CONTENTS

Page

INTRODUCTION 1

CHAPTER 1 STATE OF THE ART 17
1.1 Wireless Network Virtualization 17
1.1.1 SDN for WiFi Networks 19
1.1.2 SDN for Wireless Sensor Networks (WSNs) 20
1.1.3 SDN for Cellular Networks 21
1.1.4 SDN for heterogeneous Networks 24
1.1.5 Virtualization without SDN 26
1.2 Programmable Radio Plane 26
1.2.1 Programmable Front-end 27
1.2.2 Programmable nodes and spectrum sharing 28
1.3 Cloud Computing for Wireless Networks 31
1.3.1 Cloud solution for Cellular Networks 31
1.3.2 Cloud solution for Heterogeneous Networks 34
1.4 Full Duplex (FD) systems 38

CHAPTER 2 DESIGN OPTIMIZATION OF WIRELESS ACCESS VIRTUALIZATION BASED ON COST & QOS TRADE-OFF UTILITY MAXIMIZATION 41
2.1 Introduction 41
2.2 Traditional Heterogeneous Network (HetNet) 44
2.2.1 HetNet dimensioning 45
2.2.2 HetNet cost analysis 45
2.2.3 LTE-TDD configuration 48
2.3 Locally Virtualized Network (LVN) 49
2.3.1 LVN architecture 50
2.3.2 LVN dimensioning 52
2.3.3 LVN cost analysis 53
2.4 Clustered/Remote Virtualized Network (CVN/RVN) 54
2.4.1 Network orchestrator (NO) 55
2.4.2 Radio access network (RAN) 57
2.4.3 Core network (CN) 59
2.4.4 CVN/RVN dimensioning 60
2.4.5 RRHs cost 61
2.4.6 CPC cost 62
2.4.7 Total CVN/RVN cost 65
2.5 Hybrid Virtualized Network (HVN) 66
2.6 Data Rate and Utility Function Construction 67
2.7 Results 70

2.7.1 Optimum CVN/RVN CPC size d_{cpc} 71
2.7.2 CVN/RVN utility U_{opt} at different GP values 72
2.7.3 Optimum network utility U_{opt} of HVN for different GP values 74
2.7.4 Comparison of optimal network utility U_{opt} for different frameworks 74
2.7.5 Optimal CVN network coefficient p_c^{opt} vs. cost weight w_c and optimal CPC radius $\bar{d}_{cpc}^{opt}$ 75
2.8 Conclusions 77

CHAPTER 3 END-TO-END PROGRAMMABLE, CLOUD-BASED VIRTUALIZED HETNET: AN INTEGRATED ARCHITECTURE 79
3.1 Introduction 79
3.2 Requirements of Programmable Virtual Wireless Networks 81
3.2.1 Virtual Network (VN) Isolation 81
3.2.2 End-to-end Programmability 82
3.2.3 On demand resource provisioning 82
3.2.4 Network Function Virtualization (NFV) 83
3.2.5 Dynamic Spectrum Sharing 83
3.3 End-to-end programmable, elastic, HVWN 84
3.3.1 Management and Orchestration layer 86
3.3.2 Service layer 87
3.3.3 Application layer 88
3.3.4 SDN layer 89
3.3.5 Baseband signal processing 90
3.3.6 High capacity front-haul 91
3.3.7 Programmable virtualized radio nodes 91
3.3.8 Wireless spectrum virtualization 92
3.4 Business Cases for Programmable Heterogeneous Virtual Wireless Networks 92
3.4.1 Equipment manufacturers 93
3.4.2 Infrastructure providers (InPs) 93
3.4.3 Mobile Virtual Network Operators (MVNOs)/ Service Providers (SPs) 94
3.4.4 Third-party software companies 94
3.4.5 Inter ISP-VNO traffic offloading 95
3.5 Potential Research Issues & Challenges 95
3.5.1 Standardization of APIs 95
3.5.2 Balance between flexibility and complexity 96
3.5.3 Security threats minimization 96
3.5.4 Virtualization of wireless spectrum 97
3.5.5 Definition of isolation 97
3.5.6 Integration of Cognitive Radio (CR) 98
3.5.7 Backward compatibility 98

3.6 Conclusions 98

CHAPTER 4 SERVICE DIFFERENTIATION IN SOFTWARE DEFINED VIRTUAL HETEROGENEOUS WIRELESS NETWORKS 101
4.1 Introduction 101
4.2 Related work 102
4.3 HetNet Cloud architecture 102
4.3.1 Application layer 104
4.3.2 Software modules 104
4.3.3 Northbound API 104
4.3.4 NOS & East-Westbound API 105
4.3.5 Virtualization layer 106
4.3.6 Southbound API 106
4.3.7 InP's resource management layer 106
4.3.7.1 Network Orchestrator (NO) 106
4.3.7.2 Resource broker (RB) 107
4.3.8 Baseband processing 107
4.3.9 Radio access plane 107
4.4 Using northbound API to facilitate virtual wireless network management 108
4.4.1 Interference management 110
4.4.2 Mobility management 111
4.4.3 Traffic offloading in a HetNet eco-system 113
4.4.4 Secured network 115
4.4.5 Internet of things (IoT) 116
4.5 Service differentiation in heterogeneous wireless networks 117
4.6 Challenges 120
4.7 Conclusions 122

CHAPTER 5 DEPLOYMENT OF FULL DUPLEX MULTI-CELL SYSTEMS FOR DENSE URBAN AND RURAL ENVIRONMENTS 125
5.1 Introduction 125
5.1.1 FD single cell deployment 126
5.1.2 FD multi-cell deployment 127
5.2 User selection and scheduling 131
5.2.1 Selecting users 133
5.2.2 Centralized Scheduling 134
5.2.3 Distributed scheduling 135
5.3 Dense Urban Model: Madrid Grid (MG) 138
5.3.1 Result analysis 139
5.3.1.1 User rate vs. SIC 141
5.3.1.2 System throughput vs. fairness 144
5.3.1.3 Node activity 145
5.4 Hexagonal grid (HG) model 148
5.4.1 Result analysis 150

5.4.1.1 User rate vs. SIC 150
5.4.1.2 System throughput vs fairness 153
5.4.1.3 Node activity 154
5.5 Conclusions 156

CONCLUSION 159

BIBLIOGRAPHY 165

LIST OF TABLES

Page

Table 0.1 **List of publications from this thesis** 16

Table 1.1 Proposals for Software Defined Wireless Networks............................ 25

Table 1.2 Representative summary of work on programmable radio plane 30

Table 1.3 Representative Summary of Cloud-Based wireless Networks 37

Table 2.1 Traditional network BS parameter 48

Table 2.2 Special subframe configuration for normal CP 49

Table 2.3 RRH parameter... 66

Table 2.4 CPC cost parameter 66

Table 2.5 Evaluation scenarios 71

Table 4.1 Delay and achievable throughput for static users.............................120

Table 4.2 Delay and achievable throughput for mobile users 121

Table 5.1 Simulation parameters for MG model 138

Table 5.2 Madrid grid: JFI vs. sum throughput 145

Table 5.3 UE to UE pathloss parameter..149

Table 5.4 Simulation parameter for HG..150

Table 5.5 Hexagonal grid: JFI vs. sum throughput (co-located antenna interference cancelled by 80 dB) 153

Table 5.6 Hexagonal grid: JFI vs. sum throughput (co-located antenna interference cancelled by 120 dB)..154

LIST OF FIGURES

Page

Figure 0.1 A block representation of EPS 7

Figure 0.2 Simplified representation of a SDN architecture 9

Figure 0.3 Different models of cloud infrastructure 11

Figure 2.1 Bolck diagram of a multi-RAT SBS 50

Figure 2.2 Clustered Virtualized Network (CVN) Architecture 55

Figure 2.3 Flow chart showing a NO's decision steps 56

Figure 2.4 Functional block representation of a CVN/RVN RAN with a network orchestrator 58

Figure 2.5 CVN/RVN core network flow diagram 60

Figure 2.6 Opt. CPC size $d_{cpc}^{opt}(GP)$ vs. cost weight w_c for diff. GP values in the ref. scenario I (cf. Table 2.5) 71

Figure 2.7 Absolute optimum CPC size $\bar{d}_{cpc}^{opt}$ vs. cost weight w_c in different scenarios (cf. Table 2.5) 72

Figure 2.8 Opt. network utility U_{opt} vs. cost weight w_c for diff. GP values in ref. scenario 1 (cf. Table 2.5) 73

Figure 2.9 HVN optimal network utility, U_{opt}, vs. cost weight w_c for different GP values in reference scenario I (cf. Table 2.5) 74

Figure 2.10 U_{opt} vs. cost weight w_c in different scenarios (cf. Table 2.5) 75

Figure 2.11 Optimal CVN network coefficient p_c vs. cost weight w_c in different scenarios (cf. Table 2.5) 76

Figure 2.12 Opt. CPC radius $\bar{d}_{cpc^{opt}}$ vs. CVN network coefficient p_c vs. cost weight w_c in scenario I (cf. Table 2.5) 77

Figure 3.1 Heterogeneous Virtual Wireless Networks Scenario 86

Figure 3.2 Functional blocks of end-to-end programmable heterogeneous virtualized wireless networks 87

Figure 3.3 Flow diagram of operational steps of the management and orchestrator layer during a VNO request for setting up a VN 88

Figure 4.1 A HetNet Cloud architecture consisting of distributed WDCs 103

Figure 4.2 Schematic of a WDC processing blocks in HetNet Cloud 105

Figure 4.3 OpenFlow packet model and the modified packet model with virtual fields for virtualization...108

Figure 4.4 Multi-layer virtual network protocol stack 109

Figure 4.5 Interference management message exchange between Local Controllers belonging to the same VNO.....................................112

Figure 4.6 Message exchanges for traffic offloading between VNOs 115

Figure 4.7 Virtaul wireless networks emulation scenario 117

Figure 5.1 HD and FD single cell scenarios 127

Figure 5.2 HD and FD multi-cell scenarios...129

Figure 5.3 Madrid grid model layout 139

Figure 5.4 Madrid grid: antenna radiation pattern for Macro and Pico BSs 140

Figure 5.5 Madrid grid model: user rate vs. SIC for HD and FD systems (centralized and distributed scheduling) 142

Figure 5.6 Madrid grid model: CDF of user rate when only SIC is employed 143

Figure 5.7 Madrid grid model: CDF of user rate when SIC + *IBIC* = 30 dB is employed 144

Figure 5.8 Madrid grid model: centralized and distributed scheduling sum throughput vs JFI 146

Figure 5.9 Madrid grid model: centralized scheduling sum throughput vs JFI 146

Figure 5.10 Madrid grid model: distributed scheduling sum throughput vs JFI 147

Figure 5.11 Madrid grid model: FD node activity 147

Figure 5.12 Hexagonal grid model 149

Figure 5.13 Pathloss curve fitting from Madrid grid model 150

Figure 5.14 Hexagonal grid: geometric mean of user rate vs. SIC 151

Figure 5.15 Hexagonal grid: CDF of user rate when only SIC is applied 152

Figure 5.16 Hexagonal grid: CDF of user rate when SIC and a *IBIC* = 30 dB is applied 152

Figure 5.17 Hexagonal grid: centralized and distributed scheduling sum throughput vs. JFI 155

Figure 5.18 Hexagonal grid: centralized scheduling sum throughput vs. JFI 155

Figure 5.19 Hexagonal grid: distributed scheduling sum throughput vs. JFI 156

Figure 5.20 Hexagonal grid model: FD node activity 157

LIST OF ALGORITHMS

5.1 User Selection: Centralized Scheduling 135

5.2 User Selection: Distributed Scheduling.. 136

5.3 GetMetric(active users, W) 137

LIST OF ABREVIATIONS

BS	Base Station
eNB	Evolved Node B
CAPEX	Capital Expenditures
CDF	Cumulative Distribution Function
CN	Core Network
CPC	Central Processing Center
C-RAN	Cloud Radio Access Network
CVN/RVN	Clustered/Remote Virtualized Network
D-RAN	Distributed Radio Access Network
EPC	Evolved Packet Core
EPS	Evolved Packet System
FD	Full Duplex
GP	Guard Period
HD	Half Duplex
HVN	Hybrid Virtualized Network
HVWN	Hybrid Virtual Wireless Network
InP	Infrastructure Provider
IoT	Internet of Things
ISP	Internet Service Provider

JFI	Jain’s Fairness Index
LTE	Long Term Evolution
LVN	Locally Virtualized Network
MVNO	Mobile Virtual Network Operator
NFV	Network Function Virtualization
NO	Network Orchestrator
NOS	Network Operating System
OFDMA	Orthogonal Frequency Division Multiple Access
OPEX	Operational Expenditures
QoS	Quality of Service
RAN	Radio Access Network
RANaaS	Radio Access Network as a Service
RAT	Radio Access Technology
RB	Resource Broker
RRH	Remote Radio Head
SBS	Super Base Station
SDN	Software Defined Network
SI	Self Interference
SIC	Self Interference Cancellation
SP	Service Provider

TDD	Time Division Duplex
UE	User Equipment
VN	Virtual Network
VNO	Virtual Network Operator
WDC	Wireless Data Center
WSN	Wireless Sensor Network

LISTE OF SYMBOLS AND UNITS OF MEASUREMENTS

A	Coverage area [km]
BW	Operating bandwidth [MHz]
C_{cap_i}	CAPEX for tier i [$k]
C_{op_i}	OPEX for tier i [$k]
C_{eq}	Equipment cost [$k]
C_{sb}	Site build-out cost [$k]
C_{sl}	Site lease cost [$k]
C_{om}	Operation & Maintenance cost [$k]
C_{bh}	Backhaul cost [$k]
c_p	Cash flow at year p [$k]
$C_n^{c/r}$	CVN/RVN cost [$k]
C_{rrh}	Total cost for RRH [$k]
C_{rrh_c}	CAPEX for RRH [$k]
C_{rrh_o}	OPEX for RRH [$k]
CP	Capitalization rate
c_e	cost of power delivery/$Watt-hr$/year [$k/$Watt-hr$/year]
d_{BS}	Coverage radius of a BS [km]
d	Discount rate
d_{cpc}^{opt}	Optimal CPC radius [km]

d_{cpc}	Coverage radius of a data center [km]
G	Cyclic prefix length
g	Self interference cancellation gain [dB]
GP_{opt}	Optimal guard period
$H_{n,u}^{DL}$	DL channel gain between BS n and UE u
$H_{n,u}^{UL}$	UL channel gain between BS n and UE u
K_p	Power burden factor
K_C	Cooling burden factor
L	Load factor
N	Noise power [dB]
N_{UE}	Average number of active users
n_{sl}	Number of slices in a SBS
N_r^m	Number of RRHs for macro-coverage
N_r^{mi}	Number of RRHs for micro-coverage
N_r^p	Number of RRHs for pico-coverage
NOI	Net operating income/m^2/year [\$km^2/year]
N_{sub}	Number of data sum carriers
N_{mod}	Number of modulated bits/symbol
N_{cod}	Coding rate
N_{cpc}	Number of CPC

n	Sampling rate
oc	Occupancy factor
$P_n^{DL_{max}}$	Maximum BS transmission power [dB]
$P_n^{UL_{max}}$	Maximum UE transmission power [dB]
P_{HW}	Hardware power consumption [Watt]
p_c	CVN portion of the HVN
R_{BS}	Data rate capacity/BS [Mbps]
R_{UE}	Data rate capacity demand/UE [Mbps]
R_{MBS}	Macro BS data rate [Mbps]
R_{MiBS}	Micro BS data rate [Mbps]
R_{PBS}	Pico BS data rate [Mbps]
R_{um}	Macro user data rate [Mbps]
R_{umi}	Micro user data rate [Mbps]
R_{up}	Pico user data rate [Mbps]
R_{TDD}	OFDM data rate [Mbps]
R_{LTE}	LTE data rate [Mbps]
$R^{D}_{n,u}L$	Downlink user rate for user u connected to BS n [Mbps]
$R^{U}_{n,u}L$	Uplink user rate for user u connected to BS n [Mbps]
$SINR^{DL}_{n,u}$	DL signal-to-interference-and-noise ratio for user u connected to BS n
$SINR^{UL}_{n,u}$	UL signal-to-interference-and-noise ratio for user u connected to BS n

T_{sf}	Special subframe length
t_{enb}	Switching time of the eNB [sec]
U_{opt}	Optimal utility
w_c	cost weight
δ	Ratio of the pilot bearing-symbol to total number of symbols
γ	LVN cost increment coefficient
μ_m	User density for Macro BS [$/km^2$]
μ_{mi}	User density for Micro BS [$/km^2$]
μ_p	User density for Pico BS [$/km^2$]
ν_m	HetNet coefficient

INTRODUCTION

The Information and communication technologies (ICT) industry has been going through revolutionary changes in the last decade. These changes are manifesting themselves in the form of various new services, smart and slick personal computing devices and even in the architectural changes in the mode of computing itself. In a more *connected and communicative* information society, wireless networks are playing the role of major connection medium due to their ubiquitous presence in the information eco-system (Raychaudhuri and Mandayam, 2012). For its increasing importance as a communication channel, the stress on wireless networks for carrying significantly more traffic with varied quality of service (QoS) requirements is stronger than ever before. But it is a well known fact that with scarce radio spectrum (Akyildiz *et al.*, 2006) and limited control over the networking gear, operators are struggling to cater for such needs. With vendor locked-in devices, it is not possible to provision new network services that require novel protocols or processing algorithms for optimal service provisioning. The way forward is to rethink the way traditional networks work and re-architect them, so that, they offer more flexibility (Koponen *et al.*, 2011) and fine-grained control over the network resources that will enable operators to provide novel differentiated services, while at the same time ensuring efficient resource utilization.

Traditional cellular networks are designed to serve the peak network traffic demand. This often results in over-provisioning of network resources (Zhou and Chen, 2014), which is very expensive in terms of network deployment as well as operational costs. Network operators do not have the facility of on demand resource provisioning which would allow them to scale-up or scale-down network resources according to traffic demand at any given instant of time. Moreover, the use of complex control plane protocols and vendor locked-in devices are not amenable to provision new cellular services that might require to implement novel protocols or signal processing schemes. Future 5G networks will demand a more flexible and elastic network architecture that will facilitate provisioning novel services in lower network cost, which is not possible with current network architectures. To resolve these issues, it is imperative to re-architect current network structures in new ways that make most efficient use of available resources, use less expensive general-purpose hardware rather than expensive special-purpose hardware to reduce overall network cost and provide flexibility to incorporate new network technologies using programmable and elastic network infrastructure (Pentikousis *et al.*, 2013). Virtualizing wireless access solves to a great extent the aforementioned problems.

Virtualization has been employed in computer systems for a very long time for abstracting memory (Morin and Puaut, 1997), storage (Dimakis *et al.*, 2011), or virtual systems (Smith and Nair, 2005). In wired networks virtualization has been implemented for deploying virtual local area networks (VLANs) in enterprise networks, and also for deploying virtual private networks (VPNs) in wide area networks (WANs) (Chowdhury and Boutaba, 2010). Several internet research testbeds (Chun *et al.*, 2003), (Niebert *et al.*, 2008) have been deployed to study the technologies to overcome the architectural challenges of the current internet.

To cope with the novel service requirements of the future 5G networks and the incredible growth in user traffic, virtualization of wireless networks is being advocated by major telecom operators and vendors (nfv, 2013). Wireless virtualization enables the decoupling of the physical network infrastructure from the services that it provides. In a virtual wireless network scenario, the infrastructure providers (InPs) deploy and manage the physical network infrastructures. Different virtual network operators (VNOs) lease virtual network nodes from the InPs and deploy their own network by dynamically sharing the physical infrastructure. Since physical infrastructure is shared by multiple VNOs in a virtualized platform, a significant improvement in the capital expenditure (CAPEX) and operational expenditure (OPEX) is achieved. It has been reported that a saving of 40% can be achieved (Perez *et al.*, 2009) in CAPEX and OPEX over a period of five years by radio access network virtualization. Provisioning of differentiated services on a common physical infrastructure also ensures efficient resource utilization. By slicing the wireless network through virtualization, it is convenient to deploy newer services while supporting legacy services.

In a virtual access topology, *independent and isolated* virtual networks are built on one or more physical network substrates in which the presence of the virtual networks are transparent to each other. The virtual networks are able to use customized network protocols, signal processing and network management functionalities that best suites the intended services. In virtualized wireless networks, physical wireless infrastructure as well as the wireless spectrum is abstracted and sliced, so that, multiple VNOs share the resources to build their own networks. But the presence of each VN should be isolated from each other, so that, the operation of one VN does not affect any other VN sharing the same physical infrastructure and radio spectrum. One major distinction of the wireless network from the wired network is the stochastic nature of the wireless channel that varies with time and space and also suffers significant attenuation with propagation distances. This makes abstraction and sharing of radio spectrum very challenging.

Though there exists no universal consensus on the definition of wireless network virtualization, its scope lies in the virtualization of physical network infrastructure as well as the wireless spectrum. In such a scenario, physical infrastructure and radio spectrum are deployed and owned by one or more InPs who employ resource abstraction & isolation mechanisms to create virtual resources that are eventually shared by multiple VNOs. In this thesis, wireless network virtualization is defined as a network technology that abstracts network resources (both physical nodes and radio spectrum) in a technology-agnostic manner. These abstracted network resources are then sliced to create virtual resources which are then shared by multiple VNOs where an isolation mechanism ensures transparent co-existence of the deployed virtual networks on a common physical substrate. As an integral part of the wireless virtualization, end-to-end programmability of the virtual networks has also been emphasized in this work which will enable VNOs to deploy their own customized (virtual) networks.

Requirements of Virtual Wireless Networks

A virtualized wireless network must satisfy certain requirements, and some of these critical requirements are discussed in this section.

Virtual Network (VN) Isolation

The VNs sharing a common physical infrastructure should be perfectly isolated from each other, so that, to a VNO, it will appear that it has the sole-ownership of the (virtual) network. Operation of a VN should in no way affect the other VNs sharing the same physical resource, e.g., for two VNs sharing a common physical node, if load increases in one VN, traffic belonging to the other VN should not suffer from additional delay in processing, queuing or reduction in throughput. Service level agreements (SLAs) between the VNOs and the InPs should be always fulfilled. SLAs are basically a subset of key performance indicators (KPIs) which might comprise of minimum guaranteed processing power, memory space, bandwidth/throughput, maximum downtime of system, etc.

End-to-end Programmability

VNOs should have complete flexibility over the virtual/physical resources they lease from one or more InPs. This flexibility is demonstrated through their ability to modify (program) the underlying resources in a way that best supports their intended service requirements. For example, in a virtual LTE network implementation, a VNO should be able to program the core network switching fabric to route its core network packets through the optimum routing graph consisting of mobility management entity (MME), switching gateway (S-GW), packet data network gateway (P-GW), policy and charging rules function (PCRF), etc. Similarly, for the radio access plane, a VNO might require customized radio processing chain to process its baseband signal. Hence, provisions should be made so that, it can assemble various processing blocks (e.g., for frequency transformation, modulation, coding, etc.) in a programmatic manner (Bansal *et al.*, 2012). Also, a VNO should be able to implement its custom protocol stack to optimize its intended service performance; hence, programmatic control over the protocol layers is also necessary.

On demand resource provisioning

In the VNO-InP business model, a VNO would request for its required resources (virtual/physical) to the InP. Upon availability of the resources, the InP would assign the requested resources to the VNO, forming a service level agreement (SLA) between the InP and the VNO. During its operation, if the VNO need additional resources (e.g., computing, storage, radio spectrum, etc.) it would request the InP for the lease of these additional resources. InPs should be able to cater for such on demand elastic resource provisioning. This is where the cloud computing model comes into play in a virtual wireless network ecosystem. In this model, an InP can be seen a as cloud service provider that composes of geographically distributed *cloud* of resources. A VNO receives the lease of its requested resources from the InP without the necessity of being aware of the physical location of the resources.

Network Function Virtualization (NFV)

One of the major motivation behind network virtualization is to reduce the CAPEX and OPEX of network provisioning, so that, the operators can cope with the increasing network cost and also, new players can get affordable entry to the market. To address this issue, major telecom operators and vendors are opting for network function virtualization (NFV) (nfv, 2013). The main idea behind NFV is to separate network hardware from the software that runs on it, this will pave the way to implement different network functionalities as software instances in a general IT platform. This paradigm shift in network architecture will replace the traditional specialized network nodes which are not only expensive but also very power hungry.

Dynamic Spectrum Sharing

Scarcity of the licensed spectrum is the *Achilles' heel* for the next generation wireless networks. Despite all the advances made in network architectures, baseband processing, error correction channel coding, etc., limited licensed spectrum remains the major bottle neck for telecommunication networks. To alleviate this problem, efficient utilization of the radio spectrum in time (time division multiple access (TDMA)), frequency (frequency division multiple access (FDMA)), space (space division multiple access (SDMA)) is necessary (Niebert *et al.*, 2008). Especially in the virtual wireless networks environment, dynamically sharing the spectrum among the incumbent VNs while respecting the SLA is of utmost importance. Opportunistic sharing of the licensed spectrum in combination with utilizing the unlicensed spectrum band wherever possible might mitigate the spectrum scarcity problem to a great extent. Also, the use of millimetre (mm) wave for future 5G network is gaining momentum as many researchers from industry and academia are strongly advocating in its favor (Zhao *et al.*, 2013), (Rappaport *et al.*, 2013). Due to its ability to provide higher throughput for lower transmission distance mm wave is an ideal transmission candidate for small cells (pi and Khan, 2011). In this thesis, software defined networking (SDN) and cloud computing technologies have been considered as key enabling technologies for implementing virtualized wireless networks. A brief introduction to these technologies is given below:

Software Defined Networking (SDN)

Traditional networks are designed to have distributed control for scalability reasons. In this structure, network intelligence is distributed throughout the network, where each network node has both control and data forwarding logic. For example, a simplified representation of an evolved packet system (EPS) is shown in Fig. 0.1. It consists of mobility management entity (MME), packet gateway (P-GW), switching gateway (S-GW), home subscriber server (HSS), policy charging and rules functions (PCRF) and the evolved node Bs (eNBs). The eNBs are the last mile radio access points. Each eNB has decision making and forwarding functionalities. It makes the local radio resource management decision for allocating radio resources to individual users. It also communicates with the neighbouring eNBs via X2 interfaces to cooperate resource provisioning. Functionalities like mobility management, policy implementation, charging, access control and even access to internet are managed by decision nodes resident in the core network. The problem with this kind of network architecture is manifold; first, the network architecture is very inflexible, it operates with a fixed set of network protocols and it is not possible to implement a novel network protocol that will have optimal performance for a new

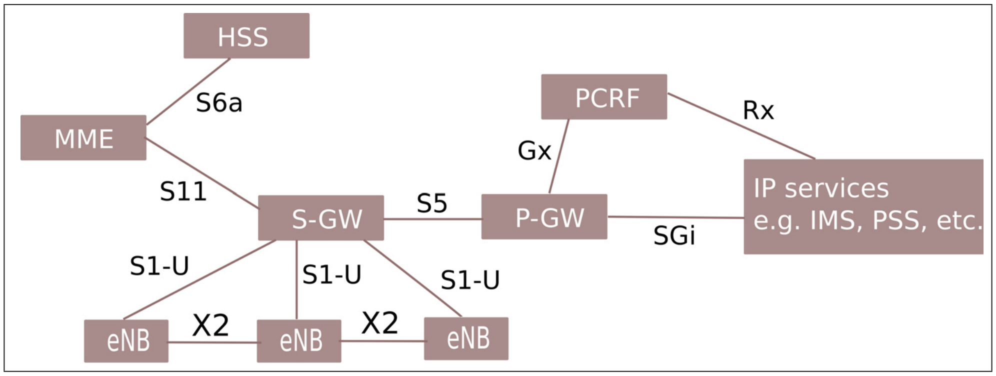


Figure 0.1 A block representation of EPS

service. Second, because of vendor locked-in network nodes, an operator has less freedom to purchase network equipment from different vendor companies because the proprietary equipment sold by vendors generally do not interoperate well enough with one another. Thirdly, the formidable cost of the network equipment discourages network operators to provision new services as it will require to add new equipment to the network, sometimes replacing previously purchased well-functioning equipment. The high cost of network roll-out also acts as an entry barrier for new entrants to the heterogeneous wireless networks' business eco-system. Finally, the special purpose hardware based networks also have higher operational costs due to higher power consumption and requirements for a significant number of highly skilled employees for operation and management of the network. Software defined networking can resolve these issues to a great extent by flexible creation and management of networks using inexpensive programmable switches and off the shelf general purpose servers.

