe in section III-B and IV.

Focusing on the data plane, authors in [9] first present key conceptual contributions for realization of a software defined cellular infrastructure. In order to achieve this, a reconsideration of the binding of wireless protocols to the processing and decision planes is proposed. A step towards achieving this is having a programmable data plane. The authors thus present OpenRadio: a programmable wireless data plane that also achieves a trade-off between flexibility and performance. This trade-off is a key challenge and still an open problem when considering cost constraints. The unique feature of OpenRadio is the software abstraction layer that enables a modular and declarative interface to program the physical and MAC layers.

For mobile wireless networks, the main characteristics of SDN in mobile operators architecture (using UMTS as an example) is described in [10] and shows the various interface interactions between modules. Using 3GPP Evolved Packet System (EPS) architecture, a proposed SDWN architecture was presented using a mobile operator's network. In this architecture, the SDN controller is a logical entity that can be decentralized into different physical boxes to improve scalability and performance.

For enterprise WLAN, a framework aimed at allowing network operators implement WLAN services as network applications is presented in [11] called Odin. Its abstraction is implemented via a light virtual access points (LVAP) which provides logical isolation of clients. Thus every client can be seen as having a fixed link to the WLAN infrastructure. LVAPs allow clients to always see the same virtual Access Point (AP), regardless of the actual physical AP the client is associated to. It however requires multiple Odin agents to run on the physical access points.

Based on the architectures reviewed, it is clear that operators have to employ different architectures depending on whether they want to support short range communication, cellular, outdoor WLAN, hybrid networks, or even satellite networks, which might play an important role in future networks with respect to ubiquitous service coverage as explained in [12]. It must be understood that no single architecture would provide the best performance and address all challenges associated. A unified but flexible framework is thus a valid option where different architectures can be developed (while keeping to the fundamental SDN principles) to suit unique or special set of use cases.

III. Use Cases and Challenges

This section first describes our vision of SDWN as well as the benefits and impacts SDN will have on wireless networks. We also discuss the challenges that have been identified for its implementation. We further note that some of these challenges have been addressed while others still remain as open problems.

We believe that SDWN extends the SDN paradigms into the wireless world by providing a generalized interface which is not limited to managing data flows in the network above the MAC layer but also allow for programability of physical parameters of an interface. For example, while in the wired world an SDN forwarding device might have a rule to forward all packet belonging to the same flow to another Ethernet port, on a wireless device the rule might be to forward the packets on a particular channel using a specific modulation and coding scheme. This SDWN approach extends the programability of regular SDN to MAC and PHY OSI layers and therefore provides a holistic and flexible interface to control a wireless network.

A. Use Cases of SDN in Wireless Networks

We provide an overview of possible benefits the introduction of programmable, flexible software define networking will have across all layers in the wireless domain (Figure 1). We structure these benefits as impacts into two areas: Services (users and service providers) and Network Operators (including equipment vendors).

1) Increased Operator Revenue: There is a rise in virtual network operators to meet growing service demands and posing new business models. The key challenge for service providers is to minimize the CAPEX and OPEX required. SDWN based architectures enable infrastructure sharing among service providers as they share the expenditure thus creating new business models to improve their profit margins. It also serves as an enabler in meeting the 5G requirement of reduced service creation time from the current average of 90 days to as low as 90 minutes. This is feasible as new service offerings and capabilities only need to be pushed via software updates to all nodes to support new services.

2) Flexible Traffic Steering: Using information of location specific capacity requirements, flexible traffic steering schemes can be implemented to achieve various operator objectives. Having the knowledge of the network state, the controller can steer the traffic and thus optimize bandwidth utilization. Flow optimization within the backhaul can also be implemented with the flexibility SDWN provides. It enables the possibility for smarter load balancing and flexible wireless backhaul with QoS based routing. This is achieved due to the support for media independent handover and protocol independent forwarding.