Software defined networking (Nunes *et al.*, 2014), (Xia *et al.*, 2015) is a relatively new paradigm in network architecture design that has created a lot of interest in both industry and academia alike. SDN is a complete makeover of the norm with which network intelligence and forwarding cooperate with each other. SDN enables *programming the underlying network as a system* by separating the control plane from the data plane. It provides high level abstraction of the network hardware, where a centralized controller can program the network. As defined in the Open Networking Foundation (ONF) white paper (onf, b), in a SDN architecture, network control and data planes are decoupled and the intelligence and state of the network are logically centralized in the controller platform. SDN facilitates traffic engineering (Akyildiz *et al.*, 2014), and it has also been used in large scale wide area network (WAN) (Jain *et al.*, 2013). SDN can also function as an enabler for network function virtualization (NFV) (Chiosi *et al.*, 2012a), (nfv, 2013) which is a major sought after technology by telecom operators around the globe. Though the immense interest on SDN is pretty recent, the core idea of programmable networks is the accumulation of research advances on different aspects in this area (kreutz *et al.*, 2015), (Feamster *et al.*, 2013). To summarize, the main components of SDN architecture are:

- separation of network data plane from the control plane;
- logically centralized control and global view of the underlying network infrastructure;
- programmability and modularity of the control plane;
- high-level abstraction of the hardware layer;
- open application programming interfaces (APIs) for data plane (McKeown *et al.*, 2008) and control pane (Gude *et al.*, 2008), (Mccauley), (Ryu), (ope, b), (flo), so that, both planes can grow independently of each other.

A simplified schematic for a SDN architecture is shown in Fig.0.2. The top tier is the application layer where the network applications reside that define the operational behavior of the network. Different applications, for example, routing, mobility management, access control via firewall, load balancer can be part of this layer. VNOs can have one or more applications *packed* together for a particular service provisioning. For easier management of different applications in the application layer, the northbound API (Reich *et al.*, 2013), (Foster *et al.*, 2011) is used to ensure the synchronous operations of different applications. The controller layer consists of network operating systems (NOSs) e.g., NOX (Gude *et al.*, 2008), POX (Mccauley), OpenDaylight (OD) (ope, b), Floodlight (FL) that interfaces the application layer with the forwarding layer. This layer is responsible for dynamically setting up (and tearing down) network paths according to the application layer instructions by modifying the underlying programmable switching fabric. For this purpose, the controllers use well defined southbound API (e.g., OpenFlow (McKeown *et al.*, 2008), NetConf (net, 2013), ForCES (Yang *et al.*, 2004), etc.) to program the underlying switching fabric. SDN is gaining increased interest

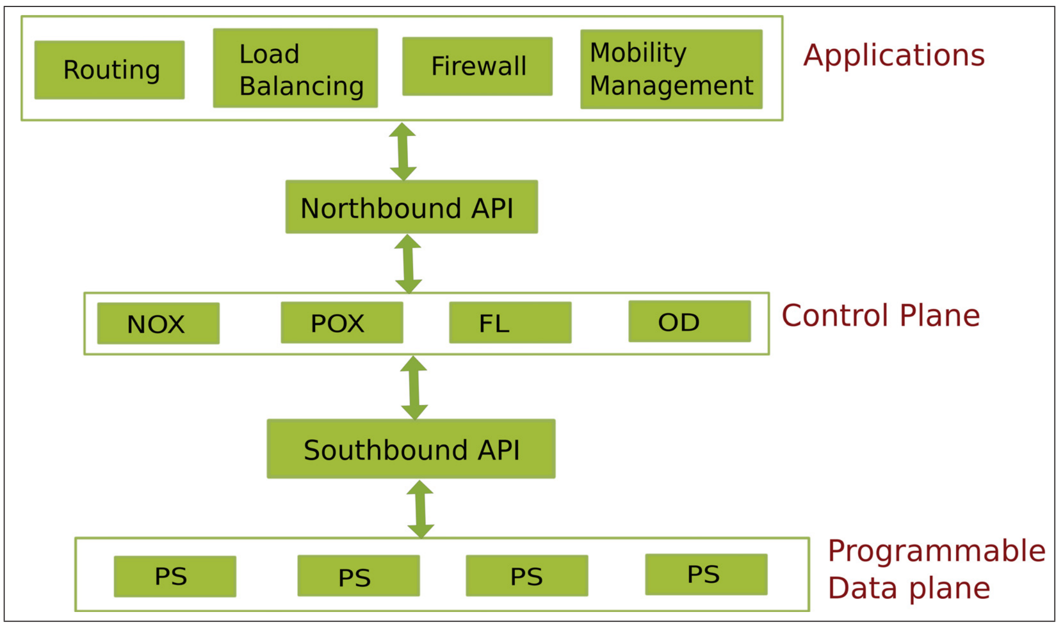


Figure 0.2 Simplified representation of a SDN architecture

from both wireless industry and academia working on wireless network research to facilitate service differentiation, and ease network management, network innovation and convergence of heterogeneous wireless networks. We classify the proposals on wireless networks leveraging SDN according to their target wireless domain, i.e., WiFi, sensor and cellular networks. We compare the proposals from their capability of providing an end-to-end programmable virtualized solution for the target wireless network domain.

Cloud computing

Cloud computing is a relatively new paradigm for large scale distributed computing. The major benefit of cloud-based infrastructure is its ability to provide on-demand computing resources in convenient pricing schemes, e.g., pay-as-you-go, paying for the leased resources that can be elastically scaled up or down depending on cloud clients' demand at a specific point of time. Cloud resources are basically composed of storage, computing and networking elements.

There are mainly three types of abstraction for cloud-based service provisioning, namely, infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS) and software-as-a-service (SaaS). In a IaaS model, processing, storage, networking and other computing resources are provided as a standardized services by cloud providers to their clients. The IaaS clients can deploy and run their own operating systems (OSes) on the leased resources from the cloud provider. For example, Amazon web services (AWS) (aws), Microsoft Azure (msA), Google Compute Engine (GCE) (gce) are some popular IaaS platforms. In case of a PaaS model, a higher level of abstraction of network resources is used and the cloud clients are provided with run time systems using which they can build (program) their own customized applications and run on the PaaS platform. Apprenda (app) is a PaaS platform for developing .Net and Java based applications. On the other hand, the SaaS model is the highest level of abstraction provided by the cloud providers where different applications are provided as services to the clients. Example of SaaS are: email services, various customer service management application, e-health software services, etc. Fig. 0.3 shows a schematic representation of different cloud models. When cloud computing technology is extended to the virtual wireless network domain, besides the traditional cloud resources (e.g., compute, storage, network, etc.), a cloud provider should also provide access to various wireless access nodes (e.g., APs, BSs, Repeaters, sensor nodes, etc.), core network elements (e.g., EPC for LTE core network) as well as access to wireless radio spectrum. VNOs will build their customized networks with the above mentioned leased resources.

Objectives

Virtualization of wireless networks is a fairly recent trend in wireless research. Different research groups in industry and academia are working to conceive the architectural model for virtualizing wireless networks. But there exists no agreed upon architectural framework for provisioning virtual wireless networks. In this thesis, we have proposed three wireless net-

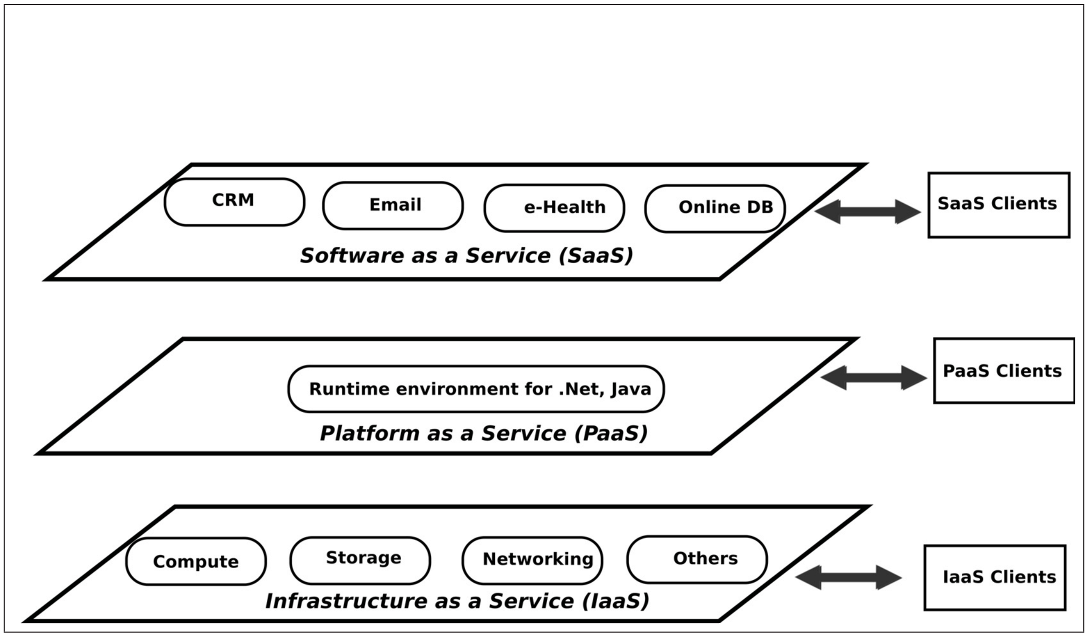


Figure 0.3   Different models of cloud infrastructure

work virtualization frameworks that provide an end-to-end virtualized solution for wireless networks. As has been mentioned previously (and in more detail in Chapter 1), current research on wireless virtualization strictly focuses on the technical aspect of virtualization. But there is also a significant economic aspect to any network architecture and there are cases where the most technically sound solution may not be implemented due to the network cost limitations for the prospective network operators. Hence it is very important to evaluate network deployment options not only from their technical merits (or demerits) but also from their economic merits. In this thesis, the proposed virtualization frameworks are analyzed from their cost perspectives as well as from their achievable QoS. The goal is to come up with a composite techno-economical model that considers a virtualized network's cost (both CAPEX and OPEX) and achievable QoS and provides deployment solution for a particular scenario while considering the expenditure constraint of a network operator/ service provider.

Next in this thesis we focus on the architectural deployment of heterogeneous wireless network virtualization. And investigate the provisioning of differentiated services as SDN policies.

As part of the thesis, we were involved in a research collaboration with Huawei Canada Research Center. The objective of the collaboration was to investigate the deployment challenges for full duplex (FD) cellular networks and develop algorithm for successful realization of the FD multi-cell multi-tier networks.

Contributions

Wireless network virtualization is a multi-dimensional problem that involves slicing of physical nodes, spectrum sharing, air interface virtualization, etc. Unlike the wired networks, the transmission channels of wireless networks are inherent broadcast in nature and also vary with time. Also mobility of users and attenuation of radio signals make wireless virtualization even more complicated. Wireless network virtualization has been approached from different perspectives: spectrum virtualization (Perez *et al.*, 2009),(Zaki *et al.*, 2010a), as well as virtualization for different wireless technologies (i.e., WLAN, WiMAX, LTE) (Singhal *et al.*, 2008),(Bhanage *et al.*, 2010a),(Bhanage *et al.*, 2010c),(R.Kokku *et al.*, 2012b),(Yap *et al.*, 2010b),(Zhu *et al.*, 2010). There have also proposals for virtualizing the access networks (Perez *et al.*, 2009) as well as the core networks (Kempf *et al.*, 2012). But there has not been any solution that provides an end to end solution to wireless network virtualization. Moreover, different kinds of virtualization architectures vary significantly from the hardware infrastructure they use and also the virtualization mechanism used. For this reason, the implementation cost vary significantly from one architecture to the other. Also the available QoS varies significantly in these different architectures. The contribution of thesis can be classified into three parts:

1) in the first part, three architectural frameworks for wireless network virtualization are proposed. The frameworks differ significantly in their architecture, especially in terms of the degree of segregation between the baseband processing and radio access units. The associated network cost also vary significantly from one framework to the other. Moreover, the achievable QoS in the proposed virtualization frameworks also varies considerably. Hence, a novel utility model has been developed to select the best network for a certain implementation scenario. This part provides:

A) the classification of virtual wireless access networks into three models (considering *Green Field* deployment scenarios):

- a special-purpose hardware-based wireless access virtualization model, referred to as Locally Virtualized Network (LVN), where a hypervisor is used to slice super base stations (SBSs) to create multiple virtual base stations (VBSs);
- a data center based wireless access virtualization model, referred to as Clustered/Remote Virtualized Network (CVN/RVN), where SDN and cloud computing technologies are used to virtualize the underlying networking fabric and computation & storage resources. In this model, fiber-distributed remote radio heads (RRHs) are used to provide radio access to users;
- a third model, referred to as hybrid virtualized network (HVN), where we properly combine both of the above mentioned models to offer the potential to balance network cost and QoS with greater flexibility than the previous two models (LVN and CVN/RVN).

B) a new multi-criteria utility function that account for network cost & QoS trade-offs to enable the design and optimization of wireless access virtualization architectures that best comply with the investment and service-level requirements of network operators (and/or service providers).

The techno-economic model suggests that the HVN framework that is composed of special-purpose hardware as well as cloud-based data center achieves the optimal balance between network cost and QoS (Rahman *et al.*, 2014b) (Rahman *et al.*, 2013), (Rahman *et al.*, 2014a), (Rahman *et al.*, 2015d).

2) in the second part of the thesis, it is argued that for realizing a programmable & flexible heterogeneous virtual network infrastructure, SDN & cloud computing technologies are the key tools to leverage. In such a network infrastructure, VNOs will be able to offer their differentiated services in their target networks (e.g., WSN, cellular or WiFi) leasing virtual resources from one or more InPs. Different proposals on wireless networks exist in the open literature (c.f. Chapter 1) that uses the SDN and cloud computing concepts

for efficient management of different wireless networks. But these proposals focus on a particular network technology (WiFi or 4G or WSN). But no approach exists that gives solution for implementing a heterogeneous virtualized wireless network platform that will enable the deployment of different virtualized wireless networks using different radio access technologies (RATs) on a common physical infrastructure. In this regards, we lay out the blueprint of an end-to-end programmable heterogeneous virtualized wireless network (HVWN) platform. The key requirements of such a heterogeneous virtual wireless network (HVWN) infrastructure have been identified. Then different components for an end-to-end solution for a programmable, elastic HVWN have been discussed following a top-down approach. Open problems and challenges in realizing a programmable, elastic HVWN have also been identified (Rahman *et al.*, 2015a).

3) in the third part, implementation of differentiated services in a virtualized platform are studied. We have focused on a particular case of the end-to-end programmable HVWN proposed in Chapter 3 where a cellular and an WiFi network are implemented on common physical infrastructure. We study how differentiated services can be implemented in such a programmable virtualized platform through extensive system level simulation in Mininet (min) platform. We propose to use the spare bits of OpenFlow (McKeown *et al.*, 2008) packet structure to implement virtual network entities, e.g., virtual networks, virtual switches, allocated radio resources of a virtual operator, etc. We also emphasize the use of northbound APIs to facilitate composing complex wireless network applications. To demonstrate the impact of using northbound APIs, we have implemented different network applications using Pyretic (Reich *et al.*, 2013) and composed them in sequence and/or in parallel to provision differentiated services in two virtual network scenario. It has been shown through system-level simulation that it is possible to implement differential services in a virtualized network platform by expressing service differentiation as high-level network policies through SDN paradigm. The intensive system-level simulation results suggests that such SDN-based virtualized platform is capable of achieving the service requirements (e.g., throughput and transmission delay) of traditional cellular networks (Rahman *et al.*, 2015b), (Rahman *et al.*, 2015c).

4) in the final part of the thesis, we focused on FD deployment of multi-cell networks. Current cellular networks are suffering from spectrum ossification problem. In a virtualized environment where multiple VNOs will compete for access to shared radio resources, the spectrum scarcity problem will be more severe. In such context, FD systems can provide an efficient solution by doubling the spectral efficiency hence, doubling the capacity. In our research, we have identified the critical challenges for real world deployment of multi-tier FD multi-cell networks. We have analyzed and report on FD performance trade-offs for a dense urban multi-tier cellular network. We have used the Madrid grid model proposed by METIS project (Agyapong and et al., 2013) that consists of macro and pico cells. We have also investigated the impact of co-located BS interference in FD performance for a single-tier homogeneous network deployment. We have proposed intelligent proportional fair joint user selection and power control algorithms to harness the gain of FD deployment. We have developed algorithms for both cloud radio access network (C-RAN) and traditional distributed RAN (D-RAN) network models. Extensive system-level simulation results show that using the devised algorithms the FD systems are able to achieve significant performance gain (Rahman *et al.*, 2016b), (Rahman *et al.*, 2016a).

The list of publications resulted from this thesis is given in Table 0.1.

Table 0.1 **List of publications from this thesis**

Article type	No.	Title	Publisher
Journal	J1	Design Optimization of Wireless Access Virtualization Based on Cost & QoS Trade-off Utility Maximization.	accepted for publication at IEEE TWC 2016.
	J2	End-to-End Programmable, Cloud-based Virtualized HetNet: Advances Made & Challenges to Address	Elsevier Computer Communications (under review)
	J3	Deployment of Full Duplex Multi-Cell Systems for Dense Urban and Rural Environments	IEEE TWC (under review).
Conference	C1	Analysis of CAPEX and OPEX Benefits of Wireless Access Virtualization	IEEE ICC'2013
	C2	Green Wireless Access Virtualization Implementation: Cost vs. QoS Trade-Offs	ICECCS'2014
	C3	Configuration Cost vs. QoS Trade-off Analysis and Optimization of SDR Access Virtualization Schemes	IEEE NetSoft'2015
	C4	Service Differentiation in Software Defined Virtual Heterogeneous Wireless Networks	IEEE ICUWB'2015
	C5	HetNet Cloud: Leveraging SDN & Cloud Computing for Wireless Access Virtualization	IEEE ICUWB'2015
	C6	Multi-Cell Full-Duplex Wireless Communication for Dense Urban Deployment	IEEE Globecom'2016

CHAPTER 1

STATE OF THE ART

1.1 Wireless Network Virtualization

Virtualizing the wireless network infrastructure enables sharing of the physical resources by multiple operators at the same time. This will ensure efficient resource utilization which is critical for the success of any successful business operation. Virtualization is the process of abstracting physical resources, so that, multiple network entities (VNOs), can have shared access to these resources to deploy their own customized network. Virtualization for the wired networks is a well studied and well understood topic (Chowdhury and Boutaba, 2010). In comparison, virtualization of wireless networks, is a fairly recent trend (Smith *et al.*, 2007), (Hamaguchi *et al.*, 2010), (Xia *et al.*, 2011), (Perez *et al.*, 2009). Virtualization of wireless networks has immense benefits. Besides providing a platform for shared access of network hardware resources, it can also enable shared access of wireless spectrum (Zaki *et al.*, 2010b), which can mitigate the long standing spectrum ossification problem (Tan *et al.*, 2012b). By shared usage of wireless resources (both equipment and radio spectrum), wireless virtualization can reduce a network's CAPEX and OPEX (Rahman *et al.*, 2013), (Rahman *et al.*, 2014a).

Virtualization has been proposed for different wireless radio access technologies. Network Virtualization Substrate (NVS) (R.Kokku *et al.*, 2012a) is a WiMAX virtualization platform for creating virtual wireless networks on a common physical substrate. It is basically a MAC layer virtualization technique that allows bandwidth-based and resource-based slicing through a slice scheduler. Moreover it also incorporates customized flow scheduling for each slice in a BS. A virtual base station architecture for a WiMAX network is presented in (Bhanage *et al.*, 2010b). In this model, virtual base stations are implemented in an external substrate that uses layer-2 switched data paths and a control path to the BS. Radio resources of a BS are virtualized to create isolated slices that can implement different flow types with customized flow scheduling algorithms. SplitAP (Bhanage *et al.*, 2010c) is a WLAN virtualization architecture, focused on fair sharing of uplink airtime across a group of users. A physical AP can be shared by different slices that can run different algorithms to control the UL airtime among different user groups. In (Zaki *et al.*, 2010b), virtualization of the air interface of the LTE network has been studied. Here, a hypervisor was used for virtualizing the wireless spectrum.

Different experimental test-beds (using SDN or not) have been developed to do research on clean-slate networking technologies leveraging virtualization. GENI (Bermana *et al.*, 2014), Planetlab (Chun *et al.*, 2003), AKARI (aka, 2009), SAVI (Kang *et al.*, 2013), OFELIA (ofe), 4ward (Niebert *et al.*, 2008) to name a few. Wireless network virtualization (WNV) can be achieved in different ways. But we deem SDN and cloud computing as significant enablers for successful realization of WNV. These technologies have gained increased attention in recent time. SDN introduces flexibility in network deployment and management while cloud computing enables on-demand, elastic resource provisioning. Recent works on wireless networks that employs SDN and cloud computing are briefly discussed in this chapter.

SDN (Bosshart *et al.*, 2013), (Kobayashi *et al.*, 2013) is able to abstract physical resources for its ability to separate the network control plane from the data plane. Thus it can provide absolute control over the network substrate through programming. The flexibility provided by SDN is instrumental in providing novel services that require changing device functionalities to provide differential services (Rahman *et al.*, 2015b). In addition to providing flexibility in managing the network infrastructure, SDN can also reduce network cost (both CAPEX and OPEX) by replacing expensive network nodes with off-the-shelf (OTS) cheaper programmable data plane equipment and centralizing controller in IT servers.

To further reduce the network cost, expensive special purpose network nodes that perform specific tasks, e.g., mobility management, gateway functionalities, billing, etc., can be implemented as software instances that run on IT servers. This can be achieved by separating network device hardware from the software that runs on it. This separation of network devices and software is known as network function virtualization (NFV) that is being actively sought after by major telecom operators and vendors around the globe (nfv, 2013). In the NFV model, the virtual network functions (VNFs) are put in centralized locations, i.e., data centers where the physical resources (e.g., processing, storage, networking, etc.) are pooled together. In a distributed pooled resources model, these data centers are basically distributed clouds of resources managed by infrastructure owners (InPs), who can provide on-demand, elastic resources to the operators (VNOs). Network nodes (e.g., base stations (BSs) and access points (APs)) can also be implemented as software instances in these data centers. To handle the high processing requirements of baseband signals, besides the software instances, special purpose FPGA-based processing boards can also be installed in the data centers.

1.1.1 SDN for WiFi Networks

SDN has been used for WiFi networks for implementing applications based service provisioning. Also, SDN is leveraged as a tool for implementing virtualization of WiFi access points (APs). Odin (Suresh *et al.*, 2012) is a software-defined wireless network prototype for enterprise WLANs. It implements a flow-based virtualization technique to enable network operators to implement different WLAN services as network applications. In this architecture, an Odin master is the central controller entity that uses OpenFlow (McKeown *et al.*, 2008) to program switches and APs that it controls. Each AP is packed with an Odin agent that communicates with Odin master by using Odin's custom protocol. The applications on top of the Odin master uses Odin's primitives to implement different enterprise services. Odin is a single operator solution to implement virtual AP abstraction and does not consider the case when multiple VNOs operate on a common infrastructure. Also, it does not consider abstraction and sharing of radio resources.

EmPOWER (Riggio *et al.*, 2013) is an experimental testbed for SDN and NFV experimentation. The testbed's data plane consists of OpenVSwitch (ope, c) and Click Modular Router (cli), while Floodlight (flo) has been used as the controller platform. It also utilizes a power management component called Arduino. This AP-based test-bed has provisions for implementing different network applications as slices. But the feasibility of implementing a resource allocation based multi-VNO platform is not discussed in the paper. Also virtualization of radio resources has also not been discussed.

1.1.2 SDN for Wireless Sensor Networks (WSNs)

To resolve the management problem of WSNs, SDN has been used for smart management of sensor networks. For example, a software-defined wireless sensor network (SD-WSN) (Luo *et al.*, 2012) proposes a flexible, generalized architecture for WSN. To overcome the resource underutilization and network management problems of traditional application specific WSNs, the authors propose a programmable sensor network by following the control and data plane separation paradigm of SDN. To handle the *data-centric* characteristics of WSN, as opposed to the *address-centric* model of OpenFlow (McKeown *et al.*, 2008), a modification of the OpenFlow protocol, named Sensor OpenFlow (SOF) has been proposed in this paper. SOF uses two different addressing schemes, *Class-1: compact network-unique addresses* and *Class-2: concatenated attribute-value pairs (CAV)* that suits the data-centric operation mode of WSNs. Managing the control channel overhead and additional latency (due to data exchange between the control and the data planes) to ensure the desired performance SD-WSN would be a challenging task.

Gante and et al. propose a WSN framework to facilitate management of a WSN (Gante *et al.*, 2014). The authors propose a distributed control mechanism by incorporating a software-defined controller in each sensor BS. Application layer above the controller dictates the flow-table format of the sensor nodes. As dictated by the application (e.g., temperature, humidity sensing), the controller collects information from the sensor nodes and defines flow tables. For calculating the optimal routes among the sensor nodes, the controllers form an adjacency matrix that consists of the connection information (e.g., distance, signal strength, energy level, etc.) between the adjacent nodes. In this model, each node forms its own neighbor table which is sent to the BS to enable the controller to build a network interconnection map. The energy-aware routing mechanism proposed in this paper is very efficient for low-powered sensor network environment but as sensor networks are power limited, periodical update of the neighbor tables from sensor nodes might create significant overhead burden.

1.1.3 SDN for Cellular Networks

A software-defined network paradigm has been proposed for cellular networks for both core and access network parts. These proposals leverage network programmability to foster rapid innovation, easier network management and also lower network CAPEX and OPEX. Some of such notable proposals are discussed in this section. SoftRAN (Gudipati *et al.*, 2013) proposes a software-defined centralized control plane for radio access network. It abstracts all the base stations (BSs) in a geographical area as one virtual big base station, composing of a programmable central controller and individual base station function as radio elements. All cross radio element resource planning is made by the controller, i.e., if decisions of one BS impact the decisions of another neighbouring BS, those decision should be made by the controller. As the controller has a network-wide view, this scheme will help in reducing interference, smooth the handover process and also can facilitate data offloading. On the other hand, decisions that are based on frequently varying radio parameters should be taken locally by the individual radio elements. SoftRAN basically targets to ease the management of a RAN by providing better control on network management issues like: load balancing and interference management.

SDMN (Pentikousis *et al.*, 2013) is a SDN based implementation of cellular core networks. It introduces a new MobileFlow stratum that decouples network control from the user plane. A MobileFlow controller controls the underlying MobileFlow forwarding engines (MFFEs) which are interconnected by IP/Ethernet network. MFFEs incorporate a standard mobile network tunnelling process, such as GTP-U, GRE encapsulation/decapsulation etc., that facilitate MobileFlow controllers to interoperate with legacy evolved packet core (EPC) nodes (e.g. MME, PGW, SGW, etc.). SDMN enables the creation of multiple virtual core networks over the same hardware resources. This work basically focuses on the core network part of cellular networks, while virtualization of radio access network has not been addressed. Also slice management issue on a shared infrastructure has also not been discussed.

CellSDN (Li *et al.*, 2012a) has been proposed as a way of simplifying design and management of cellular network using SDN. This architecture suggests allowing network control applications to express control policies based on subscriber attributes rather than the traditional trend of using network addresses and locations. Local agents are used in switches to enable fine-grained control of real time applications. It also suggests enhancing switch function capabilities to make them capable of more deep packet inspection (DPI). It proposes a slicing mechanism called CellVisor that is an extended version of FlowVisor (Sherwood *et al.*, 2009) capable of slicing cellular network resources. This work focuses on cellular network virtualization from user attribute point of view. Issues like radio spectrum virtualization (sharing among different VNOs), SLA enforcement have not been discussed in this position paper.

A scalable architecture for cellular core network is presented in SoftCell (Jin *et al.*, 2013), which offers control of high level network policies for mobile users. The central controller in this architecture implements network policies by directing traffic through a sequence of commodity middle boxes and forwarding devices. The structure of BSs and middle boxes remains unchanged but each BS is paired with an additional switch that performs packet classification of the traffic from the user equipments (UEs). In this way, part of the traffic management is offloaded toward the edge of the network and the network controller installs high level service policies to the underlying network nodes. While SoftCell architecture brings flexibility in maintaining cellular core networks, multi-tenant virtual network implementation was not studied in this work. Moreover, the work focuses on the core network part of cellular infrastructure, and the efficacy of such fine grained control over radio resources was not investigated.

SoftAir (Akyildiz *et al.*, 2015) is a software-defined network architecture for 5G wireless networks. In this architecture, core network functionalities are implemented in data centers that consist of controllers in servers and programmable switches. RAN functionality is distributed between the data center and the RRHs. Modulation and demodulation functionalities are performed at the RRH while more demanding PHY and MAC layer functionalities are pooled in the data center. To realize such a network model, it is imperative to have high capacity fronthaul between the data center and the RRHs. In a certain geographical region, where it is not possible to have fiber optic cable or high capacity microwave links between the RRH and the data center, realizing such a network model will be very difficult.

A software-defined control plane architecture for 5G networks is presented in (Yazici *et al.*, 2014). In this architecture, a hierarchical network controller model is presented that enables service differentiation by allowing varied level of performance for different core network functionalities. A connectivity management as a service (CMaaS) paradigm is also presented which is a unified approach in managing user connectivity and simplifies user mobility, handoff and traffic routing. This work's proponents argue for an all-SDN programmable future network. While it acknowledges that SDN can be instrumental for implementing NFV, it does not discuss a multi-slice solution for virtualizing wireless networks. SoftMoW (Moradi *et al.*, 2014) presents a programmable, recursive and reconfigurable cellular WAN architecture. The hierarchical construction of the architecture enables seamless inter-connection among core networks, programmable control plane and global optimization. It presents a novel label swapping mechanism for end-to-end path setup that enables each controller to operate only on its logical topology. The scalable optimization achieved in the SoftMoW architecture facilitates different network-wide optimization, for example optimal routing, handover minimization in a certain area. SoftMoW aims at resource management in cellular WAN, specifically at the network core. But efficient resource utilization through shared resource usage in a virtualized platform is not discussed in the paper.

OpenRAN (Yang *et al.*, 2013) is a software-defined virtualized RAN architecture for heterogeneous networks. It consists of three parts: a wireless spectrum resource pool (WSRP) which is responsible for virtualizing radio spectrum, a cloud computing resource pool (CCRP) that consists of physical processing pool and an SDN controller which controls the underling network by abstracting control functions of the access nodes. This model proposes virtualization in four levels: application, cloud, spectrum and cooperation levels, respectively. Though this model outlines a general model of the software-defined HetNet, it does not give any detail on the implementation technologies that might be used to realize such an architecture. Also the authors do not discuss issues like slice management, virtualization technology used, i.e., flow-level virtualization or hard-slicing (i.e., physical segregation of resources), etc.

1.1.4 SDN for heterogeneous Networks

OpenRoads (Yap *et al.*, 2010a) is a seminal work on using SDN paradigm for wireless networks. This platform uses SDN to build a programmable virtualized wireless data plane. OpenRoads consists of basically three layers: a *flow layer* where the flow-tables of different data plane nodes are modified using OpenFlow (McKeown *et al.*, 2008) protocol. Different wireless configuration parameters, like: SSID, wireless channel assignments, transmission power level are controlled and monitored by SNMP protocol. To enable resource sharing among multiple clients, a *slicing layer* is used to slice the network using the FlowVisor (Sherwood *et al.*, 2009). The *controller layer* which is built on NOX (Gude *et al.*, 2008), has a global view of the whole network and it allows the network applications (by different network users) to add/modify flow-table entries in the underlying data plane. OpenRoads is a heterogeneous platform that supports both WiFi and WiMAX networks. It has been shown that the platform supports seamless vertical handover between the disparate wireless technologies (Yap *et al.*, 2010c). But the work does not discuss virtualization of radio resources (e.g., antenna, wireless spectrum, etc.). Also the effect of elastic capacity provisioning in flow-based virtualization such as this, has not been studied in this work, which is a critical issue for an end-to-end virtual wireless network provisioning. The proposals on software-defined wireless networks are summarized in Table 1.1.

Table 1.1 Proposals for Software Defined Wireless Networks

Network Type	Proposals	Vir. Method	Layers Affected	Slice Management	Wireless Parameters under study	Summary
WiFi	Odin (Suresh *et al.*, 2012)	Flow-based	Application, Control Data planes	NA	Throughput, Handoff	Implements WLAN services as network applications.
WiFi	EmPOWER (Riggio *et al.*, 2013)	Flow-based	WiFi APs	NA	NA	An experimental testbed consisting of APs for SDN & NFV research.
WSN	SD-WSN (Luo *et al.*, 2012)	NA	Sensor nodes	NA	NA	Uses SOF protocol to communicate between the control and data planes.
WSN	(Gante *et al.*, 2014)	NA	Sensor node and BS	NA	NA	Controller installed at each BS to facilitate network management.
Cellular Access	SoftRAN (Gudipati *et al.*, 2013)	Big virtual BS abstraction	RAN functionalities	NA	NA	Addresses RAN management issues by abstracting RAN in a certain geographical area.
Cellular Core	SDMN (Pentikousis *et al.*, 2013)	Flow-based	EPC	NA	Basic 3GPP attachment and bearer establishment	Uses novel MobileFlow stratum.
Cellular Core + RAN	CellSDN (Li *et al.*, 2012a)	Flow-based	Core and RAN nodes	CellVisor	NA	Enforces control policies based on subscriber attributes.
Cellular Core	SoftCell (Jin *et al.*, 2013)	NA	Core network	NA	NA	Controller routes traffic based on UE policies.
Core	(Yazici *et al.*, 2014)	NA	Core network	NA	RTT delay, throughput	Proposes an all-SDN programmable core network.
Core	(Moradi *et al.*, 2014)	NA	Core network	NA	Handover	Hierarchical control plane leveraging level swapping mechanism.
HetNet	OpenRoads (Yap *et al.*, 2009)	Flow-based	RAN	FlowVisor	Latency, throughput	Creates overlay virtual networks on top of WiFi/WiMAX substrate.

1.1.5 Virtualization without SDN

There have been works on wireless network virtualization that necessarily do not use the SDN concept of separating network control from the data plane. Network Virtualization Substrate (NVS) (R.Kokku *et al.*, 2012a) is a WiMAX virtualization platform for creating virtual wireless networks on a common physical substrate. It is basically a MAC layer virtualization technique that allows bandwidth-based and resource-based slicing through a slice scheduler. Moreover it also incorporates customized flow scheduling for each slice in a BS.

A virtual base station architecture for WiMAX network is presented in (Bhanage *et al.*, 2010b). In this model, virtual base stations are implemented in an external substrate that uses layer-2 switched data path and a control path to the BS. Radio resources of a BS is virtualized to create isolated slices that can implement different flow types with customized flow scheduling algorithms. SplitAP (Bhanage *et al.*, 2010c) is a WLAN virtualization architecture, focused on fair sharing of uplink airtime across a group of users. A physical AP can be shared by different slices that can run different algorithms to control the UL airtime among different user groups. In (Zaki *et al.*, 2010b), the virtualization of the air interface of the LTE network has been studied. Here, a hypervisor was used for virtualizing the wireless spectrum.

Different experimental test-beds (using SDN or not) have been developed to do research on clean-slate networking technologies leveraging virtualization. GENI (Bermana *et al.*, 2014), Planetlab (Chun *et al.*, 2003), AKARI (aka, 2009), SAVI (Kang *et al.*, 2013), OFELIA (ofe), 4ward (Niebert *et al.*, 2008) to name a few.

1.2 Programmable Radio Plane

The radio plane consists of radio front-ends and radio spectrum. This last-mile access network part constitutes a very important part of the end-to-end virtual wireless network framework. For a true virtual network implementation virtualization of the radio plane is of utmost importance. This section discusses the proposals on virtualization of radio transmission chain and wireless spectrum. We classify the state-of-the art into two categories: programmable front-end and programmable nodes & spectrum sharing.

1.2.1 Programmable Front-end

Radio front-end consists of the radio transmission chain of the transceiver systems. Programmability of the front-end gives greater control over the PHY layer processing and it also paves the way for implementing novel PHY layer processing schemes. Sora (Tan *et al.*, 2011) is a programmable radio platform where the PHY and MAC layer functionalities are implemented in general-purpose processor (GPP) platform. Sora hardware platform consists of a radio front-end for wireless transmission and reception, a radio control board (RCB) for interfacing radio front-end with the processing engine in the server, and GPP servers. In (Tan *et al.*, 2011) the authors also demonstrates SoftWiFi, a software-defined wireless system that can seamlessly interoperate with IEEE 802.11 a/b/g network interface cards. Success of such implementation is very interesting for cloud-based virtualization of wireless access networks using general purpose IT-grade servers, as it shows the feasibility of such network architecture.

OpenRadio (Bansal *et al.*, 2012) is a design for programmable wireless data plane. It provides a modular and declarative programming interface for PHY later processing of the wireless protocol stack. The architecture is divided into processing and decision planes, where the processing plane includes directed graphs of different algorithmic actions (e.g., different modulation, coding schemes) and the decision plane contains the logic as to which processing plane graphs should be used for a particular wireless stack implementation. Various wireless protocols, like WiFi, LTE can be implemented using the off-the-shelf DSP chips using this model. The hardware processing abstraction enabled by OpenRadio (Bansal *et al.*, 2012) can be leveraged for virtualization of radio front-end.

MPAP (He *et al.*, 2010) is a SDR architecture based on Sora (Tan *et al.*, 2011) platform that virtualizes the radio front end to support different radio standards on the same transmission hardware. It uses a SDR service layer which is basically a virtualization layer. To minimize interference among virtual nodes sharing a common physical node, a scheduler is used. Spectrum is shared among different virtual nodes in an opportunistic manner. Use of GPP hardware to run software implementation of different radio functionality adds significant CAPEX and OPEX gain to such architecture.

1.2.2 Programmable nodes and spectrum sharing

Shared access of radio nodes as well as wireless spectrum is critically important for virtualization of wireless networks. Virtual radio (Sachs and Baucke, 2008) is a virtualization framework that proposes to virtualize wireless nodes as well as the radio spectrum. In this model, the virtualization manager which is an InP-side component, takes virtual node instantiation requests from the prospective VNOs and upon the availability of resources creates new virtual nodes on a shared physical node. The paper however does not give any insight on how isolation would be managed among the incumbent VNOs that share a common physical node. Also the authors proposes to use various multiple access schemes (e.g., CDMA, TDMA, FDMA) for spectrum virtualization. But how to handle the added degree of complexity due to the virualization of radio spectrum is not discussed.

The spectrum virtualization layer (SVL) presented in (Tan *et al.*, 2012a) is a sub-PHY layer that provides transparent abstraction for spectrum allocation. It allows dynamic spectrum allocation (DSA) to be implemented in a technology agnostic spectrum manager. SVL enables abstraction of the radio front-end which is very important for sharing (i.e., virtualizing) of the physical front-end by multiple players. One of the major advantages of SVL architecture is that it is fully implemented in software using the Sora (Tan *et al.*, 2011) platform.

Picasso (Hong *et al.*, 2012) is a full-duplex (FD) transceiver system that can simultaneously transmit and receive signals using the same frequency band. This is a significant breakthrough in the traditional half-duplex transceiver systems that we use today. The major problem of designing a FD system is the leakage transmit power received at the receiver chain, which is orders of magnitudes higher than the received signal. This phenomenon knows as self-interference (SI) makes realizing a FD difficult. Picasso resolves the SI problem by reducing SI using both analogue and digital cancellation techniques. Moreover it enables spectrum slicing using special purpose FPGA-based digital filters.

The architectures on programmable radio plane are summarized in Table 1.2.

Table 1.2 Representative summary of work on programmable radio plane

Prog.	Arch.	Prog. domain	HW platform	Layers affected	Virt.	Summary
Front-end	Sora (Tan *et al.*, 2011)	Protocol	GPP, FPGA	PHY, MAC	No	Programmable SDR platform using GPP servers.
	OpenRadio (Bansal *et al.*, 2012)	Protocol	FPGA	PHY	No	A programmable wireless data plane.
	MPAP (He *et al.*, 2010)	Protocol, radio front end	GPP, FPGA	PHY	Yes	It is a SDR application based on Sora (Tan *et al.*, 2011) platform,supports multiple radio standards.
Front-end & Spectrum	Virtual Radio (Sachs and Baucke, 2008)	Radio nodes, spectrum	Special purpose	PHY, MAC	Yes	A virtualization framework for radio spectrum and wireless nodes.
	SVL (Tan *et al.*, 2012a)	Radio spectrum	GPP, FPGA	sub-PHY Front-end & Spectrum	Yes	A sub-PHY layer providing transparent abstraction for DSA.
	Picasso (Hong *et al.*, 2012)	Radio spectrum, front end	FPGA	PHY	Yes	A RF front-end architecture with full-duplex transmission capability.

1.3 Cloud Computing for Wireless Networks

Cloud computing is a relatively new paradigm for large scale distributed computing. The major benefit of a cloud-based infrastructure is its ability to provide on-demand computing resources in convenient pricing schemes, e.g., pay-as-you-go, paying for the leased resources that can be elastically scaled up or down depending on cloud clients' demand at a specific point of time. Cloud resources are basically composed of storage, computing and networking elements.

When cloud computing technology is extended to the virtual wireless network domain, besides the traditional cloud resources (e.g., compute, storage, network, etc.), a cloud provider should also provide access to various wireless access nodes (e.g., APs, BSs, Repeaters, sensor nodes, etc.), core network elements (e.g., EPC for LTE core network) as well as access to wireless radio spectrum. VNOs will build their customized networks with the afore mentioned leased resources.

In this section, we shall discuss various cloud-based proposals for wireless networks. We have classified the state-of-the-art for cloud-based wireless networks according to the network types, i.e., cloud solutions for cellular, WiFi and heterogeneous networks. We also point out the added value of the proposals and the missing elements for end-to-end virtualized wireless network provisioning.

1.3.1 Cloud solution for Cellular Networks

Cloud-based solutions for cellular networks have been proposed for elastic, on-demand resource provisioning, reduction of network cost and easier management of the network infrastructure. We discuss some of the notable proposals of such cloud-based solutions from both industry and academia in this section. In the position paper (Bosch *et al.*, 2011), Virtual Telco, a cloud based architecture for telecommunications networks is presented. It proposes to replace several expensive centralized telecommunication control plane functionalities as distributed applications. These applications should be available on-demand and would be implemented over a pool of computing and networking resources. Operators will manage pooled hardware resources and thus basically serve as a infrastructure-as-a-service (IaaS) providers. As a use case of virtual telco, a distributed mobility management entity (dMME) for LTE core network has been also studied in this paper. The virtual Telco solution is basically a distributed cloud-based solution for several key cellular core network functionalities. The impact of such form of virtualization on the RAN below has not been investigated. Also, the virtualization techniques to be used for such virtualization have not been discussed in the paper.

China Mobile Research Institute (CMRI) (cmr, 2011) proposed a Cloud RAN (C-RAN) architecture, where data processing functionalities (layer 1 to layer 3) of BSs are pooled for centralized processing and radio access is provisioned via fiber-fed RRHs. Two modalities of C-RAN architecture are discussed: one is full-centralization, where layer 1 to layer 3 functionalities are implemented centrally and the other is partial-centralization that implements baseband (layer 1) processing as part of the RRH and all other functions in centralized pool. C-RAN is a virtual cloud-based implementation for cellular access networks where various PHY and MAC layer processing functionalities are implemented as software instances. This model does not discuss the VNF-based implementation of core network functionalities; also, spectrum sharing techniques among different VNOs are also not explained.

The wireless network cloud (WNC) (Lin *et al.*, 2011) was proposed by an IBM research group. The structure is composed of a radio front-end device that consists of RRH, antenna and A/D, D/A converters and IT-grade server platform where all the PHY and MAC layer processing take place. Besides the IT-grade servers, to satisfy the computational demand of PHY layer processing, FPGA-based implementation of the channel decoders has been proposed in the WNC architecture. 10 GbE or InfiniBand technology has been recommended to carry CPRI protocol over the optical front-haul from the baseband pool to RRHs. Timing synchronization in a TDD-based implementation has been proposed to be implemented using the IEEE 1558 precision timing protocol (PTP). A TDD WiMAX based adoption of the architecture was implemented in (Zhu *et al.*, 2010). The testbed studies the virtualization performance of a very limited number of implemented VBSs, and the scalability of such a platform for large-scale VBS pool deployment has not been discussed in the paper.

The CloudIQ (Bhaumik *et al.*, 2012) framework implements the baseband processing of BSs in a general purpose hardware platform. The authors in the paper show that at least a 22% savings can be achieved in computing resources by exploiting the variation in processing load among different BSs, when the baseband processing of a geographically grouped BSs is centralized in a common IT platform. OpenAir (ope, a), an open source implementation of LTE standard, was used to implement the CloudIQ framework. This paper mainly focuses on the computing resource management and savings in the processing load when BSs in a certain area are grouped to be processed in a common IT platform to achieve a certain statistical guarantee. End-to-end virtual cellular network implementation was not studied in this work.

Kempf et al. (Kempf *et al.*, 2012) proposes to move the control plane of the evolved packet core (EPC) of 4G networks to the cloud using SDN. Two extensions to the OpenFlow (McKeown *et al.*, 2008) version 1.2 are used to centralize the control plane of the EPC in a data center. The extensions used are: defining virtual ports that allow packet encapsulation and decapsulation and the other is to allow flow routing using the GPRS tunnelling protocol (GTP) Tunnel Endpoint Identifier (TEID). As a result, the GTP control plane can be decoupled from the serving gateway (S-GW) and the packet data network gateway (P-GW) and moved to a virtual machine (VM) situated in a data center. This proposal shows the strength of SDN technology in implementing cloud-based virtual systems. However, this paper addresses issue of cloud-based implementation of a specific cellular protocol, and it does not give a solution to cloudify the heterogeneous networks as a whole.

Huawei's SoftCOM (sof) is a vendor perspective towards a fully cloud network architecture. It envisions the cloud-based network architecture in four dimensions: Equipment-Level Cloud-Lization (decoupling hardware from the software), Network-Level Cloud-Lization (decoupling the forwarding plane from the control plane), IT system Cloud-Lization (using IT infrastructure for telecommunication purpose) and Internetized Operation (to transform telecommunications systems to internet-oriented systems). It is an all-encompassing virtualized cloud-based approach aiming to reduce CAPEX and OPEX for network operators. Important issues e.g., virtualization of the radio spectrum (both licensed and unlicensed bands), slice management for operational management of VNOs in an InP platform will need to be analyzed for successful realization of such a platform.

EASE (Taleb *et al.*, 2015) is an on-demand cloud-based model for elastic mobile core networks. The article discusses the feasibility of on-demand creation of elastic cloud-based service for EPC with their life cycle management. The authors also present several implementation variants of EPC-as-a-service model focusing on full and partial virtualization approaches.

1.3.2 Cloud solution for Heterogeneous Networks

Some cloud-based solutions for wireless networks target adhoc networks, a mix of WLAN and cellular networks, for flexible resource provisioning and easier network management. A SDN-based cloud architecture for mobile adhoc networks is presented in (Ku *et al.*, 2014). An extension of the OpenFlow (McKeown *et al.*, 2008) protocol is used to implement wireless adhoc scenarios. Here, the nodes can operate in multiple radio access technologies (RATs), and each node has a local controller that operates on behest of a central controller. If the connection to the central controller is unavailable the local controller falls back to operate using traditional ad-hoc protocols. The authors have simulated SDN routing as a cloud application to showcase the feasibility of such an implementation. It is to be noted that, while this proposal shows the feasibility of SDN-based cloud implementation of mobile ad-hoc networks, this is a very specific case of a wireless network implementation and is not suitable for an infrastructure based wireless network.

Carmen (Kim *et al.*, 2012) is a cloud-centric network architecture for providing seamless mobility in a mobile personal grid (MPG) which is a collection of networked devices owned by a user. In this architecture, the meta-states of a MPG are maintained in the cloud and a connectivity maintenance entity called *Avatar* ensures the situation-aware mobility of the user in the MPG. Carmen is a user-centric cloud approach for managing a user's mobility in different wireless environment and various user devices belonging to the same user. This is a conceptually different approach than the end-to-end virtual wireless network deployment. Rather than virtualizing the communication network, it virtualizes the user space that consists of the wireless environment a user moves in and different devices it uses.

Concert (Liu *et al.*, 2014) is a cloud-based architecture for cellular network edge. It uses SDN to decouple the control and the data planes to facilitate management of network applications. It distributes computing resources at different location to facilitate latency-dependent applications which is more like partial-centralized processing discussed in (cmr, 2011). It uses a control plane entity called *conductor* that takes care of virtualization and orchestration of data plane resources. This architecture is very interesting for its ability to virtualize the edge network and also for tackling the resource placement issue to meet the delay requirement of delay-sensitive applications. But virtualization of the core network and efficient slice management in an end-to-end programmable virtual network are the missing pieces in such a solution.

FluidNet (Sundaresan *et al.*, 2013a) is a framework for dynamically reconfiguring the backhaul in a cloud-based radio access network for small cells. It implements logically re-configurable front-haul to apply appropriate transmission strategies that matches user profile and dynamic traffic load pattern. It serves the dual purpose of maximizing traffic demand satisfaction in the access network while optimizing compute resource utilization at the BBU pool. The authors have shown that FluidNet (Sundaresan *et al.*, 2013a) achieves 50% improvement in traffic load satisfaction while minimizing BBU resource usage by 50%. Hence, this architecture further improves the efficiency of the C-RAN model.

The iJOIN (ijo) project proposes a RAN-as-a-service (RANaaS) architecture where the radio access network is implemented using virtualization in a cloud infrastructure. Rather than full centralization, it provides flexible centralization of the RAN functionalities and offers it as a service. This provides a compromise between achievable flexibility and depth of virtualization that VNO can choose during its negotiation phase with the InP.

Follow Me Cloud (FMC) (Taleb and Ksentini, 2013) is a framework for smooth migration of all or only a required portion of an ongoing IP service between a UE and the serving data center (DC). In this approach, to ensure the best quality of experience (QoE), mobile cloud services follow the respective users by migrating all or part of the services to an optimal DC. The feasibility of the FMC concept has been proven via an OpenFlow (McKeown *et al.*, 2008) based implementation in (Taleb *et al.*, 2013). This proposal gives a solution for QoE management in a cloud-based virtual network implementation. A summary of the cloud based wireless network architecture in given in Table 1.3.

Table 1.3 Representative Summary of Cloud-Based wireless Networks

Network Type	Proposals	Virt.	Layers	BB HW	Key Attributes
Cellular	Virtual Telco (Bosch *et al.*, 2011)	Yes	Core	Traditional equipment	Distributed implementation of some core network functionalities.
	C-RAN (cmr, 2011)	Yes	Access	GPP servers	Partial & full centralization of BBU pools
	WNC (Lin *et al.*, 2011)	Yes	Access	GPP servers	VBS implementation in GPP servers.
	CloudIQ (Bhaumik *et al.*, 2012)	Yes	BB processing	GPP servers	Saves processing load for BB implementation in GPP servers.
	(Kempf *et al.*, 2012)	Yes	Core (EPC)	Traditional equipment	Uses extension to OpenFlow (McKeown *et al.*, 2008) to virtualize EPC.
	SoftCom (sof)	Yes	Core, Access	GPP servers	Provides cloud abstraction for node, network and system.
	EASE (Taleb *et al.*, 2015)	Yes	Core	GPP servers	Implements EPC-as-a-Service for cloud-based EPC.
HetNet	(Ku *et al.*, 2014)	Yes	Adhoc	Both GPP servers and special purpose HW	Uses controller redundancy for SDN-based cloud architecture for Adhoc networks.
	Carment (Kim *et al.*, 2012)	Yes	Access	Special purpose	User-centric mobility management for MPG.
	Concert (Liu *et al.*, 2014)	Yes	Core, Access	GPP servers & special purpose	Distributed computing resources for managing delay-sensitive traffic.
	Fluidnet (Sundaresan *et al.*, 2013a)	Yes	Access	GPP Servers	Dynamically reconfigurable back-haul for C-RAN.
	iJOIN (ijo)	Yes	Access	GPP Servers	RANaaS architecture for virtual RAN provisioning.
	FMC (Taleb and Ksentini, 2013)	Yes	Core	GPP servers	A framework for migration of ongoing IP service between a UE and the service data center.

1.4 Full Duplex (FD) systems

Spectrum scarcity is one of the major impediments of today's network. Wireless network virtualization will further deteriorate the problem as multiple virtual networks will compete for access to shared radio resources. In such a scenario, FD cellular systems can mitigate the spectrum scarcity problem to a great extent by simultaneously using the wireless channels for both uplink (UL) and downlink (DL) transmissions. FD system has the potential to almost double the spectrum capacity.

The major problem for the FD communication of a transceiver is the leakage power from the Tx to the Rx which is referred to as self interference (SI). The SI is so much higher than the received power that it makes decoding of the received signal almost impossible. In recent time, a significant progress has been made to reduce the SI to a great extent by a combination analog and digital cancellation techniques. Such techniques allow to reduce the SI to such lower values that decoding of the weak received signals become possible. SI cancellation using multiple antennas were studied in (Bliss and et al., 2007), (Choi and et al., 2010), (Khandani, 2010), (Haneda and et al., 2010). FD operation in a single cell was studied in (Goyal and et al., 2014), (B.Di and et al., 2014), (Barghi and et al., 2012).

But FD operation in a multi-cell environment is very challenging because the lack of synchronization in UL and DL transmissions gives rise to a complicated interference scenario. Investigation in FD multi-cell systems is starting to gain momentum in recent time (Huawei, 2015), (Chung and et al., 2015). DUPLO (DUPLO, 2012) project is investigating the FD system for cellular small cell deployment; a joint UL-DL beamforming was designed for single cell deployment in (Nguyen and et al., 2014).

As can be seen from the aforementioned works (cf. Table 1.1, Table 1.2 and Table 1.3), they propose wireless network architectures that leverage SDN, cloud computing and programmable radio plane. Each proposal tries to solve a particular aspect of wireless network infrastructure, e.g., core or access networks or radio transceiver and/or radio spectrum virtualization. But a comprehensive architectural solution for end-to-end wireless network virtualization is absent in open literature. Also, the existing proposals only considers the technical aspects of wireless network virtualization. The economic aspect of virtualized network deployment and management is not investigated but it is a critical part of any commercial network deployment. In this thesis, we propose three different frameworks for wireless network virtualization that considers end-to-end service provisioning. Network deployment and management costs have also been considered in addition to achievable QoS to compare the suitability of the frameworks for different implementation scenarios (Chapter 2). Detailed architectural blueprint of an end-to-end programmable, elastic virtualized wireless network has been proposed in chapter 3 that leverage SDN and cloud computing technologies. To further investigate the efficacy of such programmable virtual wireless networks, system-level implementation of SDN-based virtual heterogeneous wireless networks have been implemented in chapter 4. Also provisioning of differentiated services in such virtual network environment has also been investigated. Extensive system-level simulation shows that such programmable virtualized wireless networks are able to meet the QoS requirements of carrier networks. Finally, as solution to mitigate spectrum scarcity problem in a virtual wireless network scenario, FD transmission in multi-cell system has been proposed. It has been shown that the novel user scheduling and power control algorithms can provide significant throughput gain in a FD system compared to the traditional half duplex (HD) systems.

CHAPTER 2

DESIGN OPTIMIZATION OF WIRELESS ACCESS VIRTUALIZATION BASED ON COST & QOS TRADE-OFF UTILITY MAXIMIZATION

2.1 Introduction

Traditional cellular networks are designed to serve the peak network traffic demand. This often results in over-provisioning of network resources (Zhou and Chen, 2014), which is very expensive in terms of network deployment as well as operational costs. Network operators cannot benefit from on-demand resource provisioning which would allow them to scale-up or scale-down network resources according to traffic demand at any given instant of time. Moreover, the use of complex control plane protocols and vendor locked-in devices are not amenable to provision new cellular services that might require the implementation of novel protocols or signal processing schemes. Future 5G networks will demand a more flexible and elastic network architecture that will facilitate provisioning novel services at a lower network cost, which is not possible with current network architectures. To resolve these issues, it is imperative to re-architect current network structures in new ways that make most efficient use of available resources, use less expensive general-purpose hardware rather than expensive special-purpose hardware in order to reduce overall network cost and provide flexibility to incorporate new network technologies using programmable and elastic network infrastructure (Pentikousis *et al.*, 2013). Virtualizing wireless access solves to a great extent the aforementioned problems.