3) Interoperability: Equipment from multiple vendors can be managed and controlled using SDWN. Instead of managing groups of devices from particular vendors, mobile operators can use software based management tools to set up, configure, and manage networks in multi-vendor environments via their APIs. Seamless interoperability among heterogeneous networks without the need for a proprietary gateway at each Point of Interconnect (PoI). This is made possible due to

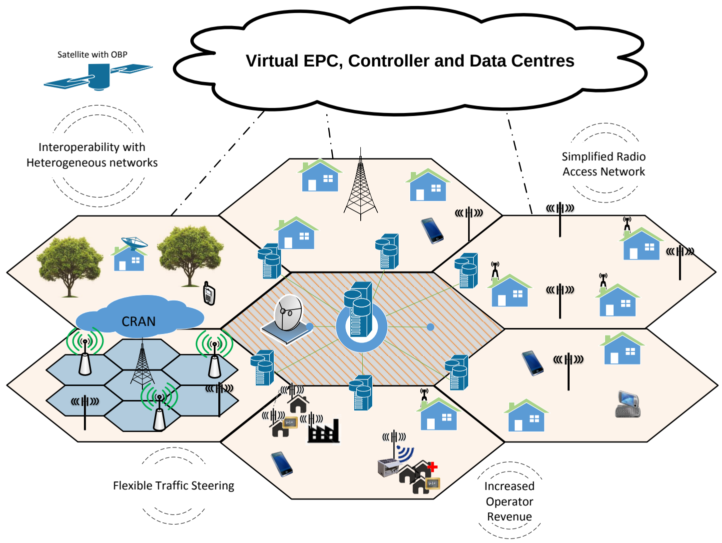

Fig. 1. Overview of Uses Cases with SDWN

abstraction layer and open interfaces at PoI. Moreover, not being limited to a single vendor further decrease the cost for an operator.

4) Simplified RAN: The concept of having generic forwarding devices is applicable in cellular networks where computationally complex processing may be offloaded from the base stations to the cloud. This means reduction in complexity and cost of future 5G RAN.

In spite of the aforementioned use cases, it must be understood that SDN is not the magic wand that will provide solutions to all challenges in wireless communications. SDN should be viewed as a tool that empowers us with the much needed flexibility as well as removing constraints in today's network architectures in order to innovate new solutions for existing problems and future challenges in wireless networks.

B. Challenges of SDN in Wireless Networks

Based on the literature review and from experience during implementation of centrally managed wireless backhaul heterogeneous networks ([13], [14], [15]), we identified the following challenges that have to be addressed in designing an efficient SDWN. The challenges listed are not exclusive to wireless networks or SDN but are even more critical when considering SDWN.

1) Decentralized bootstrap modules: Current SDN implementations advocate for all decision based modules to be centralized in the controller. However in wireless networks, establishing initial connectivity requires decisions made at the local node which implies having those modules centralized not ideal. Consider a multihop wireless backhaul network for providing internet connectivity in rural areas, e.g. [16], consisting of nodes that are only connected via wireless links with each other. When even operating in an unlicensed band a node need to perform locally a neighborhood scan in order to find adjacent nodes to connect to and, if multiple neighboring nodes are available, to select the most suitable one. This is required to establish the initial connection to the controller, which afterwards might reconfigure nodes based on its global knowledge.

2) Changing network topology and Traffic Overheads: When nodes join or leave a network, varying channel conditions, shadowing or optimization of handover policies, topology changes are triggered. Service continuity during reconfiguration due to topology change is non-trivial as well as minimizing the associated overheads due to control and signalling messages. Furthermore, it becomes more challenging to obtain real-time link state information at all interfaces of connected nodes at a single centralized point without significant overheads in updated rules in flow tables due to changes in network topology. Authors in [17] and [14] suggest implementing new messaging formats for the control traffic. Further work is however still required to demonstrate compressed messaging formats for network managing protocols and SDN based schemes that still allow the forwarding devices to update their flow tables only due changes in wireless link.