In a virtual access topology, *independent and isolated* virtual networks are built on one or more physical network substrates in which the virtual networks are transparent to each other in terms of presence. The virtual networks are able to use customized network protocols, signal processing and network management functionalities that best suit the intended services. Wireless network virtualization has been approached from different perspectives: spectrum virtualization (Perez *et al.*, 2009),(Zaki *et al.*, 2010a), as well as virtualization for different wireless technologies (i.e., WLAN, WiMAX, LTE) (Singhal *et al.*, 2008),(Bhanage *et al.*, 2010a),(Bhanage *et al.*, 2010c),(R.Kokku *et al.*, 2012b),(Yap *et al.*, 2010b),(Zhu *et al.*, 2010). Major telecommunication vendors and operators are teaming up for research in network function virtualization (NFV) (Chiosi *et al.*, 2012b). The FP7-iJOIN project (fp7) is investigating the use of cloud computing for a radio access network as a service (RANaaS) paradigm, where RAN functionalities are distributed among decentralized and centralized network entities. The model focuses on handling interference in a dense network environment consisting of a large number of small (femto) cells. For front-haul, it uses either wireless or optical transmission links. Software defined networking (SDN) is being seen as a crucial driver to virtualize wireless access (Lin *et al.*, 2014),(Pentikousis *et al.*, 2013),(Bernardos *et al.*, 2014) and core (Li *et al.*, 2012b),(Karagiannis *et al.*) networks due to its ability to introduce network flexibility by separating the control and data planes. Cloud computing is also being investigated as a significant enabler towards a shared and elastic virtual wireless network (cmr, 2011),(Rost *et al.*, 2014),(Sundaresan *et al.*, 2013b).

Each of the aforementioned works tries to solve a particular problem pertaining to virtualization but a unified solution to wireless access network virtualization that incorporates virtualization of radio resources, computing & storage resources and the underlying network fabric is absent in the open literature. Different radio access technologies (RATs) use different physical, MAC and network layer processing techniques. Hence, a virtualization solution targeted to one particular RAT (e.g., WiFi) might not be applicable to another (e.g., 3G, long term evolution (LTE), etc.). In a complete virtualized platform, all network resources are virtualized. As such, it is not sufficient to virtualize processing and storage resources; the underlying network fabric must also be virtualized in order to create isolated virtual networks (VNs) on a shared infrastructure. Also provisions should be made for shared and isolated use of radio spectrum while maintaining service level agreements (SLA) between the infrastructure providers and the virtual network operators (VNOs). Hence, a unified solution to wireless network virtualization is necessary in order to facilitate shared and efficient resource utilization among incumbent VNOs, thus enabling them to implement a customized network using a common subset of network resources. Also the economic impact of various wireless virtualization models has not been analyzed in the available open literature according to the best knowledge of the author.

In this chapter, we classify wireless access virtualization frameworks in three different categories that vary in terms of their underlying physical infrastructures. We also analyze their respective network cost and achievable QoS trade-offs from PHY and MAC layer perspectives. This analysis provides guidance in selecting the best possible virtualization model for a certain implementation scenario. It also offers the following contributions:

1) the classification of virtual wireless access networks into three models (considering *green-field* deployment scenarios);
   - a special-purpose hardware-based wireless access virtualization model, referred to as Locally Virtualized Network (LVN), where a hypervisor is used to slice super base stations (SBSs) to create multiple virtual base stations (VBSs);
   - a data center based wireless access virtualization model, referred to as Clustered/Remote Virtualized Network (CVN/RVN), where SDN and cloud computing technologies are used to virtualize the underlying networking fabric and computation & storage resources. In this model, fiber-distributed remote radio heads (RRHs) are used to provide radio access to users;
   - a third model, referred to as hybrid virtualized network (HVN), where we properly combine both of the aforementioned models to offer the potential to balance network cost and QoS with greater flexibility than the previous two models (LVN and CVN/RVN).
2) a new multi-criteria utility function that accounts for network cost & QoS trade-offs to enable the design and optimization of wireless access virtualization architectures that best comply with the investment and service-level requirements of network operators (and/or service providers).

We present a LTE HetNet model as a benchmark to compare the current network deployment approach with the proposed virtualization frameworks. The remainder of this chapter is structured as follows. In Section 2.2, we briefly present the dimensioning, the cost analysis, and the time division duplex (TDD) configuration of a typical 4G LTE HetNet as a benchmark architecture without virtualization. In Sections 2.3 and 2.4, respectively, we analyze the virtualized architectures, the dimensioning, and both the capital expenditure (CAPEX) and operational expenditure (OPEX) calculations for the LVN and the CVN/RVN frameworks. Next, we subdue the HVN framework that we advocate in Section 2.5 to the same analysis exercise. The new network utility function is introduced and defined in Section 2.6, while analysis results are presented and discussed in Section 2.7. Conclusions are drawn out in Section 2.8.

2.2 Traditional Heterogeneous Network (HetNet)

We consider here as a benchmark, an architecture without virtualization based on a multi-tier LTE HetNet consisting of macro, micro, and pico cells. It is pertinent to distinguish our network modeling with the models in (Zhou *et al.*, 2010) and (Soh *et al.*, 2013). The system model in (Zhou *et al.*, 2010) considers multiple radio interfaces per node that are capable of working on multiple channels. The paper focuses on the fact that, using multiple channels through multiple interfaces will enable higher bandwidth use, which will eventually result in higher system capacity. Though the authors consider heterogeneous channels and heterogeneous traffic, they do not consider a multi-tier heterogeneous network. The system model considered in (Soh *et al.*, 2013) consists of a two-tier network having a macro-cell tier and a femto-cell tier and both are modelled following Poisson point process (PPP). Whereas in this work, we consider a three-tier network model consisting of macro, micro and pico base stations, that are distributed across the coverage area following a deterministic distribution model. Moreover, we do not consider femto BSs (FBSs) in our analysis because FBSs are user owned devices that are deployed randomly according to the preference of users, which is beyond the control of cellular network operators.

2.2.1 HetNet dimensioning

The HetNet model considered in our analysis consists of distributed smaller cells (micro, pico, and femto) with an overlay of large macro cells. While the macro cells provides network coverage, smaller cells are normally deployed to meet capacity demands in a certain area. To estimate the BS requirements, the total number of BSs needed to cover a certain area can be expressed as

$$N_{BS} = f(A, R_{BS}, R_{UE}, N_{UE}) \tag{2.1}$$

where N_{BS} is the total number of BSs, A is the coverage area, R_{BS} is the data rate capacity per BS, R_{UE} is the average data rate capacity demand per user equipment (UE) and N_{UE} is the average number of active UEs. A network can be modeled to be either coverage or capacity limited. Hence a straightforward way to model the required number of BSs is (Johansson *et al.*, 2004):

$$N_{BS} = max\left(\frac{A}{\pi d_{BS}^2}, \frac{N_{UE}R_{UE}}{R_{BS}}\right) \tag{2.2}$$

where d_{BS} is the coverage radius of a BS. It should be noted that, in (Johansson *et al.*, 2004), only single-tier architectures are considered. The authors compared operational costs when the network consisted of any BS type (macro/micro/pico BSs). On the other hand, in this work, we consider a three-tier heterogeneous network model that consists of macro, micro and pico BSs. We dimension the macro cells in coverage-limited cases in order to provide ubiquitous network coverage, whereas smaller cells (micro and pico) are deployed in capacity-limited cases, to satisfy user data rate demands.

2.2.2 HetNet cost analysis

The total cost per tier is the aggregate of capital expenditures (CAPEX), i.e., the initial set up cost of the network and operational expenditures (OPEX), i.e., the operational cost of the network per year for a specific tier. Hence the network cost for a tier i can be expressed as

$$C_i = N_{BS_i}(C_{cap_i} + C_{op_i}) \tag{2.3}$$

where C_i is the total cost for tier i, N_{BS_i} is the number of BSs in tier i, C_{cap_i} and C_{op_i} are the corresponding CAPEX and OPEX, respectively. Further, the CAPEX can be expressed as

$$C_{cap_i} = C_{eq_i} + C_{sb_i} \tag{2.4}$$

where C_{eq_i} and C_{sb_i} are the equipment and site-buildout costs, respectively. And the OPEX can be decomposed as

$$C_{op_i} = C_{sl_i} + C_{om_i} + C_{bh_i} \tag{2.5}$$

where C_{sl_i}, C_{om_i}, and C_{bh_i} are the site-lease cost, the operation and maintenance cost and the back-haul cost, respectively. The total cost for a K-tier HetNet is

$$C_T = \sum_{i=1}^{K} C_i. \tag{2.6}$$

We adopt the *cumulated discounted cash flow (DCF)* method (Kruschwitz and Loeffler, 2005) to calculate the total cost per tier i in present time. DCF analysis is a very commonly used valuation method to estimate the attractiveness of an investment opportunity, namely in terms of net present value (NPV). This is a very widely used economic tool for cost estimation of IT infrastructure. In this form of financial analysis, all the future cash flows are estimated and discounted to give their respective values in *present time*. DCF is based on the concept of time value of money, with variations in time due to inflation, capital gains, etc. Hence, in financial analysis, all future cash flows are estimated and discounted to give their present value.

In DCF analysis, to compute the NPV of an economic opportunity, all the future cash flows and a discount rate are given as input, and the output gives the NPV. Mathematically, the discounted cost of an investment, c, at a discount rate of $d\%$, can be expressed as (Kruschwitz and Loeffler, 2005)

$$C = \frac{c}{(1+d)}. \tag{2.7}$$

In case, there are multiple cash-flows at future time periods, all future cash flows should be discounted and added together to get the NPV. For example, the NPV of a cash flow in P years can be calculated as (Kruschwitz and Loeffler, 2005)

$$C = \sum_{p=0}^{P-1} \frac{c_p}{(1+d)^p} \tag{2.8}$$

where c_p is the cash flow at year p and d is the discount rate. In our analysis, one BS is exploited for Y years, hence, for a discount rate d, the net NPV for the BS is

$$c = \sum_{y=0}^{Y-1} \frac{c_y^i}{(1+d)^y} \tag{2.9}$$

where c_y^i is the cost of a BS at tier i for the year y. Here, the CAPEX, i.e., the radio equipment, site buildout and site installation costs are accounted for the first year ($y = 0$). The annual OPEX (i.e., the site lease, O & M and backhaul costs) is assumed to be constant. The OPEX values are discounted for from $y = 1$ to $Y - 1$ years to calculate the net cost value in present time. Hence, c provides the net estimate (both investment and running costs) for the entire life-cycle of the BS in present values.

Adopting a similar approach, the total cost for the K-tier network that is exploited for Y years can be calculated as

$$\begin{aligned} C_{DT} &= \sum_{y=0}^{Y-1} \frac{C_{Ty}}{(1+d)^y} \\ &= \sum_{y=0}^{Y-1} \frac{\sum_{i=1}^{K} C_{iy}}{(1+d)^y} \\ &= \left(\sum_{i=1}^{K} C_{cap_i} + \sum_{y=1}^{Y-1} \frac{\sum_{k=1}^{K} C_{op_{iy}}}{(1+d)^y} \right) N_{BS_i} \end{aligned} \tag{2.10}$$

where C_{Ty} is the total cost for year y, C_{iy} is the cost of tier-i for year y. In our analysis, the discount rate, d is assumed to be 10% and the BSs in the network are assumed to be used for $Y = 5$ years. The cost values used are given in Table 2.1 (Loizillon and et al., 2002). Cost values used in the analysis are approximate, yet very representative. Since the goal of this chapter is to show the relative trend qualitatively rather than reporting exact cost values quantitatively, these representative values serve the purpose without impinging the quality or nature of the obtained results and conclusions even if a more realistic setup were to be adopted instead. To keep our analysis tractable, it should be noted that we have assumed the discount rate d to be constant for the duration of the calculation time period, i.e., Y years. However, in practice, the discount rate may vary according to various factors, such as, the inflation rate, the financial risk involved in the opportunity, and the higher value of other opportunities.

Again, we would like to emphasize that the network cost calculated in (Johansson *et al.*, 2004) is targeted for a single-tier *homogeneous* network; in contrast, the cost model in Eq.(2.10) represents a *heterogeneous* network that consists of three different types of BSs deployed either in coverage-limited (macro BSs) or capacity-limited (micro and pico BSs) cases.

Table 2.1 Traditional network BS parameter

Parameter	**MBS**	**MiBS**	**PBS**
Cell range	700 m	175 m	70 m
Capacity (R_{BS})	300 Mbps	100 Mbps	100 Mbps
Radio equipment cost [$k]	50	20	5
Site build-out [$k]	70	-	-
Site installation [$k]	30	15	3
O&M [$k/year]	3	1	1
Site lease [$k/year]	10	3	1
Backhaul transmission [$k/year]	5	5	5

2.2.3 LTE-TDD configuration

LTE operates in two different modes: Time Division Duplex (TDD) and Frequency Division Duplex(FDD). In our analysis, we have considered the TDD mode of operation due to its wide acceptance among mobile operators around the world (Lehpamer, 2002), (Borth, 1989). One other key motivation is that TDD, in contrast to FDD, could operate in full-duplex mode.

However, using TDD requires tight coordination and synchronization among network equipment in the same coverage area. For this reason, in TDD, the evolved nodes B (eNBs) operating in the same coverage area need to be synchronized with each other within the frame granularity. The switching electronics in the eNB and UE need time to toggle between the Tx/Rx modes. To facilitate this operation, a guard period (GP) is allocated in a special subframe to compensate for the switching time and the propagation delay.

Table 2.2 Special subframe configuration for normal CP

Special subframe configuration	**CP in OFDM symbols**		
	DwPTS	**GP**	**UpPTS**
0	3	8	1
1	8	3	1
2	9	2	1
3	10	1	1
4	3	7	2
5	8	2	2
6	9	1	2

The special subframe mainly takes care of the DL-UL synchronization. This frame is structured in three parts: the Downlink Pilot Time Slot (DwPTS), the GP and the Uplink Pilot Time Slot (UpPTS). Table-2.2 (3GPP TS 36.211) shows the subframe configuration for LTE-TDD using a normal cyclic prefix. The GP has to be sufficiently long to accommodate the propagation delay and the hardware switching time to properly enable the DL/UL transition.

2.3 Locally Virtualized Network (LVN)

We propose the LVN as a distributed virtualization model that consists of virtualized BSs distributed in a certain coverage zone. In this model, BSs are virtualized (or sliced) to create multiple VBSs that are operated by different VNOs. A flow-based virtualization method is adopted, where the incumbent VBSs in a physical BS are isolated at the flow-level. The virtualization models in (Bhanage *et al.*, 2010a), (R.Kokku *et al.*, 2012b) and (Yap *et al.*, 2010b) require modifications to the existing network nodes and use of a separate IT-based network in order to implement virtualization functionalities. But the LVN model proposed in this chapter uses a single network substrate composed of SBSs to implement VBSs. We use OpenFlow (McKeown *et al.*, 2008) for flow-level virtualization of the physical BSs; we also consider that the nodes in LVN are multi-RAT capable. A detailed description of the LVN model is given in this section along with its dimensioning and cost analysis.

2.3.1 LVN architecture

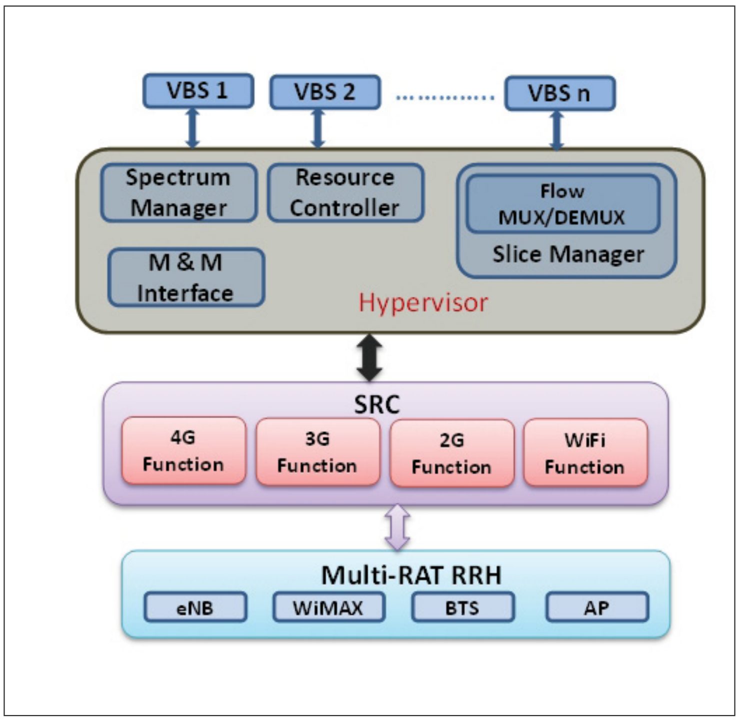


Figure 2.1 Bolck diagram of a multi-RAT SBS

For the LVN framework, we propose a BS architecture that is an enhanced version of multiple radio access technology (multi-RAT) enabled BS (Xing *et al.*, 2013) with hardware augmentation (by including a hypervisor module) to make them virtualization-capable. We refer to these newly created base stations as super base stations (SBSs). The multi-RAT SBSs are capable of supporting multiple wireless access technologies (e.g., WiFi, 3G, OFDMA based 4G systems, etc.) simultaneously to serve user equipments (UEs) using one or some if not all of these RATs. The major enhancement in the SBS architecture (cf. Fig. 2.1) is the ‘Hypervisor’ block, which virtualizes (or slices) the physical SBS into multiple virtual BSs (VBSs). Traditional BSs are operated by a single operator; hence, all the hardware (processing, storage, transmission, etc.) and radio resources are exploited and managed by that operator. On the contrary, an SBS is sliced into multiple virtual BSs (VBSs), each of which belongs to a different network operator. The hypervisor in the SBS is in fact, the virtualizing entity that manages isolation among the incumbent VBSs and provisions hardware and radio resources among them according to the service level agreement (SLA) between the virtual network operators (VNOs) that operate the VBSs and the infrastructure provider (InP), which is responsible for deployment and management of the SBSs.

The hypervisor consists of four components: a resource controller, a spectrum manager, a slice manager, and a management and monitoring (M & M) interface (cf. Fig. 2.1). The resource controller keeps track of the resources of the SBS and collaborates with the slice manager for proper resource provisioning. Specialized software libraries (SLs) are used to handle the resource allocation for each RAT. For example, the SL for OFDMA-based networks (LTE, WiMAX) assigns physical resources at the granularity of physical resource blocks (PRBs) of the OFDMA frame structure. Similarly, for other incumbent RATs, the corresponding SLs will partition resources depending on the underlying PHY and MAC layer technologies. The spectrum manager, which orchestrates air interface virtualization is basically a spectrum allocation entity that provides radio resources to the VBSs according to their need and corresponding SLAs.

The VBSs residing in the physical SBS need to be functionally isolated from each other, so that, the operation of one does not interfere with the other. As such, the VNOs operating the VBSs should do so in a way equivalent to possessing a physical base station themselves. This is provisioned by the slice manager that isolates the incumbent VBSs in flow-level. Traffic flow from the VBSs in the downlink (DL) direction is intercepted by the hypervisor and the slice manager decides which RAT module in the SRC unit this flow should be sent to. Slice-IDs are used to distinguish flows from different VBSs. Similarly, in the uplink (UL) direction, traffic flows coming from the SRC are checked for the slice-ID by the slice manager to decide on their destination VBS and directs the flow to the appropriate VBS. The flow multiplexing/de-multiplexing unit in the slice manager is responsible for the flow management in the DL and UL directions. The slice manager does the flow level virtualization (Sherwood *et al.*, 2009). For proper management of the wireless access, a VNO needs to monitor the state of its nodes and act if any change is needed. This functionality is provided by the M&M application programming interface (API) of the hypervisor.

The hypervisor interacts with the single radio controller (SRC) (Xing *et al.*, 2013), which is a unified network controller for multi-standard radio resource management. As we can see from Fig. 2.1, the SRC has 4G, 3G, 2G, and WiFi function modules which manage the corresponding transceiver units at the multi-RAT RRHs. The core network can be virtualized as described in Section 2.4.

2.3.2 LVN dimensioning

Let the number of operators in area A be n_{op}. Assuming the number of slices per SBS, n_{sl}, the required number of SBSs in area A is,

$$N_{SBS} = \frac{n_{op}}{n_{sl}} \max\left(\frac{A}{\pi d_{sbs}^2}, \frac{N_{ue} R_{ue}}{R_{SBS}}\right) \tag{2.11}$$

where d_{sbs} is the coverage radius of a SBS. In our network planning, we deploy macro-SBS in the coverage-limited case, whereas smaller (micro and pico) cells are deployed in the capacity-limited case according to traffic demand in specific places (e.g., hotspots).

2.3.3 LVN cost analysis

Since the SBS is basically an augmentation of a traditional BS, we adopt its cost as a reference value when calculating the SBS cost. We suppose that the cost of every SBS increases by γ (= 20%) with each extra slice its houses. This is just a simplified assumption to account for the economies of scale made possible by SBS resource sharing. So, the cost of the SBS radio equipment is

$$C^s_{eq_i} = C_{eq_i}[1 + \gamma(n_{sl} - 1)] \tag{2.12}$$

where C_{eq_i} is the cost of a traditional BS at tier i, n_{sl} is the number of slices in a SBS. Expenditures for site build out, site leases, power consumption and O&M are approximated in a similar fashion. Hence, the total cost per tier is

$$C^l_i = N_{SBS_i}(C^l_{cap_i} + C^l_{op_i}) \tag{2.13}$$

where N_{SBS_i} is the total number of SBSs in tier i, C^l_i is the total cost for tier i, $C^l_{cap_i}$ and $C^l_{op_i}$ are the corresponding CAPEX and OPEX, respectively, for a SBSs in tier i. Further, the CAPEX can be expressed as

$$C^l_{cap_i} = C^s_{eq_i} + C^s_{sb_i} \tag{2.14}$$

where $C^l_{eq_i}$ and $C^s_{sb_i}$ are the equipment and site-build cost, respectively. The OPEX can be decomposed as

$$C^l_{op_i} = C^s_{sr_i} + C^s_{om_i} + C^s_{bh_i} \tag{2.15}$$

where $C^s_{sr_i}$, $C^s_{om_i}$ and $C^s_{bh_i}$ are the site-rent, operation & maintenance (power consumption and maintenance), and backhaul costs, respectively. Hence, the total cost for the LVN is

$$C^l_T = \sum_{i=1}^{K} C^l_i. \tag{2.16}$$

We use the *cumulated discounted cash flow* method to calculate the total cost per tier i in present time. If, on average, one BS is exploited for Y years, then for a discount rate d, the total cost can be calculated as

$$\begin{aligned} C_{DT}^{l} &= \sum_{y=0}^{Y-1} \frac{C_T^l}{(1+d)^y} \\ &= \sum_{y=0}^{Y-1} \frac{\sum_{i=1}^{K} C_i^l}{(1+d)^y}. \end{aligned} \tag{2.17}$$

2.4 Clustered/Remote Virtualized Network (CVN/RVN)

The CVN/RVN is a cloud-based virtualization framework. In this model, computing, storage and networking resources are pooled in wireless data centers that we refer to as central processing centers (CPCs). In a CPC, BS functionalities are implemented as software instances on IT-grade servers and radio access is provided via fiber-connected, distributed and multi-RAT RRHs. When a single *large* CPC is used to cover a certain geographical area A, we refer to this network as a remote virtualized network (RVN). When a number of *smaller* CPCs are distributed to cover the area A, the network is called a clustered virtualized network (CVN). A typical CVN architecture is shown in Fig. 2.2 that consists of distributed data centers interconnected by a metropolitan optical network (MON) which is composed of optical cross connects (OXCs) and fiber optic cables. We advocate the use of SDN and cloud computing as enabling technologies for implementing the proposed CVN/RVN model. By separating the control and data planes, SDN enables network programmability and innovative service provisioning in otherwise closed telecommunication networks. Resource sharing as well as elastic and on-demand resource provisioning are possible in the new cloud computing paradigm. There are mainly three parts in this architecture: the Network Orchestrator (NO), the Radio Access Network (RAN) (cf. Fig. 2.4), and the Core Network (CN) (cf. Fig. 2.5). We discuss the detailed architectural components of the CVN/RVN framework in this section. We also present the dimensioning of a CVN/RVN network that follows with the cost analysis of this model.

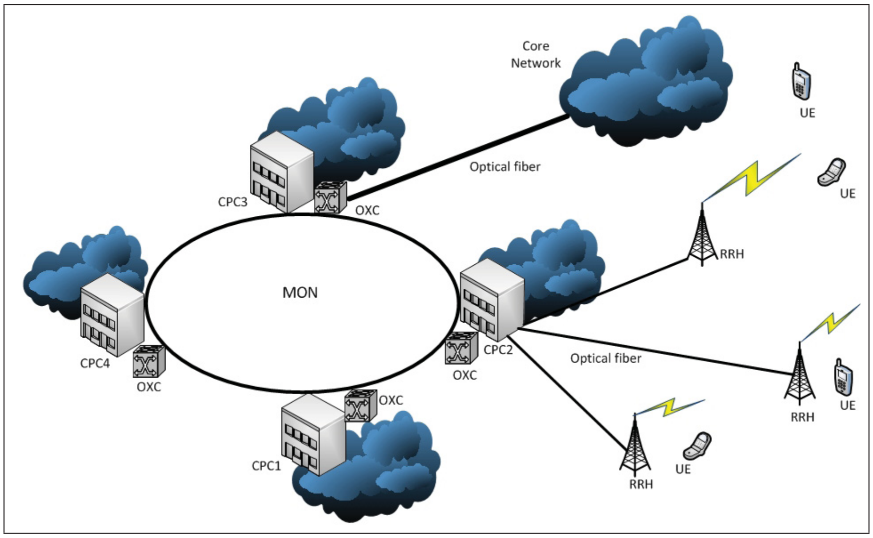


Figure 2.2 Clustered Virtualized Network (CVN) Architecture

2.4.1 Network orchestrator (NO)

The NO is the central control point for both the access and core networks. It controls the underlying physical and virtual resources. It consists of both RAN and CN controllers. It also provides a configuration & monitoring interface to the VNOs and SPs. Each VNO has a network controller that manages the underlying SDN-based network fabric. The compute & storage controller manages the computing and storage resources. The conventional NO is motivated by the SDI resource management system in (Lin *et al.*, 2014), which is used to control and manage the underlying networking & computing resources in a wired network environment. The flow-chart in Fig. 2.3 shows the various steps involved in the NO's decision making in the creation and subsequent operation of VNOs. A prospective VNO requests its required resources from the NO (managed by an InP). The NO consults its resource database to see if the VNO's request can be satisfied. If resources are insufficient, it would notify the VNO that its request cannot be fulfilled. But if the InP has available resources to satisfy the VNO's demand, the compute & storage controller of the NO will allocate these resources to the VNO. The VNO can then install its virtual network functions (VNFs) (e.g., switching gateway (SGW), packet data network gateway (PGW), mobility management entity (MME), etc. for a MVNO case) in the allocated memory locations. Similarly, the network controller unit of the NO assigns network resources in accordance to the VNO's request. The VNO can build its customized network using its own network controller application that programs the underlying programmable switching and radio plane devices. Hence, a VNO has its own network consisting of VNFs and a virtual network.

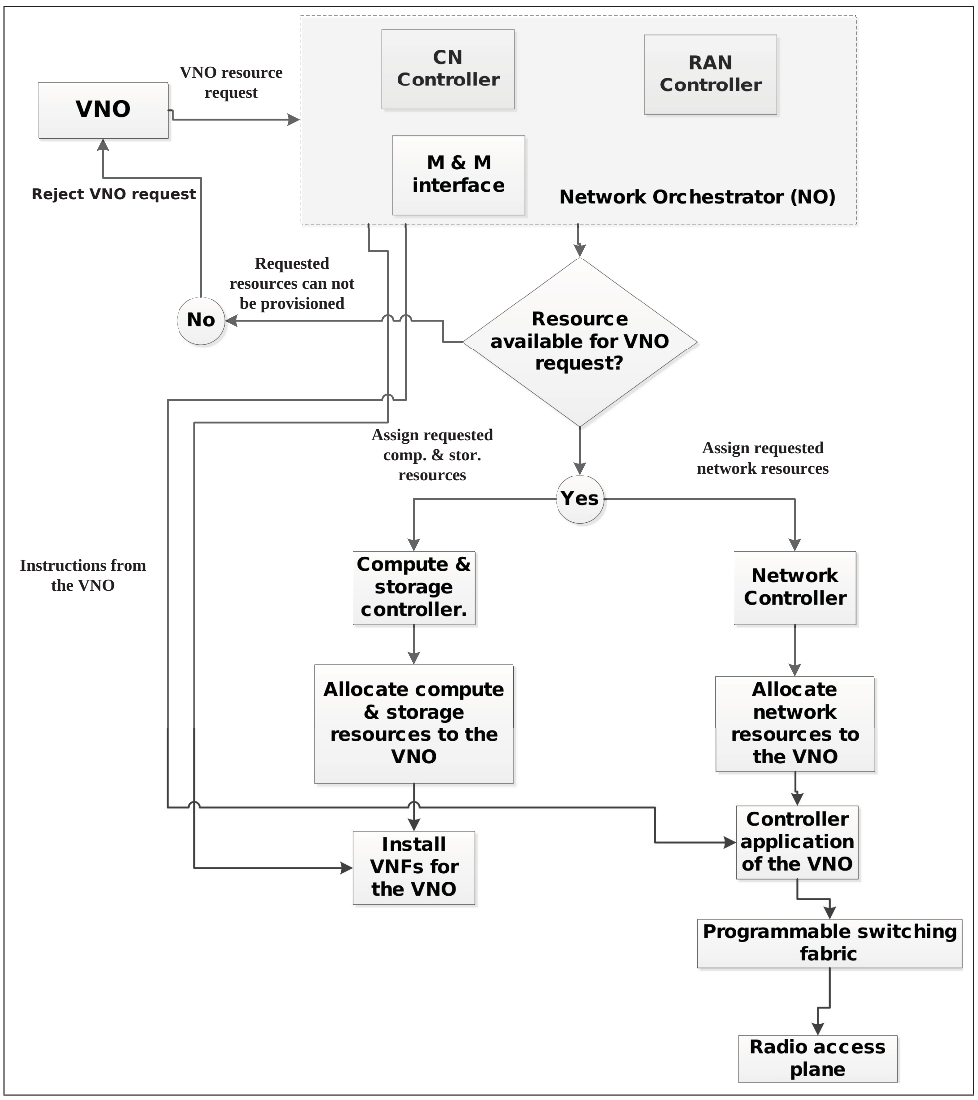


Figure 2.3 Flow chart showing a NO's decision steps

2.4.2 Radio access network (RAN)

The CVN/RVN RAN consists of the network fabric and the compute & storage parts. A detailed network diagram is shown in Fig. 2.4. This section describes these platforms in detail.

Network fabric

The network fabric consists of programmable switches and radio devices (RRHs) that can be programmed following the SDN paradigm. A virtual network operator (VNO) or service provider (SP) can build its own customized network in the networking fabric by programming its allocated network resources. VNOs express the functional behavior of their networks by different SDN applications. The controller platform (e.g., POX (Mccauley), NOX (Gude *et al.*, 2008), Ryu (Ryu), FloodLight (flo), etc.) converts the high level network policies from the application layer and expresses them in a form compatible with the underlying programmable switching fabric. For this purpose, the controllers use a southbound API, e.g., OpenFlow (McKeown *et al.*, 2008) to modify the forwarding behaviour of underlying switches. A multi-RAT interface layer (ADC/DAC) translates the information to the appropriate RAT by the optical (or microwave) front-haul.

For virtualizing the network fabric, a controller (e.g., FlowVisor (Sherwood *et al.*, 2009)) is used which is basically a transparent proxy that ensures isolation among the virtual operators (SDN applications). Different SDN applications (e.g., VNO, HD video provider, sports channel provider, gaming companies, etc.) can be built using a high-level network programming API (e.g., Pyretic (Reich *et al.*, 2013)). Domain-specific programming languages like Pyretic are programmer-friendly, provide high-level network abstraction, and enable a programmer's task of writing modular network applications easier.

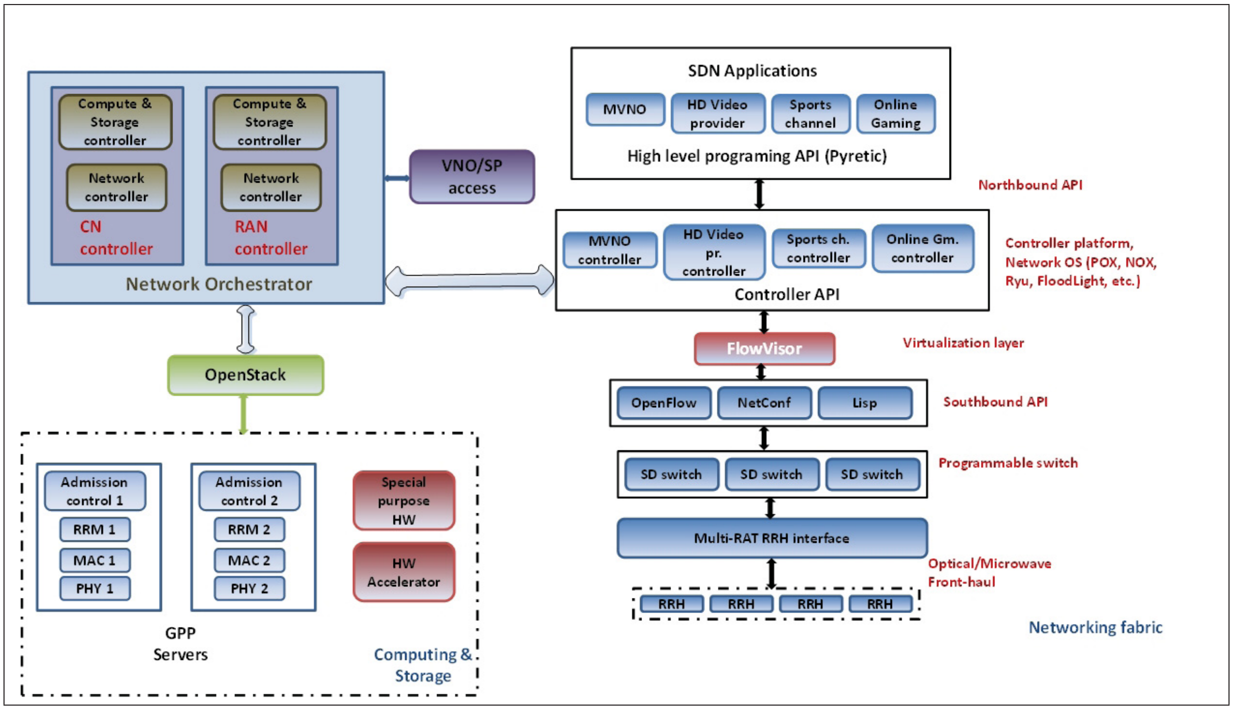


Figure 2.4 Functional block representation of a CVN/RVN RAN with a network orchestrator

Compute & storage unit

The network applications and various signal processing software components are stored and executed in the compute & storage unit. The compute & storage controller takes the high-level requirements from third parties (e.g., MVNOs and SPs) and allocates computing, storage and radio resources. For such an "infrastructure as a service (IaaS)" deployment, we have used the open source cloud computing platform, OpenStack (OpenStack).

Current heterogeneous multi-RAT technologies use different PHY and MAC layer and radio resource management (RRM) functions. To facilitate the development of customized RAT technologies, different PHY, MAC and RRM techniques are implemented as individual software modules in GPP servers (see bottom-left part in Fig. 2.4). As such, any VNO or SP can combine different modules that efficiently implement its intended service & application. A VNO can also develop its own customized PHY, MAC or RRM protocols. For demanding PHY-layer processing features, special purpose hardware and hardware accelerators are used.

2.4.3 Core network (CN)

The CN is implemented in GPP servers using OpenStack (OpenStack) technology to enable the cloud computing paradigm. It has three main parts as illustrated in Fig. 2.5, an interface layer, user-state database (DB), and CN functional modules. The CN interface layer is a communication interface with the network controller that sends/receives network configuration instructions for the computing & storage and the networking sections. It also communicates control signals and data with legacy (non virtualized) network elements. The user state DB compiles all state information for the users. Hence, the underlined virtual entities can be stateless.

The core network control-plane functions such as the mobility management entity (MME), the policy & charging rule function (PCRF), the home subscriber server (HSS), the authentication, authorization & accounting (AAA), etc. are implemented as software modules. As such, the VNOs/SPs can create their (virtual) components for the respective service provisioning. For data-plane forwarding, FlowVisor (Sherwood *et al.*, 2009) virtualizes the underlined software-defined programmable switch fabric.

The CVN/RVN model proposed here uses software instances of BSs implemented in servers with distributed fiber-connected RRHs and OpenFlow (McKeown *et al.*, 2008) for virtual wireless access rendering. It also provisions for multi-RAT RRHs. In contrast,the model in (Zhu *et al.*, 2010), considers VBS pooling over two servers only and does not analyze the more realistic case when the scale of VBS pooling becomes as large as that of a data center. Also the work in (Zhu *et al.*, 2010) does not address critical virtualization issues like slice isolation and customized network stack implementation capabilities for VNOs. Moreover, unlike the proposed CVN/RVN model, the C-RAN architecture in (cmr, 2011) does not use OpenFlow (McKeown *et al.*, 2008). As such the proposed OpenFlow-based (McKeown *et al.*, 2008) CVN/RVN architecture therefore accounts for the aforementioned features. And the radio signal transmission over fiber (RoF) actually becomes a critical issue for the implementation of large data center. The new CVN/RVN model takes into consideration the RoF issue and pro-

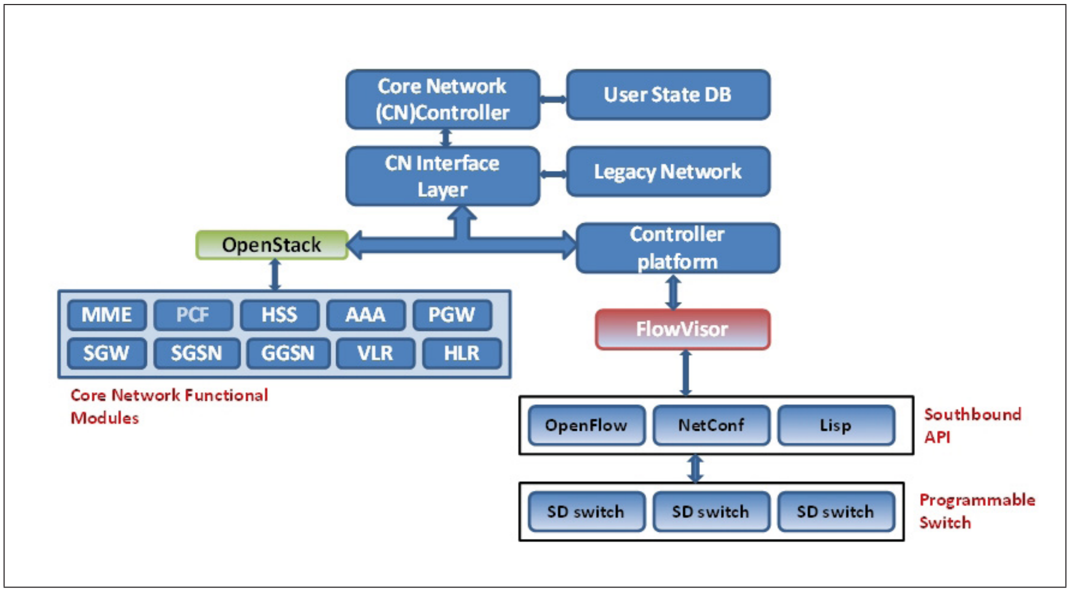


Figure 2.5 CVN/RVN core network flow diagram

vides a guideline for wireless data center dimensioning, a key aspect that has not been studied in the open literature to the best of the authors' knowledge. From a broader perspective, we envision the distributed CPCs as a "cloud of wireless data centers". As a proof of concept, a virtual heterogeneous wireless access network model was implemented by the authors of this work in (Rahman *et al.*, 2015b) using an emulation platform, where service differentiation was studied for two virtual networks that were implemented in a common subset of network infrastructure. Emulation results suggested that virtual wireless networks are able to achieve the QoS requirement of carrier networks while ensuring efficient resource utilization by sharing a common subset of network infrastructure.

2.4.4 CVN/RVN dimensioning

The required number of RRHs for macro-coverage can be calculated as

$$N_r^m = n_{op} \max\left(\frac{d_{cpc}^2}{\pi d_m^2}, (\nu_m(\mu_m A_{cpc})R_{um})/R_{MBS}\right) \tag{2.18}$$

where d_{cpc} and A_{cpc} are the CPC size and coverage area, respectively, and μ_m, ν_m and R_{um} are the user density, the HetNet coefficient (i.e., the ratio of macro, micro and pico cells) and the average user data rate, respectively. Similarly, the number of RRHs for micro-coverage is

$$N_r^{mi} = (\nu_{mi}(\mu_{mi}A_{cpc})R_{umi})/R_{MiBS} \tag{2.19}$$

and the number of RRHs for pico-coverage is

$$N_r^{p} = (\nu_{p}(\mu_{p}A_{cpc})R_{up})/R_{PBS}. \tag{2.20}$$

Please note that the work in (Zhu *et al.*, 2010) dedicates two processor cores for the implementation of one macro VBS only. Since micro and pico cells serve lower loads than a macro cell, it is intuitive that the micro and pico base stations will require less processing hardware. From a "processor core" point of view, the required number of servers required for a CPC, considering servers with eight-core processors, can be calculated as

$$N_{ser} = (N_r^{m} pc_m + N_r^{mi} pc_{mi} + N_r^{p} pc_p)/8 \tag{2.21}$$

where pc_m, pc_{mi}, and pc_p are the numbers of dedicated processor cores required for macro, micro, and pico VBSs, respectively; N_r^m, N_r^{mi}, and N_r^p are the numbers of RRHs for macro, micro, and pico cell coverage in the concerned area. It is worth noting in our analysis that we assumed each cell to have its own dedicated RRH. The number of server racks is $N_{rack} = N_{ser}/N_{ser}^{rack}$, where N_{ser}^{rack} is the number of servers per rack. The number of switches and OXCs are approximated as $N_{sw} = N_{rack}$ and $N_{oxc} = N_{rack}$, respectively.

2.4.5 RRHs cost

The RRH cost is calculated as

$$C_{rrh} = C_{rrh_c} + C_{rrh_o} \tag{2.22}$$

where C_{rrh_c} and C_{rrh_o} are the RRHs' CAPEX and OPEX, respectively. C_{rrh_c} consists of the radio equipment (c_{rrh_e}) and the site installation costs ($c_{rrh_{si}}$), whereas the OPEX consists of O&M costs only. No site lease nor backhaul costs are considered for the RRHs since fiber optic cables are used for radio signal transmission. Hence, the RRHs' cost in a CPC is

$$C_{rrh} = N_r^m C_{rrh}^m + N_r^{mi} C_{rrh}^{mi} + N_r^p C_{rrh}^p \tag{2.23}$$

where C_{rrh}^m, C_{rrh}^{mi}, and C_{rrh}^p are the separate RRHs' costs for macro, micro and pico coverage, respectively. The cumulated discounted cash flow for the RRHs over Y years is calculated as

$$C_{rrh}^t = \sum_{y=0}^{Y-1} \frac{C_{rrh_y}}{(1+d)^y} \tag{2.24}$$

where C_{rrh_y} is the cost of the RRHs in year y.

2.4.6 CPC cost

The CPC cost accounts for different expenditures that cover the data-center's occupied space, the power consumption for hardware processing and cooling, the personnel salaries, software costs, etc. For real estate expenses, we adopt the following model proposed in (Patel and Shah, 2005).

Space cost

The real estate value for a CPC per year can be calculated as (Patel and Shah, 2005)

$$C_{sp} = \frac{NOIA_{cpc}oc}{CP} \tag{2.25}$$

where NOI is the net operating income per square meter per year, CP is the capitalization rate, A_{cpc} is the CPC area and oc is its occupancy factor.

Power delivery cost

A power delivery system in a typical data center is expected to feed air conditioning, battery back-up, on-site power generation, and both delivery and generation redundancies. Depreciation or amortization and maintenance costs are associated with the infrastructure that encompasses all the aforementioned functions. Hence, the cost burden of power delivery per year can be expressed as (Patel and Shah, 2005)

$$C_{pwr} = (1 + K_p) c_e P_{HW} \tag{2.26}$$

where c_e is the cost of power delivery per watt per year, P_{HW} is the hardware power consumption, and $K_p = JC_{am}^{pwr}/c_e$ is the power burden factor where J is the capacity utilization factor and C_{am}^{pwr} is the amortization & maintenance cost per watt per hour.

Cooling cost

The cooling cost can be estimated as (Patel and Shah, 2005)

$$C_{col} = (1 + K_c) L c_e P_{HW} \tag{2.27}$$

where K_c is the cooling burden factor and $L = \frac{P_{cooling}}{P_{HW}}$ is the load factor.

Personnel costs

Let the number of personnel per rack in a data center be composed as follows: IT technicians H_{IT}, facility service employees H_f, and administrative clerks H_a. If the average yearly salary is C_{ap}, then the personnel costs per year can be calculated as (Patel and Shah, 2005)

$$C_{per} = (H_{IT} + H_f + H_a) C_{ap} \tag{2.28}$$

Software cost

The software cost for a data center is

$$C_{sofw} = N_{rack} C_{swr} \tag{2.29}$$

where N_{rack} is the total number of racks in the data center and C_{swr} is the average yearly cost of software and licenses per rack.

IT equipment cost

IT equipment consists of servers, switches, and OXC. Their cost for a CPC is calculated as

$$C_{IT} = N_{rack} C_{rack} + N_{sw} C_{sw} + N_{oxc} C_{oxc} \tag{2.30}$$

where N_{sw} is the number of switches, N_{oxc} is the number of OXCs, and C_{rack} , C_{sw}, and C_{oxc} are the unitary costs of servers per rack, switches, and OXCs, respectively.

Optical-fiber deployment cost

The optical-fiber deployment cost is expressed as

$$C_{fb} = (C_f L_{af} + C_{tr}) N_{tot}^{rrh} \tag{2.31}$$

where C_f is the fiber cost per km, L_{af} is the average optical fiber length, C_{tr} is the cost of an optical transponder, and N_{tot}^{rrh} is the total number of RRHs.

Total CPC cost

The CAPEX of a CPC is

$$C_{cap}^{cpc} = C_{fb} + C_{IT} \tag{2.32}$$

whereas its OPEX is

$$C_{op}^{cpc} = C_{sp} + C_{pwr} + C_{col} + C_{per} + C_{sofw} \tag{2.33}$$

Hence, its total cost is

$$C_{cpc} = C_{cap}^{cpc} + C_{op}^{cpc} \tag{2.34}$$

and its cumulative discounted cost is

$$C_{cpc}^{t} = \sum_{y=0}^{Y-1} \frac{C_{cpc_y}}{(1+d)^y} \tag{2.35}$$

where C_{cpc_y} is the CPC cost in year y. The total cost for a CPC network, including its distributed RRHs, is therefore calculated as

$$C_{cpc}^{T} = C_{cpc}^{t} + C_{rrh}^{t} \tag{2.36}$$

2.4.7 Total CVN/RVN cost

The number of CPCs is $N_{cpc} = A_{rgn}/A_{cpc}$, where A_{rgn} is the area covered by the network and A_{cpc} is the coverage area of a CPC. Hence the total CVN/RVN cost is

$$C_{n}^{c/r} = N_{cpc} C_{cpc}^{T}. \tag{2.37}$$

The itemized cost values of the RRHs and CPC nodes, inspired from (Johansson *et al.*, 2004) and (Patel and Shah, 2005), respectively, are listed in Tables 2.3 and 2.4. Please note that the RRHs do not incur any noticeable costs for their site build-out and lease or for their baseband signals' transmission. Also please note that the costs for the CPC nodes were properly set after careful consultation of different vendor websites and that the costs of real estate, power consumption, and other items were approximated by representative values (Patel and Shah, 2005).

Table 2.3 RRH parameter

Parameter	MBS	MiBS	PBS
Cell range (R)	700 m	175 m	70 m
Capacity (W_{BS})	300 Mbps	100 Mbps	100 Mbps
Radio equipment cost [$k]	10	4	0.1
Site build-out [$k]	-	-	-
Site installation [$k]	5	2	0.5
Annual O&M [$k/year]	0.3	0.1	0.1
Annual site lease [$k/year]	-	-	-
Annual transmission [$k/year]	-	-	-

Table 2.4 CPC cost parameter

Parameter	Cost [$k]
Server	11
Switch	8
OXC	10
Fiber optic cable	0.01/unit area
Site buildout	100
Site installation	40
Annual O&M	5
Annual site lease	15
Annual transmission	0

2.5 Hybrid Virtualized Network (HVN)

The HVN framework is a combination of the LVN and CVN/RVN models. It consists of CPCs as well as selectively-distributed SBSs. The cost advantages of the CVN/RVN depend on application-specific QoS penalties that impose minimum acceptable thresholds. To alleviate this problem, a HVN, which is basically a combination of a LVN and a RVN, offers the best cost vs. QoS trade-offs. Indeed, a HVN deploys data centers with SBSs distributed in the coverage area to meet the service requirements of delay-sensitive traffic. As one example, suppose that a data center of either RVN or CVN type covers a certain metropolitan area. Assume also that there are many offices in the downtown of that metropolitan area that generate a significant amount of voice and live-video traffic during office hours. A data center with distributed RRHs might not be able to cope with this highly delay-sensitive traffic demand. To alleviate this problem, a number of SBSs can be distributed throughout the downtown area in order to handle the delay-sensitive traffic (e.g., voice, live video, etc.) and off-load the more delay-tolerant traffic (e.g., text, file transfer, web browsing, video streaming, etc.) to the data center. A network designer has to take into consideration the demography and expected traffic patterns of any given deployment area and specify a HVN that is able to handle the traffic QoS demand in the most efficient way. A HVN model can be expressed in terms of weights as

$$HVN = p_c RVN + (1 - p_c) LVN \tag{2.38}$$

where p_c is the portion of the HVN that exploits a data center (i.e., the CVN/RVN part) and $(1 - p_c)$ is the remaining portion of the network that exploits SBSs (i.e., the LVN part).

2.6 Data Rate and Utility Function Construction

The virtualization frameworks presented in the previous sections are quite different in terms of their underlying network structure and hardware choices. Hence, they have their relative pros and cons as far as the network cost, energy efficiency (Rahman *et al.*, 2013), and QoS are concerned. As one example, using IT-grade network equipment in a CVN/RVN architecture is more cost efficient than using SBSs in a LVN framework. But carrying signals over RoF from a CPC to the RRHs (and vice-versa) has its own challenges and limitations from a QoS point of view. To investigate the trade-offs between a network operator's budget and the service quality requirements of the intended service, we have developed an analytical model for the proposed virtualization frameworks. This model considers both network cost and QoS (achievable data rate) as well as the operator's preference for cost effectiveness and service quality of the network. In our analysis, we have only considered single-RAT multi-tier networks for the sake of simplicity and conciseness. The most general multi-RAT multi-tier HetNet case is beyond the scope of this thesis and can be subject of future work. We have also considered LTE-TDD downlink transmission. The granularity of physical resources is adjusted down to the level of the physical resource block (PRB) of the OFDMA frame structure. The data rate for an

OFDMA system can be calculated as (Nuaymi, 2007):

$$R_{TDD} = \frac{N_{sub}N_{mod}N_{cod}}{[N_{FFT}/(nBW)](1+G)} \quad (2.39)$$

where N_{sub} is the number of data subcarriers and N_{mod} and N_{cod} are the numbers of modulated bits per symbol and the coding rate, respectively; *BW, n, and G* are the operating bandwidth, the sampling factor, and the cyclic prefix length, respectively.

In a TDD system, maintaining time synchronization between the uplink and downlink transmissions is critical. The lack of synchronization can disrupt proper decoding of the transmitted information. In the CVN/RVN framework, this issue is more critical since the radio propagation path involves the whole span of optical fiber between the RRHs and the CPC. The time slot in an OFDMA subframe that enables this time synchronization is called the guard period (GP). In our design, we utilize this GP to accommodate the transmission delay for carrying radio signal over the optical fiber cables that spans from the CPCs to the RRHs. The data rate for such an OFDMA system employing RoF transmission can therefore be expressed as

$$R^{*}_{TDD} = \frac{N_{sub}N_{mod}N_{cod}(T_{sf} - t_{enb} - d_{cpc}l)}{[N_{FFT}/(nBW)](1+G)T_{sf}} \quad (2.40)$$

where T_{sf} is the length of the special sub-frame, t_{enb} is the switching time of the eNB, and d_{cpc} is the coverage size the CPC, and l is the latency per km for radio transmission in the fiber. To avoid over/under provisioning, we have adopted in our analysis a square shape for both total coverage and the CPC areas.

The extra delay incurred by transmissions over the optical fiber in the transmission causes losses in the achievable *goodput*. We characterize this error as the frame error rate (FER)

$$FER = exp(-\alpha\sqrt{\delta}) \quad (2.41)$$

where $\delta = \frac{14-GP}{14}$ is the ratio of the pilot-bearing symbols to the total number of symbols in a OFDMA sub-frame and α is a parameter that models in a simple way the severity of the channel by the degradation rate at which identification and synchronization errors increase and, hence, the throughput decreases through the negative impact of a lower pilot to sub-frame ratio δ. This parameter should depend on most of the PHY-layer parameters like the channel bandwidth, the SNR, the modulation, the coding rate, etc. Taking into account the $\overline{FER} = 1 - FER$, the data rate in equation (2.40), referred to here as R_{LTE} since we consider here LTE HetNets, reduces to

$$R_{LTE} = \frac{N_{sub}N_{mod}N_{cod}(T_{sf} - t_{enb} - d_{cpc}d_l)}{[1/(n\frac{BW}{N_{FFT}})](1+G)T_{sf}}\overline{FER}. \tag{2.42}$$

Higher FER not only further degrades QoS uniformly across all types of users by reducing spectrum efficiency, but will further impact it, yet unequally, i.e., more so over delay-sensitive links, by increasing requests for packet retransmissions. While we account for the former effect on QoS, we do not for the latter's. As such, our data rate term should be properly modified to render both impediments. One way to do so is to redefine it as follows:

$$R^{'}_{LTE} = p_s R_{LTE}^{1/e_s} + p_v R_{LTE}^{1/e_v} + p_{sd} R_{LTE}^{1/e_{sd}} + p_{id} R_{LTE}^{1/e_{id}} \tag{2.43}$$

where p_s, p_v, p_{sd}, and p_{id} denote percentages (i.e., positive values less than 1) of speech (or voice), video, delay-sensitive, and delay-insensitive links, respectively, i.e., we have

$$p_s + p_v + p_{sd} + p_{id} = 1$$

and where e_s, e_v, e_{sd}, and e_{id} denote the delay-severity impact exponents for speech, video, delay-sensitive data, and delay-insensitive data links, respectively.

Now, we formulate the multi-criteria network utility function that is composed of network cost and achievable data rate. Network operators should be able to express their preference in terms of level of importance to network cost (both CAPEX and OPEX) or QoS (data rate). This preference indicates how important one criterion is against the other in the framework selection process. Since network cost and QoS are not compensatory in the selection of a particular framework, the nullity and unity of the utility function is important (Vuong *et al.*, 2013). For this reason, we compose the network utility as the geometric product of the normalized network cost and QoS gains:

$$\begin{aligned} U_{opt}(args_1) &= max_{args_2}[U(args)] \\ &= \left(\frac{C_{max}-C}{C_{max}}\right)^{w_c}\left(\frac{R'_{LTE}}{R^{max}_{LTE}}\right)^{(1-w_c)} \end{aligned} \tag{2.44}$$

where w_c and $(1-w_c)$ are the cost and data-rate weights, respectively, and $args_2$ = d_m, d_{cpc}, ϕ, ν, BW, GP, $args_1$ = *other PHY and MAC layer parameters*, and $args = args_1 \cup args_2$. Also $C_{max} = max_{(d_m,\phi,\nu)}C$ and $R^{max}_{LTE} = max_{(BW,GP,d_{cpc})}R'_{LTE}$.

2.7 Results

The choice of a certain framework essentially is based on a given compromise between the corresponding network cost and the achievable QoS. The LVN can reduce cost to some extent but its implementation complexity increases due to the pooling of (virtual) network nodes and the introduction of a hypervisor. The CVN/RVN is the most cost-effective solution due to its usage of inexpensive general purpose IT hardware for baseband signal processing. But the inclusion of optical fibers in its network architecture places limitations on the achievable QoS due mainly to additional RTTD for radio transmission over fiber optic cables. The HVN is a more balanced approach to network cost and QoS optimization. In this section, we assess the impact of the PHY and the MAC layer parameters on the CPC size. We also investigate the impact of different wireless access configuration parameters on the achievable network utility performance.