3) Control Signalling: Control signalling between controllers, forwarding devices and between a controller and a forwarding device may be implemented via a wireless link. However, in most wireless networks (e.g. IEEE 802.11 based networks), the luxury of having a dedicated RF channel for control signalling is unlikely. Nevertheless, the option of using in-band signalling could lead to increased latency for data traffic on the same channel and decreased network throughput should not be ignored in low rate wireless networks.

4) Unified network architecture: Based on the generic SDN framework of separating the application, control and forwarding layers, various architectures have been proposed showing different functional modules and their ideal locations in different layers. With parallel activities by Open Networking Foundation (ONF), OpenDaylight and European Telecommunication Standards Institute (ETSI), there is no unified architecture. The challenge here is that no single architecture is

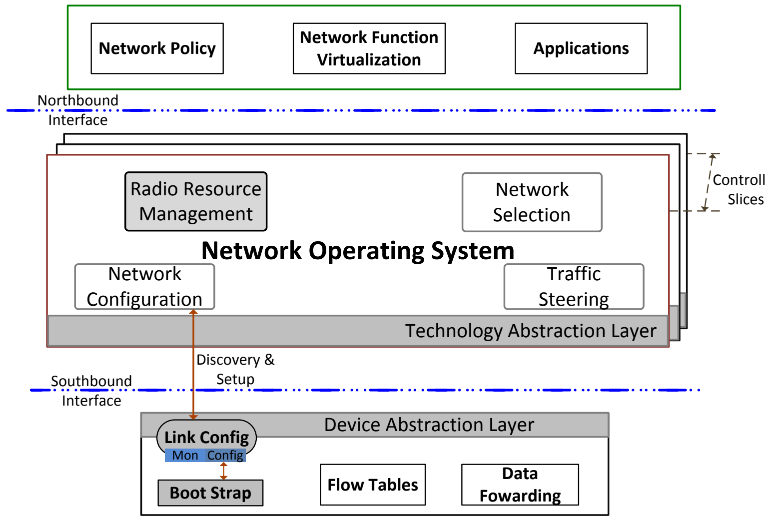

Fig. 2. SDWN Logical Framework

suitable to address all use cases in the wireless domain due to nature of multi-technologies and need to ensure compatibility with existing networks. This is required to help operators manage the growing complexity of network architectures and to support interoperability. In [9], a modular declarative interface is proposed for programming the data plane irrespective of the technology used at this layer. A similar modular architecture is also proposed by [14] to ensure the network can be adapted for different use cases by connecting existing modules based on a simple and extensible network framework. What is now required is a universal framework for SDWN where operators can adapt based on the use cases being addressed.

IV. Unified Software Defined Wireless Networking Framework

Wireless network architectures have continued to evolve over the past few decades. This emanated from the fact that architectures were first designed to support only voice applications but later evolved to support voice and data. New architectures were yet proposed to effectively support mobility and data traffic as the main traffic requirement. As requirements for future 5G architectures are being discussed, it should be noted that architectures designed to address specific use cases in the future will become suboptimal when new use cases or killer apps that cannot be accurately envisaged, begin to emerge. What is thus required is providing flexible, programmable, modular frameworks that can easily be adapted by network operators as new service opportunities emerge. A generic framework for achieving this is directly inspired by the SDN framework with specific adaptions for wireless networks.

The key design principle however is choosing the right level of abstraction in separation of the control plane form the dataplane. In this section, we thus provide a framework upon which specific architecture implementations can be used depending on the use case considered. Figure 2 shows our proposed SDWN logical framework directly inspired from the framework used in SDN. A description of the key modules is provided as follows:

A. Service Plane

This layer is responsible for adding and managing new features as desired by the network operator to the network. This includes new applications, network features and policies. It may also be understood as a business application layer when new service offerings can be pushed to users in the network with fast service creation time, e.g. location based services. Separation of the service and control plane greatly reduces the time to market new services. The service layer for SDWN is inherited from the architecture defined by the ONF in [18]. It can however be extended to include specific features required for managing wireless networks such as Operation Administration Management and Provisioning (OAMP) and service differentiation offerings. The interface between the service layer (commonly referred to as the application layer) and the control layer is the northbound interface.