Table 2.5 Evaluation scenarios

Scenario	**BW [MHz]**	ϕ **[**$/km^2$**]**	d_{MBS} **[km]**	**HetNet** $[\%,\%,\%]$	α
1 (reference)	20	1000	0.7	[20,30,50]	1.4
2	10	1000	0.7	[20,30,50]	1.4
3	20	1000	0.5	[20,30,50]	1.4
4	20	1000	0.7	[100,0,0]	1.4
5	20	100	0.7	[20,30,50]	1.4
6	20	1000	0.7	[20,30,50]	3.0

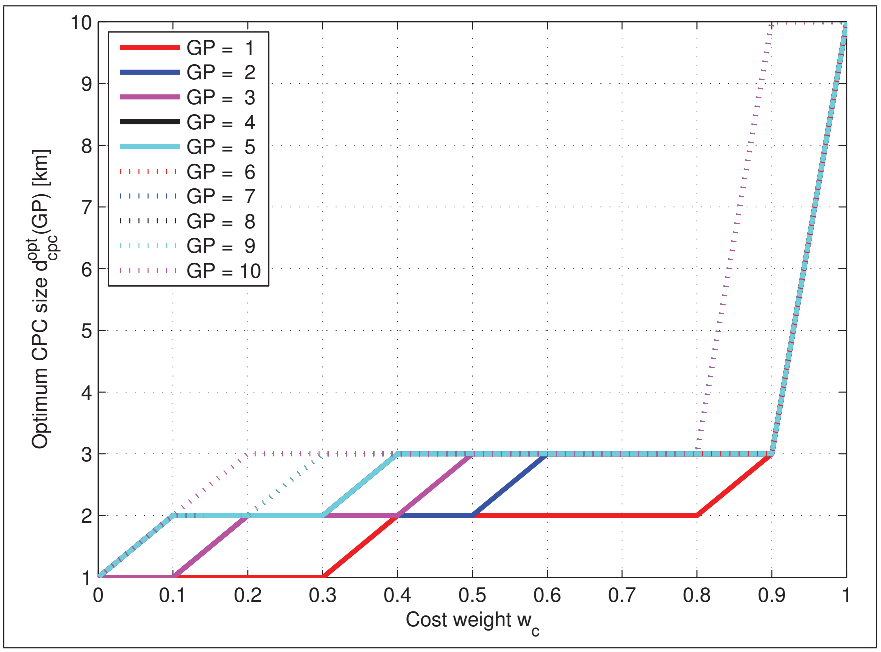


Figure 2.6 Opt. CPC size $d_{cpc}^{opt}(GP)$ vs. cost weight w_c for diff. GP values in the ref. scenario I (cf. Table 2.5)

2.7.1 Optimum CVN/RVN CPC size d_{cpc}

The optimal size of a CPC depends on many parameters such as the system bandwidth, the coverage radius of the macro base stations, the network architecture (i.e., whether it is homogeneous or heterogeneous), etc. One of the most critical parameters affecting the CPC size is the *GP* value of an OFDMA subframe. Fig. 2.6 shows how the optimum CPC size d_{cpc}^{opt} versus the cost weight w_c varies with GP values in the reference scenario 1 of Table 2.5. When the primary concern is QoS (i.e., less emphasis on cost), smaller CPCs should be preferred. But when

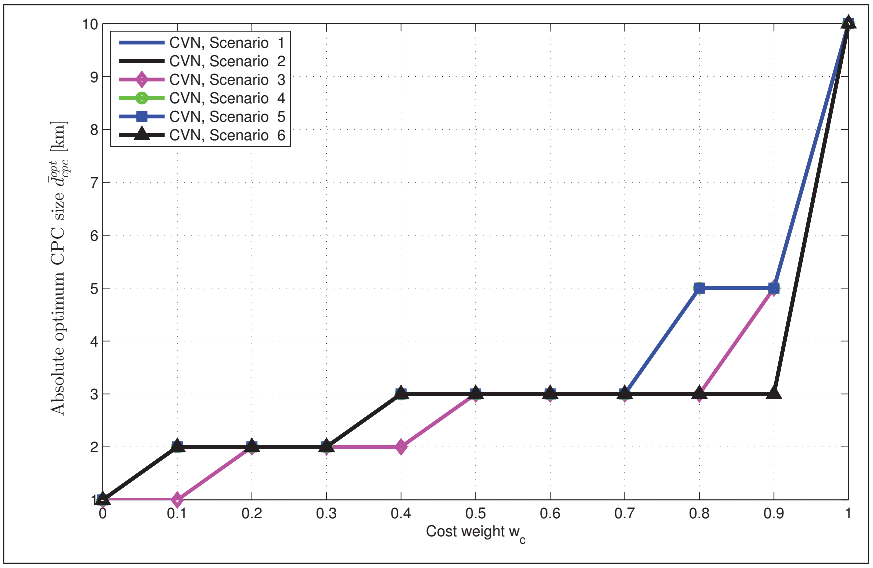


Figure 2.7 Absolute optimum CPC size $\bar{d}_{cpc}^{opt}$ vs. cost weight w_c in different scenarios (cf. Table 2.5)

the operational budget is constrained, network designers should favor relatively larger CPCs with relatively wider coverage areas. A CPC of 1 to 3 km radius in a coverage area of 20 km radius is preferred for a wide range of w_c values. Interestingly, in the extreme case when there is no budget restriction (i.e., $w_c = 1$), the optimal CPC size is with a 10 km radius, meaning that a RVN (i.e., a single CPC covering the whole area) can never be an optimal design choice. It is worth mentioning that MAC layer parameters like GP can be optimized along with the cost-QoS trade-off in a CVN/RVN model. The severity of the transmission channel condition (modelled by α) impacts the optimal GP value GP_{opt}, i.e., when $\alpha = 1.4$, $GP_{opt} = 4$ symbol periods whereas for $\alpha = 3.0$, $GP_{opt} = 5$ symbol periods. For a coverage area with 20 km radius, the impact of different parameters (cf. different scenarios in Table 2.5) on the absolute optimal CPC size $\bar{d}_{cpc}^{opt} = d_{cpc}^{opt}(GP_{opt})$ (i.e., using optimized GP value G_{opt})) is illustrated in Fig. 2.7.

2.7.2 CVN/RVN utility U_{opt} at different GP values

The effect of GP on the total utility behaviour is also of prime importance. Fig. 2.8 shows the CVN/RVN utility behavior for different GP values in the reference scenario 1 (cf. Table 2.5). The CVN has better utility performance than the RVN for the same GP value. And the utility performance of both is worst for a $GP = 1$ symbol period. Indeed the optimal values of $d_{cpc}(GP)$ become relatively the smallest in this case (i.e., d_{cpc}= 0.7 km when w_c = 0), thereby increasing the network cost by a great extent. The maximum network utility is achieved with $GP = 4$ symbol periods (when α = 1.4) because it balances both the cost and QoS in the most efficient manner. When $GP = 1$ in the RVN case, the network utility is severely penalized because just one symbol period is not large enough to account for radio propagation delays over a fiber distance of 20 km for adequate OFDM DL-UP synchronization. Hence the RVN architecture can never be a favorite choice, because the network's QoS is severely penalized due to the RVN's inability to properly resolve PHY (resolving transmission channel severity issues) and MAC (DL-UL synchronicity) layer issues.

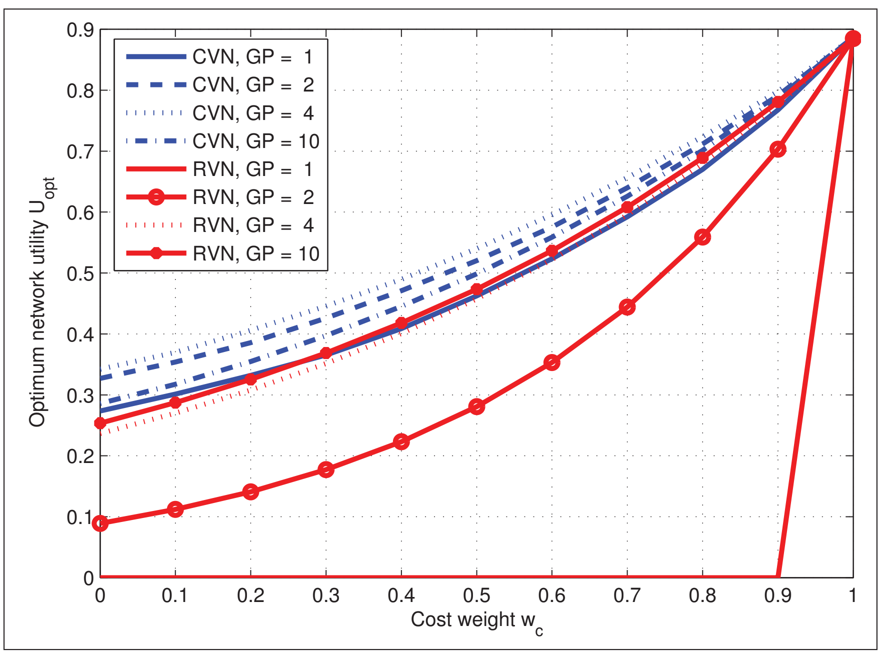


Figure 2.8 Opt. network utility U_{opt} vs. cost weight w_c for diff. GP values in ref. scenario 1 (cf. Table 2.5)

2.7.3 Optimum network utility U_{opt} of HVN for different GP values

Fig. 2.9 illustrates the optimal network utility, U_{opt}, of a HVN network for different GP values. At lower cost weights, i.e., when $w_c \leq 0.4$, U_{opt} behavior is almost independent of the GP value variation. This is due to the fact that, in this range of the w_c values, the dominant part of the HVN is composed of SBSs which do not incur any QoS degradation for RoF transmission delays, hence the invariance towards the GP value. But the interesting part of the graph is between $w_c = 0.4$ to $w_c = 0.8$, because in this design region, the HVN offers the most balanced trade-off between network cost and achievable QoS. This become more evident from the results of the following subsection.

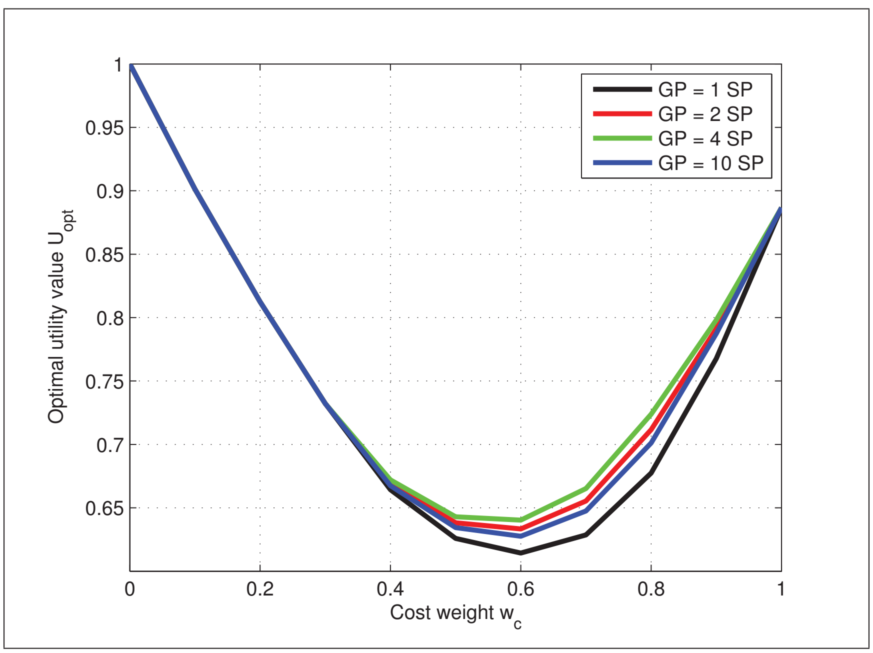
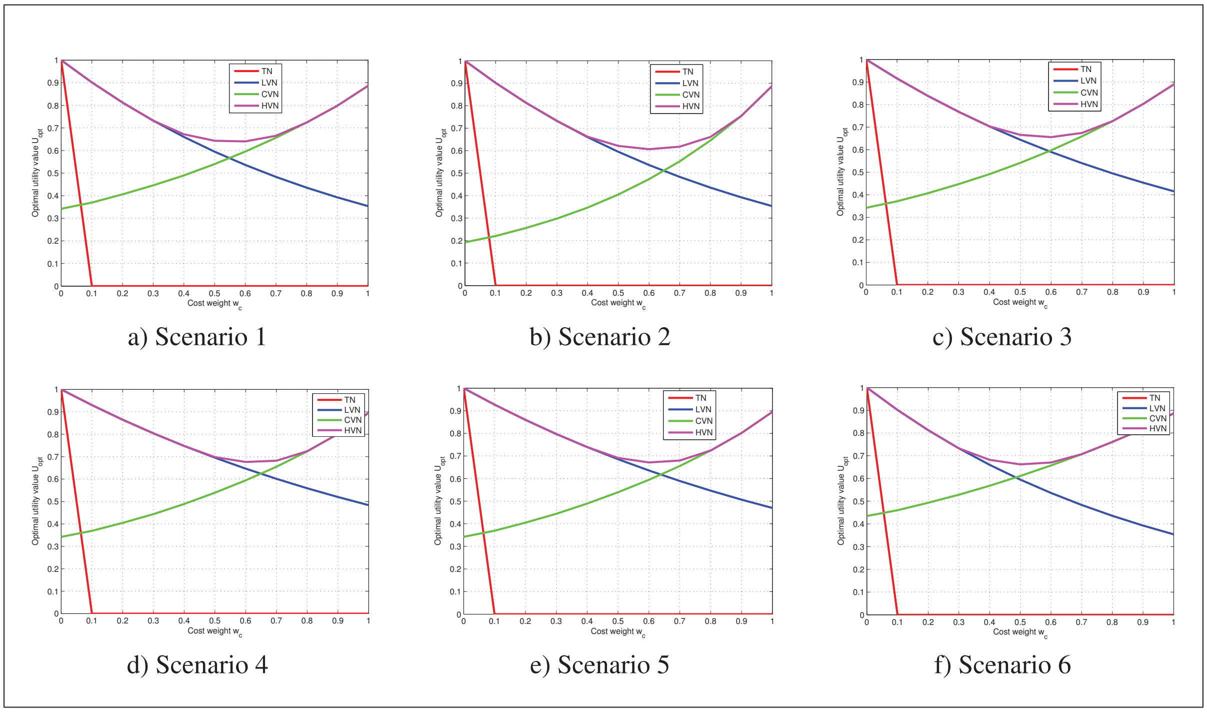


Figure 2.9 HVN optimal network utility, U_{opt}, vs. cost weight w_c for different GP values in reference scenario I (cf. Table 2.5)

2.7.4 Comparison of optimal network utility U_{opt} for different frameworks

Fig. 2.10 illustrates the network utility behavior for different frameworks and also a traditional LTE network (referred to as TN) using optimal GP values (i.e., GP_{opt} = 4 when α = 1.4 and

GP_{opt} = 5 when α = 3.0). In all the scenarios, HVN has the best utility behavior. For mid-



a) Scenario 1 b) Scenario 2 c) Scenario 3
d) Scenario 4 e) Scenario 5 f) Scenario 6

Figure 2.10 U_{opt} vs. cost weight w_c in different scenarios (cf. Table 2.5)

range values of w_c (e.g., when $w_c = 0.4 - 0.8$ in scenario 2 of Table 2.5), the HVN clearly has the best utility performance. For lower or higher w_c values, the LVN and the CVN approaches ultimately match the HVN in utility performance at either end of the w_c range, respectively, but never outperform it. Acknowledging both facts that HVN offers lower cost than the LVN at lower w_c values and higher QoS than the CVN at higher w_c values, it stands up unambiguously as the best network design choice. The value of w_c is a subjective design choice that depends on given MVON's/SP's investment constraints and intended services.

2.7.5 Optimal CVN network coefficient p_c^{opt} vs. cost weight w_c and optimal CPC radius $\bar{d}_{cpc}^{opt}$

To observe the dependence of the deployment ratio of CVN and LVN on the cost weight w_c, Fig. 2.11 shows the optimal CVN network weight coefficient p_c^{opt} within a HVN for different

w_c values. It is observed that for lower cost weights (i.e., $0 \leq w_c \leq 0.3$ (0.4 for scenario-5)), when very high QoS is required, the optimal CVN coefficient is p_c=0, which means that the whole network should be a LVN. If the offered service has lower QoS demand (e.g., file transfer, non real time applications, etc.), the SP should opt for building its network from the virtual resources of a data center (CPC). In contrast, if the offered service has strict QoS demand (e.g., voice, live video, etc.), the SP should integrate a larger share of special purpose hardware (LVN) that guarantees much faster PHY and MAC layer processing and also much lower transmission delays.

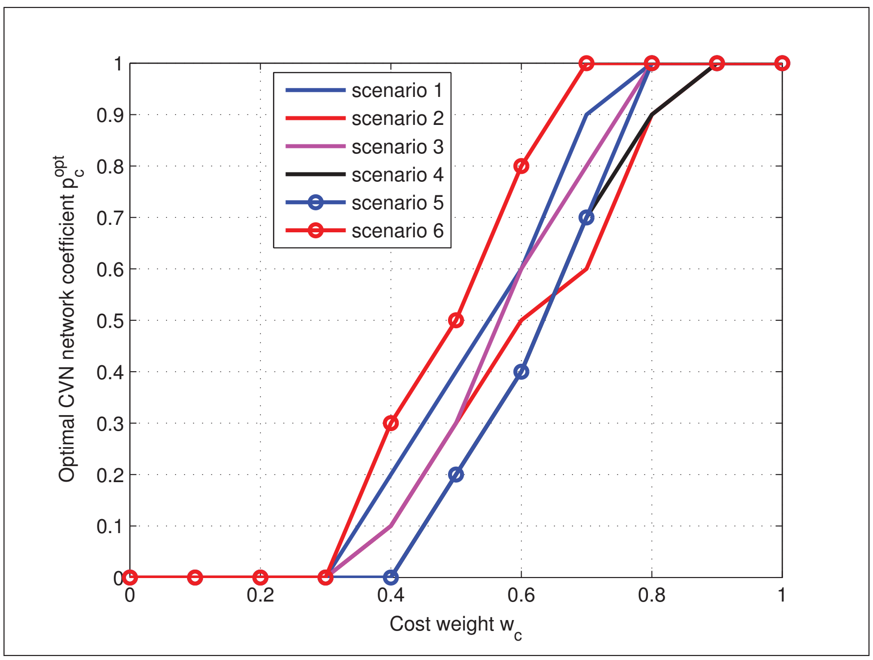


Figure 2.11 Optimal CVN network coefficient p_c vs. cost weight w_c in different scenarios (cf. Table 2.5)

To have an overall idea of the dependence of the CVN network coefficient p_c on the optimal CPC radius $\bar{d}_{cpc}^{opt}$, and cost weight, w_c, we plot in Fig. 2.12, its variation with d_{cpc}^{opt} and w_c. It is to be noted that for a low CVN coefficient (i.e., p_c=0.1), the optimal CPC radius $\bar{d}_{cpc}^{opt}$ is independent of the cost weight w_c, which is intuitive because if most of the wireless coverage is provided by distributed SBSs, a smaller wireless data center (i.e., a lower d_{cpc}) is sufficient for CVN coverage of rest of the area. But it is interesting to note that as coverage by a CVN is

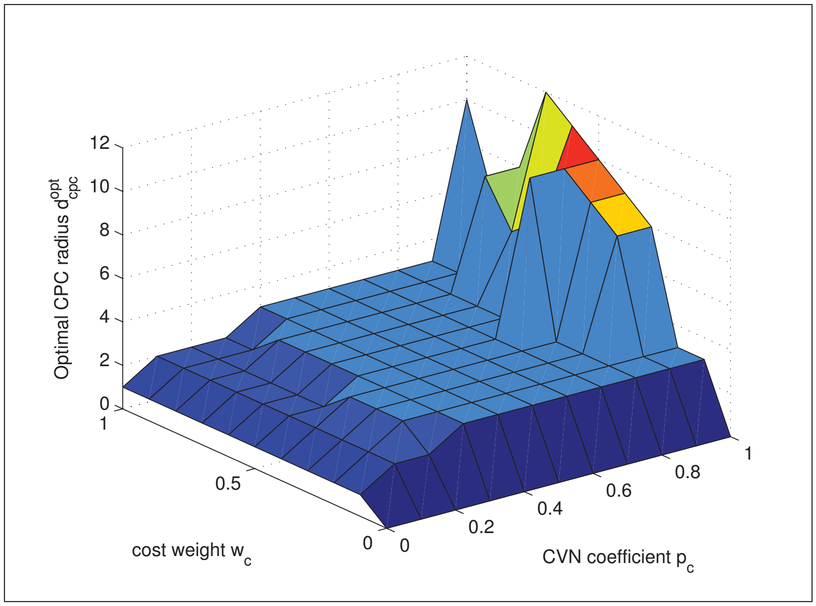


Figure 2.12 Opt. CPC radius $\bar{d}_{cpc^{opt}}$ vs. CVN network coefficient p_c vs. cost weight w_c in scenario I (cf. Table 2.5)

increased (i.e., when $0.1 \leq p_c \leq 0.7$), a CPC with radius 2 to 3 km is optimal design choice. This indicates that even if most of the wireless coverage is done through CVNs, the size of the CPCs should remain smaller. This is because of the fact that as d_{cpc} increases, the length of the fiber-optic cables that connect the RRHs to the CPCs, also increases which in turn, increases the RTTD of the signals transmitted from the CPCs to the RRHs and vice-versa. Such an increase in RTTD degrades the achievable throughput, hence the lower QoS. For this reason, a lower d_{cpc} is preferred by the utility model (cf. equation (2.44)).

2.8 Conclusions

Wireless network virtualization is considered as an important component of future 5G networks for their ability to enable efficient resource sharing and to promote network innovation by providing greater flexibility in network design. Wireless networks vary widely in terms of the services they provide and also the radio access technologies they use. For this reason, implementing a generalized virtualization architecture that enables deployment of different kinds of virtual wireless networks is a challenging issue. In this work, we have proposed three different models for wireless access network virtualization that differ in terms of their underlying physical infrastructures. The models have different set-up and operational costs; their performance also varies in terms of achievable network QoS. In the presence of multiple possible frameworks, the selection of an appropriate model for a certain scenario is a critical multidimensional challenge. In order to compare the proposed virtualization frameworks, we have built a composite multi-criteria utility model that considers both the economic and technical aspects (from PHY-MAC layer efficiencies) of the frameworks. It has been found that MAC layer parameters such as the guard period (GP) in an OFDM frame structure can be optimized from a network's cost-QoS perspective. The composite utility model presented herein provides guidance to network designers on choosing a network model that fulfils the operator's investment target and service requirement constraints. It is observed that the CVN/RVN model has a cost advantage while the LVN provides a better QoS guarantee. For a network design, neither *only network cost (i.e.,* $w_c = 1$*)* nor *only achievable QoS (i.e.,* $w_c = 0$*)* can be of concern. There must be a compromise between the two. From the analytical results presented in this chapter, it can be concluded that, the HVN can in fact, attain a balance between network cost & QoS according to a VNO's/SP's investment constraint and service provisioning goal. In order to make the analysis tractable, a rather simplified model has been assumed for network performance analysis. This model does not consider advanced PHY-MAC technologies such as coordinated multi point (CoMP), joint resource scheduling and processing among neighbouring BSs, interference management for a centralized control plane architecture, etc. In our future work, we shall include these features in the analysis of the frameworks along with the hand-off and interference management phenomena in multi-RAT HetNets.

CHAPTER 3

END-TO-END PROGRAMMABLE, CLOUD-BASED VIRTUALIZED HETNET: AN INTEGRATED ARCHITECTURE

3.1 Introduction

From the network cost and QoS analysis of Chapter 2 it is evident that data-center based network (i.e. the CVN/RVN) will be an integral part of future virtualized wireless infrastructure. We argue that SDN and cloud computing are the key enabling technologies for implementing the future virtualized network architecture. In this chapter we lay out an integrated architectural framework for an end-to-end programmable, cloud-based virtualized HetNet.

Wireless network ecosystem consists of different types of networks that differ from the services they provide, their key performance indicator (KPI) requirements (e.g., throughput, delay, bandwidth need, etc.), also the type of network nodes used and their power requirements. For example, wireless sensor networks (WSNs) are application specific, they use low power nodes that communicates sporadically with bursty traffic. Power efficient operation of the WSNs is the main concern due to their limited battery capacity. WiFi networks operate in the unlicensed spectrum band and normally the applications (e.g., web browsing, file transfer, etc.) used in these networks are not very delay sensitive but might result in high amount of data traffic. On the other hand, cellular networks operate on licensed spectrum and they provide guaranteed service for various delay-sensitive applications (e.g., voice, live video, etc.). Hence they have stringent requirements on network delay and throughput. One of the problems with current network implementation is that the networks are service specific, hence, network nodes are tailored to serve a specific type of service.

This service specific network deployment has both higher capital and operational expenditures (CAPEX & OPEX). The astounding cost of network operation also impedes new business players to enter the market. The vision of future networks (e.g., 5G and beyond) demands the presence of various network operators that would provide differentiated services to users in heterogeneous wireless network environments. In this respect, it is necessary to have a service agnostic deployment of physical resources that can be used economically and flexibly in an on-demand basis for provisioning various wireless services. We argue that to realize such a network architecture, a virtualized network infrastructure is required to be composed of programmable and flexible network resources which will be shared by different virtual network operators (VNOs). The physical resources (including the licensed wireless spectrum) will be provided by one or more infrastructure providers (InPs) who will deploy, manage and lease the physical and virtual resources to the VNOs. Also, to alleviate the problem of vendor specific inflexible nodes that are not amenable for adopting new wireless networking technologies, software defined networking (SDN) technologies should be incorporated in implementing the network infrastructure. And on-demand, elastic resource provisioning should be ensured following a cloud-based network deployment model. Wireless network virtualization (WNV) can be achieved in different ways. But we deem SDN as a significant enabler for successful realization of WNV.

SDN (Bosshart *et al.*, 2013), (Kobayashi *et al.*, 2013) is able to abstract physical resources for its ability to separate the network control plane from the data plane. Thus it can provide absolute control over the network substrate in the form of programming it. The flexibility provided by SDN is instrumental in providing novel services that require change of device functionalities to provide differential services (Rahman *et al.*, 2015b). In addition to provide flexibility in managing the network infrastructure, SDN can also reduce network cost (both CAPEX and OPEX) by replacing expensive network nodes with off-the-shelf (OTS) cheaper programmable data plane equipment and centralizing controller in IT servers.

Hence, we argue that for realizing a programmable & flexible heterogeneous virtual network infrastructure, SDN & cloud computing technologies are the key tools to leverage. In such a network infrastructure, VNOs will be able to offer their differentiated services in their target networks (e.g., WSN, cellular or WiFi) leasing virtual resources from one or more InPs. In this chapter, we identify the key requirements of such a heterogeneous virtual wireless network (HVWN) infrastructure; then we discuss different components for an end-to-end solution for a programmable, elastic HVWN following a top-down approach. For realizing an end-to-end programmable HVWN, programmability in the radio plane, network plane and on-demand elastic resource provisioning are very important. The existing wireless network architectures proposals that leverage SDN, cloud computing and programmable radio plane have been presented in Chapter 1. As it is seen, these proposals try to provide solution for a particular radio access technology (RAT) (e.g., WiFi, cellular, WSN, etc.) or a particular part of a network (e.g., cellular core vs access networks). But an integral solution for programmable, elastic, virtuilized heterogeneous networks is not available in the open literature. In this chapter, we lay out the architectural blueprint of and end-to-end programmable HVWN where different VNOs can implement their customized networks using a RAT that best suites there services. We also identify the critical business cases and open problems & challenges in realizing such a network architecture.

The rest of the chapter is organized as follows: in Section 3.2, key requirements of HVWN are identified. Section 3.3 describes different layers for the end-to-end programmable HVWN solution. Business cases for HVWN is discussed in Section 3.4 and in Section 3.5, potential research issues and challenges have been identified. Finally, we conclude the chapter in Section 3.6.

3.2 Requirements of Programmable Virtual Wireless Networks

A virtualized wireless network must satisfy certain requirements, some of these critical requirements are discussed in this section.