B. Control Plane

In wireless networks, the control plane consists of more than just a single physical controller or network entity. This is the core of the network architecture that serves as a platform for the network operating system. It orchestrates the traffic forwarding and signaling behavior of the network via its interface to the lower layers. Provisioning of network services, defining forwarding rules, traffic routes and radio resource management is enforced at this layer. The control plane is responsible for enforcing polices defined in the service plane. As regards to topology of the underlying infrastructure, this plane also incorporates functionalities for load balancing and Network Function Virtualization (NFV). In wireless networks, especially in infrastructure-less mobile adhoc wireless networkss, topology management, including mobility, is key due to new nodes joining and leaving the network on a more frequent time scales.

Depending on the type of technology used in the wireless network, the control plane would consist of different network elements. For example in cellular networks its main network elements will be located in the core network, in IEEE 802 wireless[1] networks as a master node or controller managing a set of wireless access points. The framework shown in Figure 2 is a logical representation. In practice, it is expected to utilize physically distributed instances of controllers managing different slices of the network while synchronization of all functionalities remains at a centralized point in this plane. The key modules which are technology agnostic which should be implemented on this plane include network selection, network configuration for managing forwarding devices on the data plane, and traffic routing for path computation and enforcing routing policies.

C. Data Plane

The final layer responsible for data forwarding and connecting to the end user device is the data plane. SDN concept advocates for devices that are simplified, low cost, minimal processing but specialized for forwarding packets. In essence, the devices here should primarily receive forwarding rules from the control plane and take actions based on a set of pre-configured traffic routes stored in its flow tables. Statistics of the network are periodically sent back to the control plane for optimizing forwarding rules.

The peculiarity of the data plane in wireless networks requires configuration of the wireless links between forwarding devices, as well as the wireless link to their controller. It is thus pertinent to include a link configuration module in forwarding devices specifically for link monitoring and configuration. This includes simple localized functions, particularly for modules that cannot be performed efficiently from a physically centralized point. In essence, a boot strap phase will be used for initial setup connection and configuration of the device, as already mentioned previously.

In cellular networks for example, the forwarding device can be viewed as the nodes in the access network (eNodeB and Femtocell in LTE). In IEEE 802.11 based networks, the forwarding devices are the wireless access points. The interface between these forwarding devices and their gateway or controller in the control plane is referred to as the southbound interface.

D. Southbound Interface

To ensure interoperability between vendors as well as to support heterogeneous networks, a standardized interface is specified to describe communication between devices on the data plane and network elements in the control plane. This communication is done over a secure channel and the main objective is to manage the forwarding devices or nodes, including device configuration. Various protocols can be used to achive this, such as Forwarding and Control Element Separation (ForCES), Network Configuration Protocol (NETCONF), SNMP4SDN (an extension of SNMP), Interface to Routing System (I2RS) and Path Computation Element Communication Protocol (PCEP). Protocols used for secure control-data plane communications vary based on the type of network and use case being considered. OpenFlow is currently the most widely adopted but as it is just one protocol on the southbound interface, it is important to ensure the southbound interface remains ”Open” for other protocols (exisitng and new) in a non-proprietary way.