3.2.1 Virtual Network (VN) Isolation

The VNs sharing a common physical infrastructure should be perfectly isolated from each other, so that, to a VNO, it will appear that such a VNO has the sole-ownership of the (virtual) network. Operation of a VN should in no way affect the other VNs sharing the same physical resource, e.g., for two VNs sharing a common physical node, if load increases in one VN, traffic belonging to the other VN should not suffer from additional delay in processing, queuing or reduction in throughput. Service level agreements (SLAs) between the VNOs and the InPs should always be fulfilled. SLA is basically a subset of key performance indicators (KPIs) which might comprise of minimum guaranteed processing power, memory space, bandwidth/throughput, maximum downtime of system, etc.

3.2.2 End-to-end Programmability

VNOs should have complete flexibility over the virtual/physical resources they lease from one or more InPs. This flexibility is demonstrated through their ability to modify (program) the underlying resources in a way that best supports their intended service requirements. For example, in a virtual LTE network implementation, a VNO should be able to program the core network switching fabric to route its core network packets through the optimum routing graph consisting of mobility management entity (MME), switching gateway (S-GW), packet data network gateway (P-GW), policy and charging rules function (PCRF), etc., nodes. Similarly, for the radio access plane, a VNO might require a customized radio processing chain to process its baseband signal. Hence, provisions should be made so that, it can assemble various processing blocks (e.g., for frequency transformation, modulation, coding, etc.) in a programmatic manner (Bansal *et al.*, 2012). Also, a VNO should be able to implement its custom protocol stack to optimize its intended service performance; hence, programmatic control over the protocol layers is also necessary.

3.2.3 On demand resource provisioning

In the VNO-InP business model, a VNO would request its required resources (virtual/physical) for the InP. Upon availability of the resources, the InP would assign the requested resources to the VNO, forming a service level agreement (SLA) between the InP and the VNO. During its operation, if the VNO needs additional resources (e.g., computing, storage, radio spectrum, etc.) it would request the InP for the lease of these additional resources. InPs should be able to cater for such on demand elastic resource provisioning. This is where the cloud computing model comes into play in a virtual wireless network ecosystem. In this model, an InP can be seen a as cloud service provider that is composed of a geographically distributed *cloud* of resources. A VNO receives the lease of its requested resources from the InP without the necessity of being aware of the physical location of the resources.

3.2.4 Network Function Virtualization (NFV)

One of the major motivations behind network virtualization is to reduce the CAPEX and OPEX of network provisioning, so that operators can cope with the increasing network cost and also, so that new players can get affordable entry to the market. To address this issue, major telecom operators and vendors are opting for network function virtualization (NFV) (nfv, 2013). The main idea behind NFV is to separate network hardware from the software that runs on it, as this will pave the way to implement different network functionalities as software instances in a general IT platform. This paradigm shift in network architecture will replace the traditional special-built network nodes which are not only expensive but also very power hungry.

3.2.5 Dynamic Spectrum Sharing

Scarcity of the licensed spectrum is the *Achilles' heel* for next generation wireless networks. Despite all the advances made in network architectures, baseband processing, error correction channel coding, etc., limited licensed spectrum remains the major bottle neck for telecommunication networks. To alleviate this problem, efficient utilization of the radio spectrum in time (time division multiple access (TDMA)), frequency (frequency division multiple access (FDMA)), space (space division multiple access (SDMA)) is necessary (Niebert *et al.*, 2008). Especially in the virtual wireless networks environment, dynamically sharing the spectrum among the incumbent VNs while respecting the SLA is of utmost importance. Opportunistic sharing of the licensed spectrum in combination with utilizing the unlicensed spectrum band wherever possible might mitigate the spectrum scarcity problem to a great extent. Also, the use of millimetre (mm) wave for future 5G networks is gaining momentum as many researchers from industry and academia are strongly advocating in its favor (Zhao *et al.*, 2013), (Rappaport *et al.*, 2013). Due to its ability to provide higher throughput for lower transmission distance mm wave is an ideal transmission candidate for small cells (pi and Khan, 2011).

3.3 End-to-end programmable, elastic, HVWN

In a heterogeneous wireless network ecosystem, we observe different kinds of network deployments targeted for specific purposes. For example, wireless sensor networks (WSNs), adhoc networks, WiFi networks, cellular networks, etc. These networks have a varied range of performance requirements which translate to varied levels of spectrum (licensed or unlicensed) requirements, signal processing demand, wireless transport mechanism, security provisioning, billing mechanism, etc. In a virtual wireless network environment, VNOs will provide different kinds of services targeting various commercial applications that will require them to deploy one or more of the above mentioned network types. For this reason, it is imperative to have an end-to-end solution for provisioning programmable, elastic, heterogeneous virtual wireless networks, so that, VNOs can build their own customized network leasing the required resources from one or more InPs. Fig. 3.1 shows a typical scenario of heterogeneous virtual wireless networks. Here, the physical infrastructure deployed by InPs consists of wireless data centers (WDC) that are interconnected via a metropolitan optical network (MON), geographically distributed virtualized base stations (BSs), WSN, WiFi hotspots, home networks, etc. The WDC houses storage and computing resources (e.g., blade servers) as well as a programmable networking fabric (e.g., software-defined switches) for implementing network functions as software instances for different network types. Some of the servers contain network controllers for flow-based virtualization (Sherwood *et al.*, 2009) implementation. The controllers dynamically program the underlying programmable switching fabric (we use the generic term 'switch' to refer to programmable switches, routers and other middle boxes) as dictated by a specific VNO. A mobile virtual network operator (MVNO) providing LTE service can implement its core network components, e.g., MME, PGW, SGW, PCRF, etc., as software instances in WDC.

For baseband signal processing, the MVNO can process signals in software instances of BSs implemented in the WDC and transport the processed signals from the WDC to remote radio heads (RRHs) via optical fiber front haul. But the additional delay incurred in centralized processing in WDCs might not meet the QoS requirements of certain delay-sensitive applications (e.g., voice, video conferencing, etc.). In such a case, processing of such traffic should be done in distributed virtualized BSs that are distributed in the coverage area of the MVNO (Rahman *et al.*, 2015d).

A WiFi VNO implements the applications (e.g., authentication, authorization, accounting, mobility and interference management) necessary to run its end-to-end operation in WDC and programs the programmable WiFi nodes leased from the InP via its controller platform which translates the applications requirements to instructions recognized by the nodes. The nodes can be connected to the WDC either via optical fiber or microwave links depending on the available logistics. Also a MVNO can share all or part of the WiFi nodes' slices to increment its coverage in the area and also for offloading traffic. The virtualizer in the WDC is responsible for managing the isolation between the VNOs sharing the same physical nodes.

For virtual sensor networks, instead of deploying application specific sensor nodes, an InP deploys programmable generic sensor nodes that is capable of sensing various environmental aspects (e.g., temperature, humidity, wind speed, etc.). VNOs providing sensor network services lease slices of the sensor nodes to collect their intended environment data. This data can be forwarded to a WDC through a slice of a BS (cf. Fig. 3.1) deployed in the sensor network vicinity. Virtualizer in the WDC forwards the data to the appropriate VNOs (applications) where the data is processed to extract the desired information.

Over the top (OTT) service providers, for example, IPTV, online gaming providers, can lease processing and storage resources form the InPs and implement various processing blocks (optimized to for the intended services) as software instances in WDC and provide their services to the end users. Shared access of physical nodes (in WDC and customer peripheral equipment

(CPE)) and wireless spectrum can be administered by incorporating local controllers in the CPEs in addition to the global controller at the WDC.

A layered representation of HVWN is given in Fig. 3.2. In this section, we briefly discuss the various layer of the HVWN model.

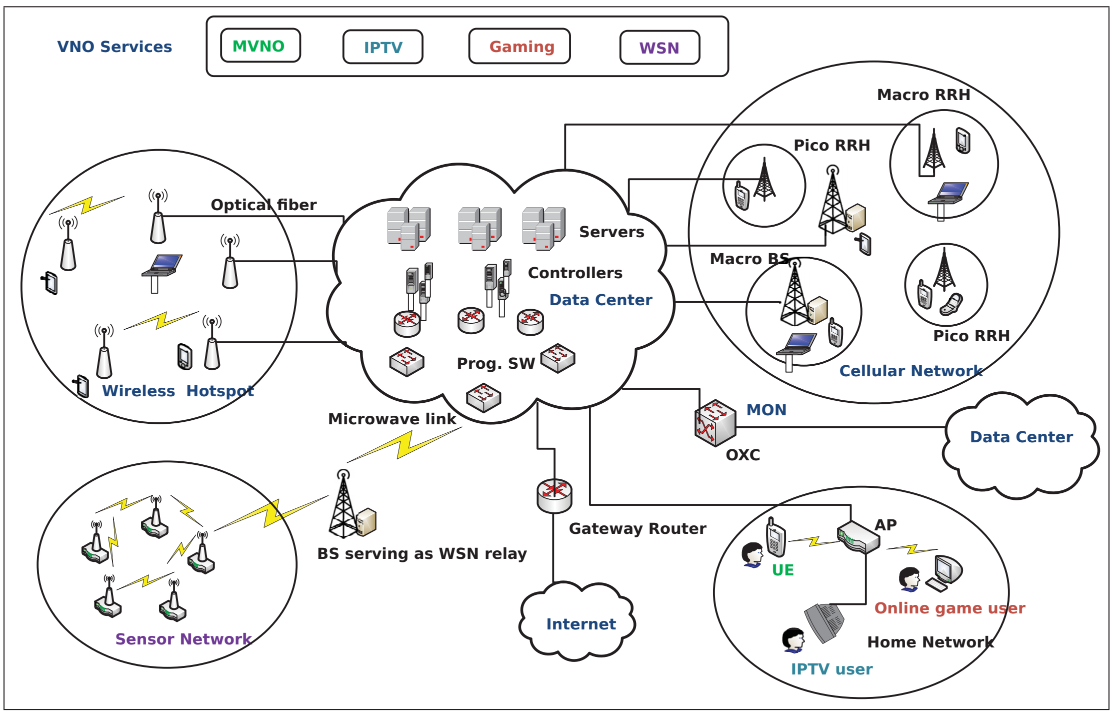


Figure 3.1 Heterogeneous Virtual Wireless Networks Scenario

3.3.1 Management and Orchestration layer

The management and orchestration layer manages the resources and VNFs of the virtualized platform. It consists of physical and virtual resource managers that control the physical and virtual resource provisioning, admission control of new VNOs, etc. The VNF manager is responsible for the instantiation, management and life cycle management of the VNFs. The spectrum manager is responsible for shared (virtualized) access of the radio spectrum among different VNOs. It can provide either static or dynamic spectrum sharing among the incumbent

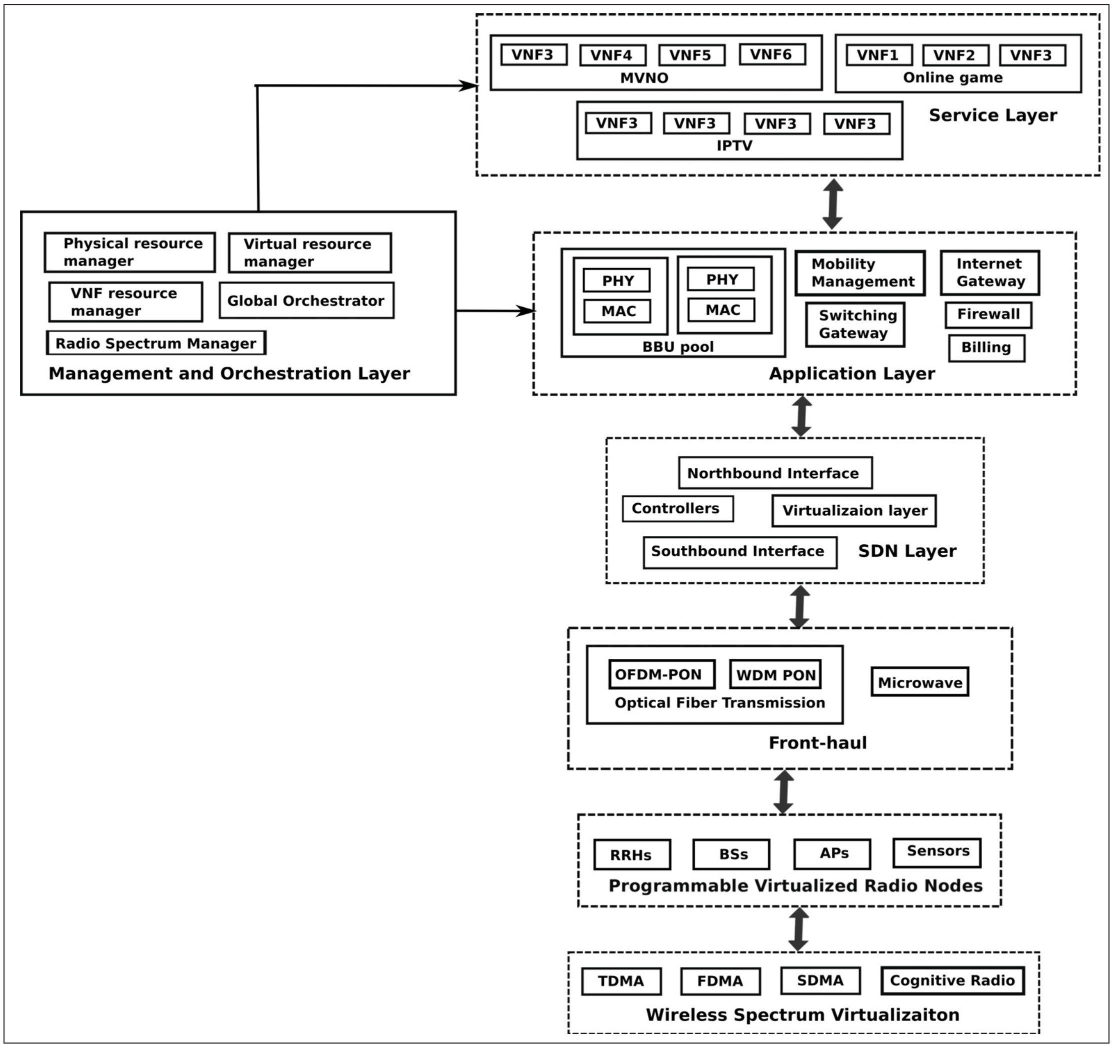


Figure 3.2 Functional blocks of end-to-end programmable heterogeneous virtualized wireless networks

VNOs. The global orchestrator orchestrates the overall operation of an InP's platform. Fig. 3.3 gives a flow representation of different steps followed by the management and orchestration layer during a VNO request for setting up a virtual network.

3.3.2 Service layer

Services provided by VNOs can be very different from each other. For example, a VNO can be either a mobile virtual network operator (MVNO), an IPTV provider, an online game provider,

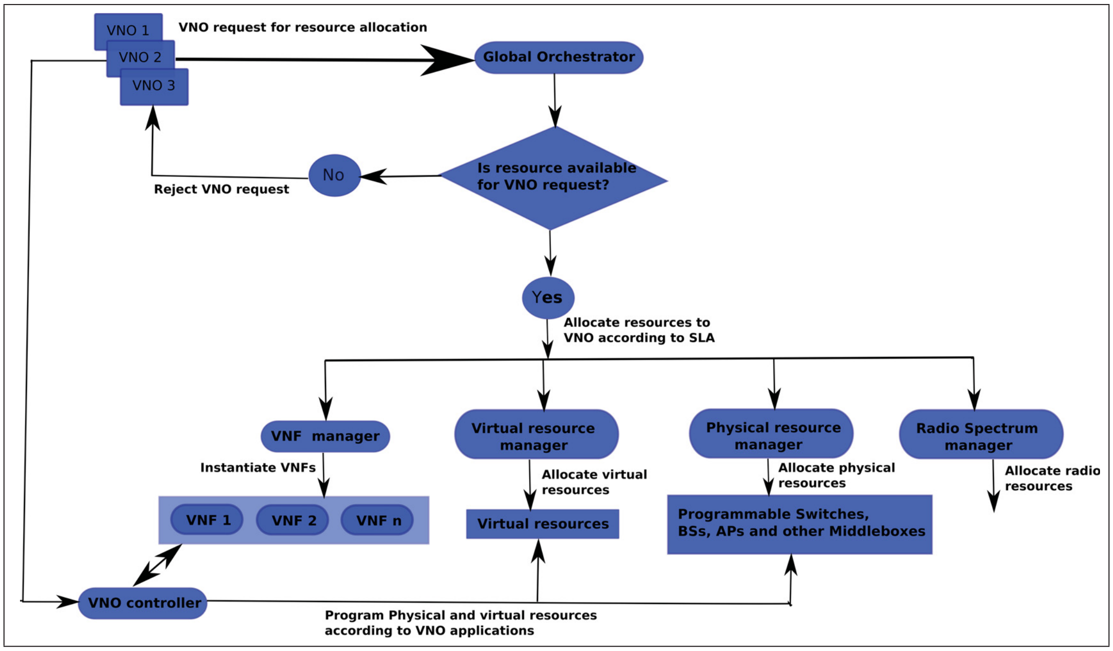


Figure 3.3 Flow diagram of operational steps of the management and orchestrator layer during a VNO request for setting up a VN

etc. The service layer mainly expresses the services of different VNOs as forwarding graph of different virtual network functions (VNFs). Depending on the service requirement of a particular VNO, a VNF in a VNF graph can be unique or it can be a common component shared with other VNOs (cf. Fig. 4.2).

3.3.3 Application layer

This layer consists of different network applications that perform various network operations. For example, routing of traffic in the (virtual) network, managing the mobility of the users, blocking malicious traffic, etc. These applications are in fact, the virtual network functions (VNFs) that performs different network functions. These applications instruct the controller layer, which in turn programs the underlying switching fabric to implement the application functionalities. For VNs that require performing baseband operation in the WDCs, software instances of the network nodes (e.g., BSs, APs, etc.) and baseband unit (BBU) pools are also implemented in this layer.

3.3.4 SDN layer

To enable network programmability, separation of control and data planes is necessary. In a SDN paradigm, network intelligence is centralized in network controllers that programmatically modify the forwarding behavior of the underlying data plane devices. The main components of the SDN layer are the virtualization and the controller sub-layers.

Virtualization layer

A virtualization layer creates an abstraction of the underlying physical infrastructure. It also enforces isolation among the VNs that share common physical resources, such that, to the VNOs it appears that they own the data plane of their individual networks. The isolation or separation enforced by a virtualization entity, i.e., a hypervisor can be either physical or logical. In physical isolation, which is also known as *hard slicing*, dedicated physical resources are provided to the VNOs. Whereas in logical isolation, known as *soft slicing*, instead of dedicated physical resources, a VNO is provided access to resources that are shared with other VNOs while respecting the SLA (satisfying agreed upon KPIs) between the InP and the VNOs. While hard slicing provides a dedicated resource guarantee, it also can result in inefficient resource utilization. On the other hand, with soft slicing, efficient resource utilization can be ensured with minimum guarantee on resource availability for each VNO. One of the most popular modes of achieving soft slicing is flow-based virtualization approach (Sherwood *et al.*, 2009), which is basically bundling the flows from individual VNOs and isolating the bundles from each other.

The controller platform

A controller is a centralized control plane intelligence for a VNO. It has the global view of the virtual network and it operates on the behest of the applications residing at the upper layer. A northbound interface/API (Reich *et al.*, 2013) can facilitate the synchronous operation of multiple applications at the upper layer by arbitrating conflicting operation of multiple applications, that try to implement rules that might conflict with each other trying to modify a certain data plane device at the same time. A southbound interface/API is a control channel protocol. It communicates the instructions from the controller to the underlying switching fabric. For example, in an OpenFlow (McKeown *et al.*, 2008) based SDN platform, a controller modifies the forwarding behavior of the underlying switches by populating the forwarding tables of the switches with match-action based rules to dictate the forwarding of the packets traversing the switches.

3.3.5 Baseband signal processing

Two different varieties of baseband signal processing are possible. One is *in-situ processing* which is similar to the processing mechanism of traditional BSs (R.Kokku *et al.*, 2012a), i.e. the signal is processed in the physical BSs that have been virtualized (sliced) into multiple virtual base stations (VBSs). The other is to push the baseband signal processing to a centralized location that contains baseband unit (BBU) pools (Lin *et al.*, 2011); this is in fact, pushing the baseband processing to the cloud. The two different models have their relative pros and cons. The *in-situ processing* will provide faster signal processing and transmission latency will be very low but as the VBSs run on special purpose hardware, the VNOs will have less flexibility in curtailing the processing characteristics of the VBSs to better fit their service requirements. Moreover, this type of implementation will be more expensive for their use of special purpose (e.g., FPGA-based) hardware. On the other hand, in cloud-based baseband processing in BBU pools, the VNOs will enjoy more flexibility and control over the baseband processing chain. In this model, BSs are implemented as software instances in IT servers; hence modifications to the processing chain are merely including software patches. Scaling (up/down) of resources is very convenient in this model because if any VNO need additional VBS instances, it can request the InP to allocate more VBS instances and in an IT-based platform, it is very convenient to elastically scale the resources. But the downside of this implementation is, the software VBSs have to full-fill the real time processing need of wireless networks which is quite significant. Moreover, carrying the processed signal from the BBU pools to the radio end introduces additional latency which might deteriorate performance of delay-sensitive applications like voice, live video, etc. Hence, a VNO has to lease resources according to the requirements of its intended service provisioning.

3.3.6 High capacity front-haul

Data traffic in wireless networks is increasing in an exponential manner due to video traffic domination. For this reason, in a cloud-based network implementation, a high capacity front-haul is necessary to carry traffic from the data-center to the RRHs. Being a high capacity traffic conduit, fiber optic cables are capable of carrying very high amount of traffic with very low latency, thus making them an obvious solution for high capacity front-haul. Different passive optical network (PON) solutions for fiber optic communication are available in the market, for example, wavelength division multiple access PON (WDM-PON), orthogonal frequency division multiple access PON (OFDM-PON), etc. But due to geographical and logistic limitation, it might not be possible to use fiber optic front-hauls in some places. For those locations, high capacity microwave links should be used to carry traffic to and from the data-centers to the RRHs.

3.3.7 Programmable virtualized radio nodes

Depending on the deployment modalities radio access can be provided either by remote radio heads (RRHs) that are composed of simple radio transceiver and antennas (for the C-RAN model (cmr, 2011), (Lin *et al.*, 2011)) or as part of virtualized base stations (R.Kokku *et al.*, 2012a) (Bhanage *et al.*, 2010b), programmable WiFi APs and sensor nodes. For the RRH deployment, sharing of the nodes by different VNOs can be facilitated by making the RRHs full-duplex (FD) capable (Hong *et al.*, 2012), so that, different VNOs can share the antenna at the same time either in uplink (UL) or downlink (DL) direction by scheduling them in time domain. Furthermore, fine-grained control over the PHY layer processing blocks can be achieved by using a programmable radio (Bansal *et al.*, 2012) based front-end. But implementing distributed RRHs for radio access demands high capacity front-haul links. Optical fiber is the obvious choice for such front-haul links due to their very high capacity and ultra low transmission delay. But deploying fiber-based front-haul might not be possible in every geographical scenario. In such cases, network coverage should be provided by in-situ physical BSs that have been sliced (virtualized) to be share among multiple VNOs (cf. Fig. 3.1). These BSs will be connected to the WDC via high capacity microwave links.

3.3.8 Wireless spectrum virtualization

Radio spectrum is the bottle neck for wireless networks, especially those operating in licensed spectrum band. Hence, licensed spectrum should be virtualized, so that, different VNOs can synchronously share them in time (TDMA), frequency (FDMA) and space (SDMA). The spectrum manager in the management and orchestration layer is responsible for managing the sharing of the licensed spectrum among the incumbent VNOs. To mitigate the spectrum scarcity problem, unlicensed spectrum should also be used in opportunistic manner wherever possible. Leveraging cognitive radio (CR) (Mitola and Maguire, 1999) technologies, VNOs can opportunistically share a free frequency band that is not currently being utilized by the primary users. Discussion of CR technology is out of the scope of this thesis; interested readers can read the afore mentioned paper and the references within. Opportunistic spectrum use can be administered by the Local controllers in the transmission nodes which have access to spectrum availability information in the area where the node is operating.

3.4 Business Cases for Programmable Heterogeneous Virtual Wireless Networks

The virtual wireless network paradigm in a HVWN can bring drastic changes to the business eco-system of heterogeneous wireless networks. A flatter and simplified data-plane with centralized programmable control plane architecture has the potential to introduce novel business dynamics in this area. There will be significant role change among equipment manufacturers (i.e., vendor companies), network operators and third party software providers. In this section, the roles of different players in a HVWN environment is briefly discussed.

3.4.1 Equipment manufacturers

The manufacturers of network nodes will produce simplified data plane devices which can be programmed through high-level abstraction. Unlike present day, complicated hardware, new data plane devices will have open APIs through which high level programs will be able to modify their forwarding behaviors. The devices can range from simple match-action based forwarding gear to more sophisticated APs, routers and middle boxes, capable of doing deep packet inspection (DPI). Standardized southbound API (e.g., OpenFlow (McKeown *et al.*, 2008)) support should be provided by all vendors. This will get rid of the closed, ossified construction of present day network equipments and operators will be able to easily integrate components from different vendor companies.

3.4.2 Infrastructure providers (InPs)

Physical infrastructure of the network is established and maintained by the InPs. The physical infrastructure includes the computing, storage and networking resources, as well as the radio access nodes and backhaul links. They are also responsible for creating virtual network resources by *slicing* the physical resources. These virtual resources in turn are leased by the VNOs to roll-out their own (virtual) network. From cloud computing perspective, the InPs can be seen as infrastructure as a service (IaaS) providers. An InP can also own licensed spectrum which it will share among the VNOs it is hosting. In addition, an InP can also function as a VNO to provision any specific service. To enable the cloud tenants (i.e., the VNOs) to implement their customized network applications, InPs should support standardized northbound APIs (Reich *et al.*, 2013) that will open-up a flexible, high-level programming abstraction of the underlying virtual resources, so that, the network programmers can write sophisticated applications without having to be aware of the physical resources. Currently, there is no standardized northbound API but efforts (onf, 2013) are being put forth in this direction.

3.4.3 Mobile Virtual Network Operators (MVNOs)/ Service Providers (SPs)

MVNOs/SPs lease virtual network resources from the InPs and build their own customized networks. Using the flexible abstraction of the physical resources, the VNOs (MVNOs/SPs) can implement their network with customized network protocols tuned for optimal performance of their intended services. VNOs can offer a variety of services using the IaaS platform of the InPs. For example, MVNOs can implement their services using customized mobility management, policy enforcement and charging policies. Their virtual existence is transparent to the general users as there is no change required on the UEs and service offering model is similar to that of a physical network operator today.

Over the top (OTT) service providers like YouTube, Netflix, online game providers etc., can lease resources (e.g., wireless spectrum) from InPs to ensure a minimum QoS (rather than Internet's best effort services) of their services, so that user satisfaction can be ensured which is very critical for their continued business success. As service differentiation is one of the major benefits of software-defined HVWN, it is technically very convenient for the InPs to ensure QoS for such OTT SPs. By having their own virtual network, SPs can implement their customized processing (e.g., transcoding for HD video, accelerated streaming protocols, etc.) functions to enhance the quality of their services.

3.4.4 Third-party software companies

Different network services are implemented as software packages in HVWN. Third party software companies, specialized on network applications can produce different applications as per requirement of the InPs and VNOs. In this model, software packages can be provisioned as *managed services* by the software companies. This will relieve the network operators of having a dedicated software department for implementing new services and making patches for the existing services. The advantage of such a business model is two fold: on one hand, the VNOs can save operational expenditure by not maintaining a software team and on the other hand, new software companies can flourish that are specialized in developing network applications.

3.4.5 Inter ISP-VNO traffic offloading

As mentioned earlier, mobile users' traffic is dominated by high volume video applications. It has been observed that, users tend to be static (Lee *et al.*, 2013) while using high data rate applications like: HD movie, online gaming, etc. Most often the users are under the coverage of a WiFi hotspot, typically run by an internet service provider (ISP). To reduce the strain on the licensed spectrum of a VNO, the high data rate traffic of static users can be offloaded to the ISP's WiFi. The ISP can charge the VNO according to their service level agreement (SLA). Depending on the SLA, an ISP can have various access and charging policies for different VNOs.

3.5 Potential Research Issues & Challenges

HVWN is a very promising network architecture that make possible building end-to-end programmable, on-demand virtual networks for a range of wireless network environments. But a number of challenges need to be addressed for a successful realization of HVWN. Some of these challenges are briefly discussed in this section.