E. Northbound Interface

Currently, there is no standardized interface describing the clear separation between the service (application) plane and the control plane. However a major project that can be applicable for the wireless domain and service providers due to their service abstraction layer is Pyretics, which was created as part of the Frenetic project [19]. It enables network operators to write modular applications by providing high level abstractions using Python scripts. Pyretics translates network policies to functions and uses parallel and sequential composition of controller applications that are required on the same type of traffic. Besides Pyretics, secure communication over the northbound interface can also be implemented using Representational State Transfer (REST) APIs initially designed for abstraction of the World Wide Web, Open Service Gateway initiative (OSGi) protocol as well as various sub-projects developed in the OpenDaylight community. It should be noted that at this nascent stage, most deployments currently incorporate applications and features of the service plane into the control plane.

V. Future Research Directions

We present existing challenges in the form of questions that need to be addressed in order for full adoption of SDN paradigms in the wireless domain.

1) Performance, Flexibility or Cost?: For network operators, monetary cost may be their most important KPI while equipment vendors may favor performance of their solution. However, the importance of flexibility in terms of system architecture to provide new solutions with reduced service creation time brings in a new dimension. It is thus necessary to investigate the tradeoff and Pareto optimal point between performance, flexibility and cost. The impact of having full flexibility and freedom to develop technology agnostic, protocol independent packet processing solutions needs to be accessed.

2) What is lost by making interfaces generic?: Due to the heterogenous nature of wireless networks, the southbound interfaces deals with more compatibility issues both from technologies and various protocols within a specific technology as compared to wired networks. By defining specifications for open generic intefaces, what specilised networks functionalities may be lost? Do such heterogenous networks with common set of primitives have higher performance than existing propritary implementations? Quantitave comparisons are required to demonstrate the benefits of using open, generic interfaces based on unique use cases and deployments.

[1]WiFi, WLAN, WiMAX, etc

VI. Conclusion

It is clear that SDN will have a significantly impact on wireless networks. However, there exists a fundamental trade-off between achieving flexibility in the network and optimum performance, which leads to the consideration of different architectures. We highlighted use cases based on the impact its adoption will have on service providers and end users. Furthermore various other impacts on network operators and equipment vendors were listed. As with any new solution, SDWN is also faced with a couple of challenges. Most of these challenges are due to the nature of wireless propagation channel and additional required features such as radio resource management. Our reference to initial attempts to address these challenges [9], [13], [14], [19] indicates that they will not be a major barrier against its successful adoption.

Based on the literature review of existing architectures, we have come to the conclusion that no single architecture will be adequate for all use cases and deployments. We thus describe a logical framework for software defined wireless networks which addresses some of the challenges previously identified. The main impact is addressing the general question of how centralized decision making modules have to be for wireless networks. We realized that certain controller functions have to remain at the forwarding devices, i.e. initial channel scan, link status neighbor sensing. This is especially important during the bootstrap phase. Further description of a technology abstraction layer and device abstraction layer were described with references for more detailed technical description.

We have presented an overview of the need, use cases, challenges and architectural framework for SDWN. The research community, operators as well as equipment vendors better understand the fundamental principle of SDN and its adoption in wireless networks. Software defined wireless networks will certainly create the required enablers for the next generation of wireless networks and our case study for wireless backhaul networks is presented in [13].

Acknowledgment

This work has been supported by the BATS research project which is funded by the European Union Seventh Framework Programme under contract n317533. The views and conclusions contained here are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the BATS project or the European Commission. The authors would like to thank all non academic staff in the research community. This work was partially carried out during the tenure of an ERCIM "Alain Bensoussan" Fellowship Programme. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n 246016.

References

[1] N. Feamster, J. Rexford, and E. Zegura, "The road to SDN," *Queue*, vol. 11, no. 12, pp. 20:20–20:40, Dec. 2013.

[2] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: Enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

[3] N. McKeown, "Software-defined networking," *INFOCOM keynote talk*, 2009.

[4] K.-K. Yap, R. Sherwood, M. Kobayashi, T.-Y. Huang, M. Chan, N. Handigol, N. McKeown, and G. Parulkar, "Blueprint for introducing innovation into wireless mobile networks," in *Proceedings of the Second ACM SIGCOMM Workshop on Virtualized Infrastructure Systems and Architectures*, ser. VISA '10. New York, NY, USA: ACM, 2010, pp. 25–32.

[5] A. Lara, A. Kolasani, and B. Ramamurthy, "Network innovation using openflow: A survey," *Communications Surveys Tutorials, IEEE*, vol. 16, no. 1, pp. 493–512, First 2014.

[6] Y. Jarraya, T. Madi, and M. Debbabi, "A survey and a layered taxonomy of software-defined networking," *Communications Surveys Tutorials, IEEE*, vol. PP, no. 99, pp. 1–1, 2014.

[7] B. Nunes, M. Mendonca, X. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *Communications Surveys Tutorials, IEEE*, vol. PP, no. 99, pp. 1–18, 2014.

[8] K.-K. Yap, M. Kobayashi, R. Sherwood, T.-Y. Huang, M. Chan, N. Handigol, and N. McKeown, "Openroads: Empowering research in mobile networks," *SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 1, pp. 125–126, Jan. 2010.

[9] M. Bansal, J. Mehlman, S. Katti, and P. Levis, "Openradio: A programmable wireless dataplane," in *Proceedings of the First Workshop on Hot Topics in Software Defined Networks*, ser. HotSDN '12. New York, NY, USA: ACM, 2012, pp. 109–114.

[10] C. Bernardos, A. de la Oliva, P. Serrano, A. Banchs, L. Contreras, H. Jin, and J. Zuniga, "An architecture for software defined wireless networking," *Wireless Communications, IEEE*, vol. 21, no. 3, pp. 52–61, June 2014.

[11] L. Suresh, J. Schulz-Zander, R. Merz, A. Feldmann, and T. Vazao, "Towards programmable enterprise wlans with odin," in *Proceedings of the First Workshop on Hot Topics in Software Defined Networks*, ser. HotSDN '12. New York, NY, USA: ACM, 2012, pp. 115–120.

[12] C. Niephaus, M. Kretschmer, and G. Ghinea, "Towards qos-aware load distribution in heterogeneous networks," in *Communications (MICC), 2013 IEEE Malaysia International Conference on*, Nov 2013, pp. 151–156.

[13] C. Niephaus, O. Aliu, S. Hadzic, M. Kretschmer, and G. Ghinea, "WiBACK: A back-haul network architecture for 5G networks," *IET International Conference on Frontiers of Communications, Networks and Applications (IET ICFCNA)*, vol. Accepted for publication, 2014.

[14] M. Kretschmer, P. Batroff, and G. Ghinea, "Topology forming and optimization framework for heterogeneous wireless back-haul networks supporting unidirectional technologies," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 698 – 710, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S108480451200269X

[15] M. Kretschmer, T. Horstmann, P. Batroff, M. Rademacher, and G. Ghinea, "Link calibration and property estimation in self-managed wireless back-haul networks," in *2012 18th Asia-Pacific Conference on Communications (APCC),*, Oct 2012, pp. 232–237.

[16] C. Niephaus, M. Kretschmer, and K. Jonas, "Qos-aware wireless back-haul network for rural areas in practice," in *Globecom Workshops (GC Wkshps), 2012 IEEE*, Dec 2012, pp. 24–29.

[17] R. Bin Rais, M. Mendonca, T. Turletti, and K. Obraczka, "Towards truly heterogeneous internets: Bridging infrastructure-based and infrastructure-less networks," in *Communication Systems and Networks (COMSNETS), 2011 Third International Conference on*, Jan 2011, pp. 1–10.

[18] Open Networking Foundation, "OpenFlow wireless and mobile working group."

[19] J. Reich, C. Monsanto, N. Foster, J. Rexford, and D. Walker, "Modular SDN programming with pyretic," *USENIX; login,*, vol. vol. 38, no. 5,, October 2013.