3.5.1 Standardization of APIs

For a successful, well accepted design solution, it is important to ensure the interoperability of applications and equipments developed by different players i.e., equipment vendors, operators and third party software companies. To realize this, standardization of different interfaces (i.e., northbound, southbound, east-westbound) is of utmost importance. OpenFlow (McKeown *et al.*, 2008) is already a well accepted standardized southbound API (maintained by ONF (onf, a)) for interfacing the network nodes with the controller layer. Many commercial products by different vendor companies are available in the market that use OpenFlow as a southbound API. Standardization of northbound API (Reich *et al.*, 2013), (Foster *et al.*, 2011) is also necessary, so that, network programmers can build modular, reusable applications (Monsanto *et al.*, 2013) without worrying about the underlying physical hardware and control platform. ONF has already started a working group (onf, 2013) to standardize the northbound API. Similarly, for control platform interoperability, a standard east-westbound API is also needed which should be agreed upon by all the parties (e.g., vendors, operators) involved.

3.5.2 Balance between flexibility and complexity

Different levels of abstraction are possible for building a software-defined virtual wireless network. A FlowVisor (Sherwood *et al.*, 2009) abstracts the physical network in flow-level granularity, so it is convenient to build flow-based virtual wireless networks using this hypervisor model. But this hypervisor model does not give any control over the processing modality of the devices, hence, it is not possible to modify the PHY, MAC layer processing chains of devices using this model. On the other hand, OpenRadio (Bansal *et al.*, 2012) provides a rich platform to compose novel wireless protocol stack by separating the protocol from the hardware. It supports different wireless protocols (e.g., WiFi, 4G) on a common hardware platform and enables a programmer to select processing blocks at the PHY layer. But the platform does not provision for upper layer management, for example, building complex, modular network applications is not possible through this architecture. Hence, during network build-up, a compromise has to be made between the level of flexibility and the depth of control that the platform will provide for building sophisticated, efficient programmable virtual wireless networks on top of a common physical substrate.

3.5.3 Security threats minimization

While SDN allows the creation of network applications to provide a secured wireless network, it has its fair share of security pitfalls. Interestingly, the unique characteristics of a SDN paradigm, i.e., the separation of control & data planes and network programmability opens up doors for various security threats. Kreutz et al. (Kreutz *et al.*, 2013) identified couple of threat vectors for software defined networks, e.g., faked traffic flows, switch vulnerability, compromise of controller and control plane communications, etc. Interestingly, these are all specific to the software defined networking paradigm. To alleviate the afore mentioned security threats, some precautionary measures are also proposed in (Kreutz *et al.*, 2013), for example, replication of network controller and applications, so that, a back up node can take charge in case the acting one fails. There should be diversity among the controllers and they should be provisioned for auto-healing mechanism to recover from a security attack. Security measures to protect the network should be an integral part of the network design process from the very early phase.

3.5.4 Virtualization of wireless spectrum

Due to stochastic nature of radio propagation environment, abstraction of wireless spectrum is very challenging. Virtualization of wireless spectrum can be achieved by static or dynamic allocation of spectrum among the incumbent VNOs. While static allocation provides guaranteed spectrum allocation, it might result in inefficient spectrum utilization. On the other hand, dynamic spectrum allocation that ensures fairness requires efficient scheduling algorithms. Opportunistic spectrum sharing in non-contiguous frequency band along with the use of non licensed band can be beneficial for the VNOs.

3.5.5 Definition of isolation

Virtualization bring about the contradictory concept of maintaining isolation among VNOs while they share common resources. Hence the definition of slicing need to be agreed upon among the concerned parties (e.g., vendors, operators, third parties, etc.). This brings forth the discussion on *hard slicing* vs *soft slicing*. Hard slicing refers to dedicated resources allocated to a VNO and soft slicing means that there will not be any dedicated resource allocation per say but the VNOs will have guarantee for minimum KPIs through the agreed upon SLA. While hard slicing will ensure perfect isolation and higher customer satisfaction by providing higher QoE, it will result inefficient resource utilization. On the other hand, soft slicing will increase resource utilization but it might also impact the performance (in the form of achieved QoE) of the VNOs.

3.5.6 Integration of Cognitive Radio (CR)

Spectrum scarcity is a pressing problem in wireless networks especially for cellular operators. Cognitive Radio (CR) technology (Mitola and Maguire, 1999) can go a long way to minimize the spectrum scarcity problem by enabling the use of idle spectrum by other users. The main idea behind CR is, to allow secondary users to utilize unused (idle) radio spectrum belonging to primary users. A CR senses its surrounding environment and adaptively tunes its transmission parameters to transmit data while maintaining required QoS. For its ability to *virtually* expand the radio spectrum, it has been proposed (Ahokangas *et al.*, 2013), (Xiao *et al.*, 2013) to be used in existing networks.

3.5.7 Backward compatibility

Using SDN and Cloud computing for wireless networks is a fairly recent trend in wireless research. The ultimate goal is to replace the distributed traditional networks with logically centralized control platform. But for its gradual inclusion to the existing network infrastructure, it is very important to ensure its smooth inter-operation with the existing wireless network infrastructures. Various *buffer devices/applications* can be placed at the interface of the two modes of networks that will complete the necessary translation of control and data plane information. Keeping the processing delay in such buffer devices/applications to a bare minimum will be a challenging issue.

3.6 Conclusions

In future heterogeneous wireless network scenarios, different types of service providers will provide services in various target network environments. A cost-effective network solution for the services providers can be offered through a virtualized infrastructure. In this chapter, we have brought forth the convergence of virtualized heterogeneous wireless network infrastructure that facilitates abstraction of physical resources, hence paving the way for their efficient utilization.

The two key requirements that the future service providers will need are programmability and elasticity of their networks which will provide them enough flexibility & control over the network substrate and make them able to scale up/down their network resources to meet customer demands. In this respect, we have presented an end-to-end programmable, cloud-based solution for heterogeneous wireless networks named HVWN. It provides programmability in both network core and access by employing SDN and programmable radio technologies. The current virtualization solution in the open literature propose to virtualize a specific network type (e.g., cellular (core or access) or WiFi or WSN). Whereas the HVWN provides a virtualization solution for a heterogeneous array of networks that works with different RATs, different data transmission modes (e.g., bursty vs. continuous transmissions of WSN and cellular networks, respectively) and various QoS requirements on a common subset of physical hardware. In HVWN, at the core, different networks are implemented as individual services which are a connected graph of VNFs. These VNFs are software instances that are agnostic of underlying hardware platform. For traffic routing and radio access provisioning, programmable nodes are used that can be shared, easily upgraded through open programming APIs. To meet the service requirements of different kinds of networks, HVWN uses cloud-based resource pools in distributed WDC as well as virtualized APs that use general purpose hardware and in-situ signal processing. VNOs can lease appropriate resources from the InPs to deploy their customized virtual networks. Business cases for virtual wireless networks have also been discussed. Finally, we explored the critical research issues and challenges to resolve in implementing programmable virtualized heterogeneous networks.

To sum up, virtualization of heterogeneous wireless networks is very significant tool to combat different logistical problems of current network deployments as well as to cater for future network demand. But a broad range of research issues and challenges need to be tackled. In this chapter, we have presented the current technologies that are instrumental in realizing a HWVN platform. We have also explored the missing pieces of the puzzle that are needed for successful realization of HVWN.

CHAPTER 4

SERVICE DIFFERENTIATION IN SOFTWARE DEFINED VIRTUAL HETEROGENEOUS WIRELESS NETWORKS

4.1 Introduction

There has been a drastic hike in cellular network traffic in recent time and it is growing at an ever-increasing rate. Also with the advent of smart phones and tablets novel services are emerging that have high QoS requirements. Mobile operators, with their limited licensed spectrum and vendor locked-in network gear are struggling to cope with such a paradigm shift of the traditional voice-centric networks to a more data-centric one. In such a context, network operators and vendors all over the world, are seriously considering network function virtualization (NFV) (Chiosi *et al.*, 2012a), as an inevitable evolution of carrier networks, to ensure efficient resource utilization while decreasing capital and operational expenditure (CAPEX & OPEX). Virtualization is the process of abstracting network resources (both physical nodes and radio spectrum), so that, multiple isolated virtual network operators (VNOs) can have shared access to these resources to build their own customized (virtual) networks.

In Chapter 3 we proposed an end-to-end programmable HVWN that provides a common physical substrate to build different virtual networks that uses different RATs. In this chapter, we now focus on a particular part of that generalized architecture, i.e., the case of programmable virtualized wireless networks that consist of cellular and fixed WiFi networks. More specifically we study how differentiated services can be provided in such a programmable virtualized platform. We have proposed to use the spare bits of OpenFlow (McKeown *et al.*, 2008) packet structure to implement virtual network entities, e.g., virtual networks, virtual switches, allocated radio resources of a virtual operator, etc. We also emphasized the use of northbound APIs to facilitate composing complex network applications.

The rest of the chapter is structured as follows: in Section 4.2, a brief summary of related work is presented. Section 4.3 gives a brief description of architectural components of the cloud model of the heterogeneous wireless networks that we denote as 'HetNet cloud'. The use of northbound API to facilitate virtual wireless network management is discussed in Section 4.4. In Section 4.5, we present the emulation results for service differentiation for two virtual wireless networks that share the same physical infrastructure. Challenges in implementing a HetNet cloud model are discussed in Section 4.6. Finally, we conclude the chapter in Section 4.7.

4.2 Related work

Use of SDN and cloud computing for implementing wireless networks is receiving increased attention from both industry and academia alike in recent time. OpenRoads (Yap *et al.*, 2010a) is one of the first works on virtual wireless network using SDN, where multiple virtual networks running on a common switching fabric are isolated at flow level using a FlowVisor (Sherwood *et al.*, 2009). Relevant works on SDN and cloud computing for wireless networks have been discussed in Chapter 1. Pertinent architectural models are: cloud RAN (C-RAN) (cmr, 2011) proposed by the China Mobile Research Institute (CMRI) that proposes partial and full centralization of baseband signal processing for RANs. Moving EPC to the cloud was proposed by Kempf et al. in (Kempf *et al.*, 2012). A RAN as a service (RANaaS) model is analyzed by the iJOIN (ijo) project and here the RAN is implemented in a cloud infrastructure. EPC as a service (EASE) (Taleb *et al.*, 2015) proposes a cloud-based elastic mobile core network model; this article also describes different implementation models of EASE.

4.3 HetNet Cloud architecture

In its most generic form, a HetNet cloud architecture is composed of distributed wireless datacenters (WDCs) interconnected by a high capacity optical network. In its business model, the physical and virtual infrastructure is deployed and managed by an infrastructure provider (InP) and the mobile virtual network operators (MVNOs) or service providers (SPs) lease the virtual nodes from the InP and deploy their own customized services. It is to be noted that a InP can also play the role of a MVNO or SP.

A typical HetNet cloud architecture for an urban area is presented in Figure 4.1, where the WDCs are connected by a high capacity optical fiber network composed of fiber optic cables and optical cross connects (OXCs) for wavelength routing. In this form of implementation, radio access to the user equipments (UEs) is provided through optical fiber front-haul, connecting the transmitting RRHs/APs to the WDCs. The functional blocks of a WDC appear in Figure 4.2, in this section, we briefly describe different parts of a WDC.

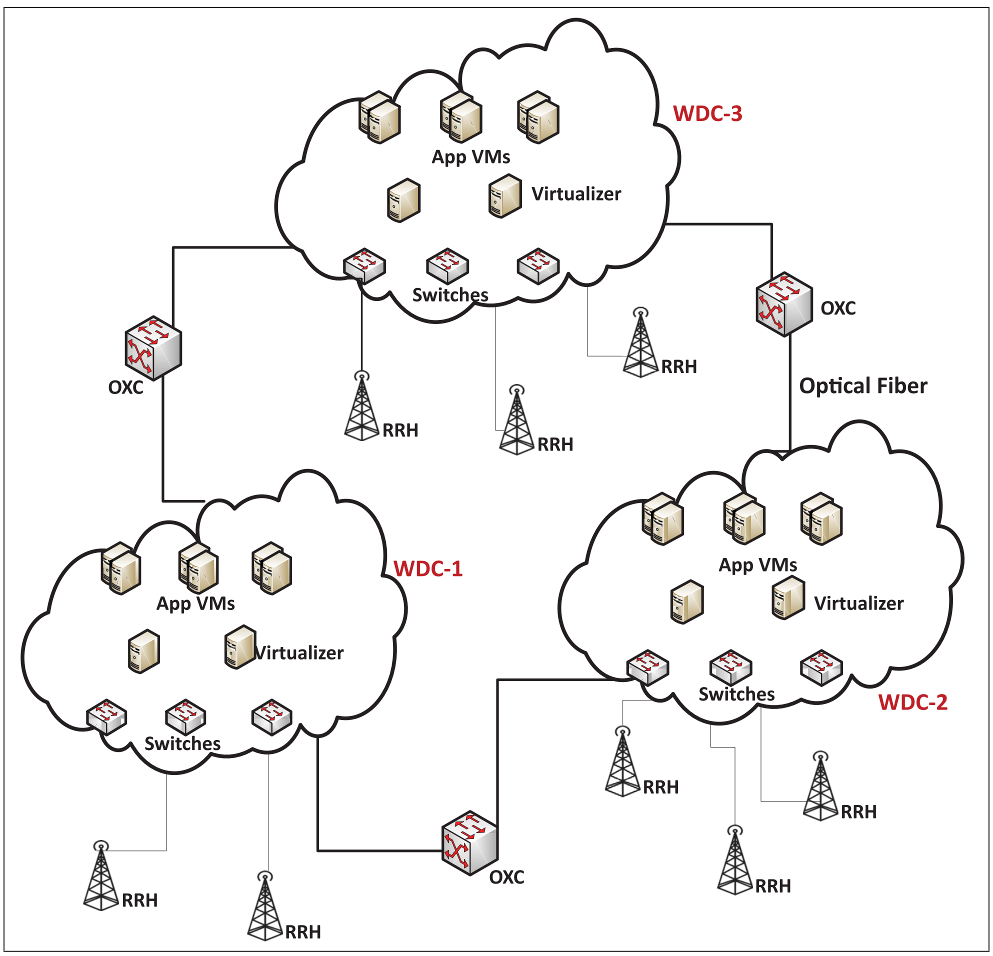


Figure 4.1 A HetNet Cloud architecture consisting of distributed WDCs

4.3.1 Application layer

A MVNO or SP (e.g. online gaming provider, YouTube, Netflix, etc.) implements its service functionalities by selecting the network processing components from the InPs either as software components in virtual machines (VMs), and/or dedicated physical hardware modules (especially for baseband processing for services requiring high QoS). Network applications (e.g. routing, load balancing, offloading, mobility management, etc.) in the application layer manage the end-to-end network connectivity by dictating the forwarding behavior of the underlying programmable switches, routers and RRHs. Proper synchronization among different application modules are very important for predictable network behavior, a northbound API (Monsanto *et al.*, 2013) can ensure the synchronous composition of various network applications.

4.3.2 Software modules

Different network functional and processing nodes, implemented as software modules in VMs belong to this layer. They consist of the software implementation of various EPC nodes, i.e. PGW, SGW, MME, PCRF, HSS. It also contains software modules for baseband processing, e.g. soft-eNB, soft-BS, soft-APs, etc.

4.3.3 Northbound API

For a VNO operation, different functionalities are needed for end-to-end service provisioning. For example, a routing application should program the switches to route the packets to its destination; a load-balancer should divert excess traffic to a neighboring lightly-loaded cell when a cell become overloaded; to filter malicious traffic, a firewall application is needed; specific applications are also needed for different deep packet inspection (DPI) purposes. To write a single monolithic controller application using the match-action based programming semantic of OpenFlow (McKeown *et al.*, 2008) (which is the de-facto SDN southbound API) is technically challenging and there is high possibility of coding bugs that disrupts proper

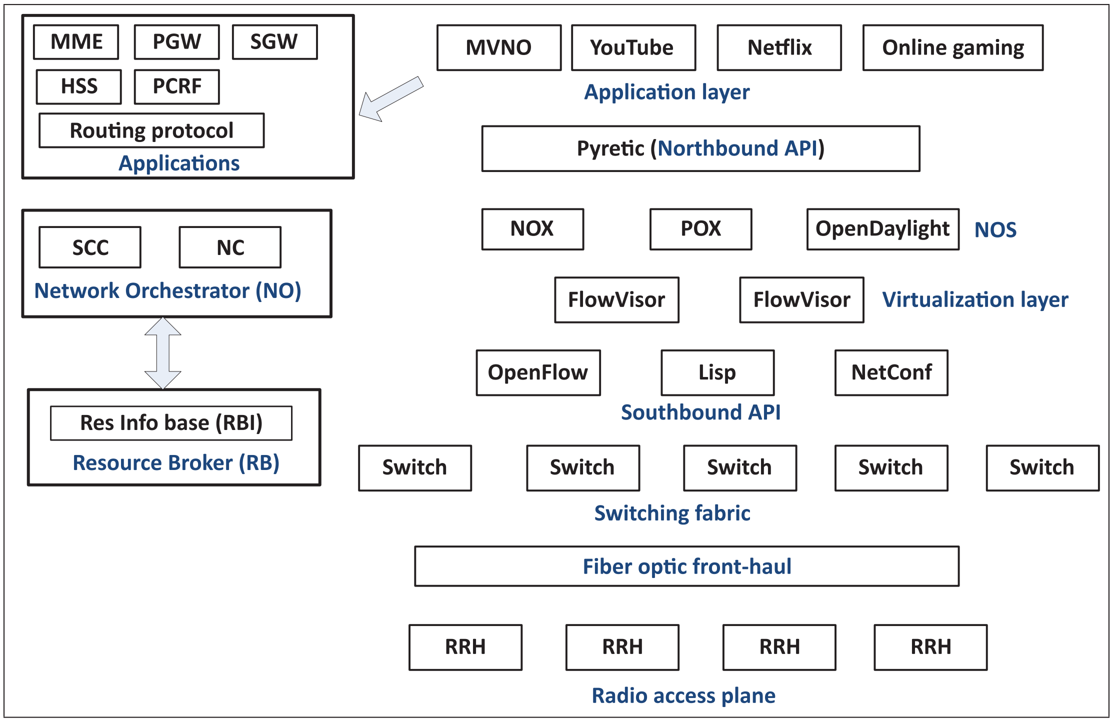


Figure 4.2 Schematic of a WDC processing blocks in HetNet Cloud

network functioning. To alleviate this problem, a high level abstraction for programming the underlying network, made possible by northbound APIs like Frenetic (Foster *et al.*, 2011) and Pyretic (Monsanto *et al.*, 2013) is extremely useful. Using these northbound APIs, modular and re-usable network applications can be built and composed in sequence or in parallel (Monsanto *et al.*, 2013).

4.3.4 NOS & East-Westbound API

A network operating system (NOS) abstracts the global view of a network and allows a network programmer to write different control applications as a centralized platform. Most popular NOSs are: NOX (Gude *et al.*, 2008), POX (Mccauley), OpenDaylight (ope, b) which use OpenFlow (McKeown *et al.*, 2008) based flow level control mechanisms. For horizontal control information exchange between NOSs from different platforms, standardization of a east-westbound API is also important.

4.3.5 Virtualization layer

Virtualization enables multiple isolated VNOs to share a subset of network nodes as well as the radio spectrum. In the HetNet cloud architecture, flow-level virtualization (Sherwood *et al.*, 2009) is used. The virtualizer acts as a transparent proxy sitting between the network controller and the southbound API and enforces isolation between slices by rewriting policies, dropping conflicting rules, etc.

4.3.6 Southbound API

The controller platform modifies the forwarding behavior of the forwarding elements (switches, APs, RRHs, etc.) via a southbound API, it acts like a compiler for transforming the controller instructions to low level instructions that the nodes understands. The de-facto southbound API is OpenFlow (McKeown *et al.*, 2008), that is used extensively by both academia and industry alike. Other popular southbound APIs are NetConf (net, 2013), LISP (Ratal *et al.*, 2012), etc.

4.3.7 InP's resource management layer

The resource management layer of an InP keeps track of the usage of physical and virtual resources. It consists of a network orchestrator (NO) and a resource broker (RB).

4.3.7.1 Network Orchestrator (NO)

A NO manages the computing, storage and networking resources shared by different VNOs. It has two sub-modules:

a network controller (NC): responsible for provisioning and management of virtual network nodes to VNOs; a storage & compute controller (SCC): it is in-charge of the assignment and subsequent management of storage and computing resources to various VNOs.

4.3.7.2 Resource broker (RB)

It acts as the central resource information base for the InP. It manages the usage status of resources (compute, storage & networking), so that, the NC and SCC can have a global usage view of VNOs and manage them efficiently.

4.3.8 Baseband processing

The HetNet cloud support multiple radio access technologies (RATs). VNOs can provision either cellular (LTE, WiMAX, 3G, etc.) networks , WiFi ISP, or device-to-device (D2D) communication, e.g. smart grids or even sensor network services. To facilitate such diverse RATs, radio processing chains are decomposed into different PHY and MAC layer processing blocks, so that, a VNO can choose the blocks required for its service and provision its customized network. It is to be noted that carrying out all the baseband processing in a WDC might not be optimal for traffic of all QoS classes as some might have very tight requirement of processing delay. Hence the delay-sensitive traffic (voice, live video, etc.) should be processed at the RRHs capable of baseband processing, while more delay tolerant traffic might be pushed to the WDC for processing. It is worth noting that, the length of fiber optic cable from the WDC to the RRH is a very important design consideration (Rahman *et al.*, 2015d).

4.3.9 Radio access plane

Radio access to the UEs is provided by fiber-fed RRHs, high volume of PHY layer processing signals justifies the use of optical fiber front-haul. Due to the varying nature of the wireless environment, dynamic provisioning of radio resources, strict QoS management and handling user mobility requires frequent exchange of control information between the controller and the underlying network substrate. For a centralized control architecture this might become a serious bottle-neck. Hence, local controllers need to be installed at RRHs to handle frequent local events, like: user mobility between neighboring cells, transmission power management, dynamic frequency allocation, etc.

4.4 Using northbound API to facilitate virtual wireless network management

Wireless networks need various applications to run simultaneously to achieve full network operability. These applications range from traffic routing, mobility management, resource scheduling, policy enforcement, billing functionalities etc. While SDN enables the creation of different applications, it does not make it easier because of the low-level machine language-like primitives used in southbound APIs, e.g. in OpenFlow (McKeown *et al.*, 2008). Moreover, creating portable modular applications is very difficult using standard southbound API, if not impossible. Domain specific programming language like Pyretic (Reich *et al.*, 2013) (built on top of POX (Mccauley) controller platform) make building modular network programs an ease. We propose to use a northbound API like Pyretic to build modular applications for virtual wireless networks. The parallel and sequential composition operators of the language make it possible to compose complicated network applications by composing (in parallel or in series) simpler applications. Using the abstract packet model in Pyretic, OpenFlow (McKeown *et al.*, 2008) packet header fields can be extended to include virtual fields, that can be used to associate packets with high level meta data. In (Monsanto *et al.*, 2013), Monsanto et al. gives a comprehensive description of the usage of Pyretic language model.

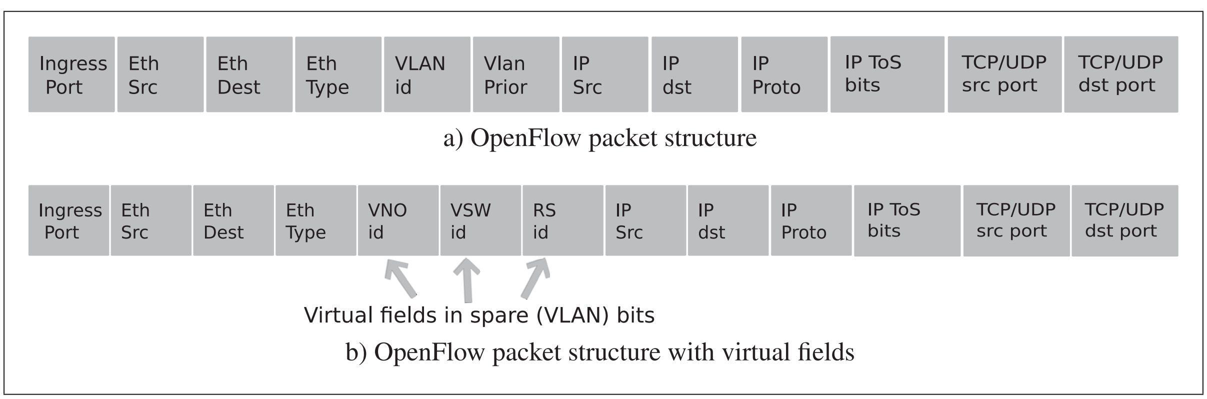


Figure 4.3 OpenFlow packet model and the modified packet model with virtual fields for virtualization

We propose to extend the abstract packet model in Pyretic to implement virtual wireless networks through *abstract topology* (Monsanto *et al.*, 2013). The spare bits (e.g. VLAN, MPLS fields) in an OpenFlow packet (notice Fig. 4.3a) are used for specifying virtual networks, virtual network node and wireless spectrum to be used for transmitting that particular packet. One such packet model is shown in Fig.4.3b. VNOs are identified by a *VNO* id in the virtual field; these ids are unique as VNOs should be uniquely identifiable. Virtual nodes (switches, BSs, APs, middle boxes, etc., we generically refer to them as 'switches' throughout this chapter) are identified with a *virtual switch (VSW)* id. These ids are unique to a InP but different InPs can use the same VSW id, as it is locally significant. For flexible allocation of radio resources a *Radio Spectrum (RS)* id is used to specify the transmission frequency for a VNO. This gives a great flexibility in being able to do wireless resource allocation on a per packet (per flow) granularity which will facilitate tackling different radio propagation problems, like interference management, traffic offloading, etc.

Figure 4.4 Multi-layer virtual network protocol stack

Virtual wireless networks can be built using the network object (Monsanto *et al.*, 2013) model of the Pyretic platform. It consists of the abstract virtual network (VN) topology along with different network policies depicting the behavior of the topology. The VNs are composed of different virtual components from the physical and/or virtual switches that a VNO leases from InPs. By using the virtual fields in the abstract packet, high level information (like virtual network id) can be used in the packet header to identify a VNO. In addition to the *width* expansion of the packet header by including virtual fields, *height* can also be increased by *stacking* multiple values for a single field (both extended and normal OpenFlow) (Monsanto *et al.*, 2013). For example, a virtual switch id can be stacked on top of a physical switch id which will hide the identity of the physical switch and applications above can operate on the abstract virtual switch. The mapping between the virtual and physical resources is handled by the Pyretic runtime. This ability of vertical abstraction enables the creation of multiple nested layers of virtual networks (c.f. Fig. 4.4) on a physical substrate and all the nitty-gritty detail of the implementation is left to the runtime system, so that, a programmer does not have to worry about the underlying networking detail and can write complex modular network applications. The runtime maintains a unique identifier corresponding to a unique set of virtual fields and the stacked fields of OpenFlow-compliant fields. The identifiers are stored in spare bit locations of the packets and a table is maintained by the runtime system for mapping between the identifier and the extended data (Monsanto *et al.*, 2013). We discuss several use cases of the HetNet Cloud in the following sub-sections.

4.4.1 Interference management

Present day wireless networks is a heterogeneous mix of macro, micro, pico, femto cells, RRHs and WiFi APs. While decreasing the cell size provides a performance leap, it also poses various challenges, especially interference among different wireless nodes (Pérez *et al.*, 2011). Hence interference mitigation in such a wireless environment is a critical and challenging issue. Interference management applications can be implemented as a dynamic policy that changes the network behavior dynamically depending on network state. If a UE experiences an interference level above the acceptable threshold, the controller can get this information from the channel quality information (CQI) fed-back from the UE. Upon receiving this information the controller can adjust transmission parameters (e.g. transmission power, DL frequency) to alleviate the problem. For example, if the interfering BS belongs to the same operator, it is very convenient for the operator to change the DL frequency to an appropriate for the interrupted UE as it has a global view of the network. In case the interfering BS belong to a different VNO sharing the network resources from the same InP, the operator can exchange the interference information relatively quickly (through high speed network interconnect of the wireless data center) to resolve the issue. This will reduce service degradation and increase the quality of experience (QoE) for the UEs.

Fig. 4.5 shows an example of message exchanges between two local controllers for managing interference to a UE located at the cell border of RRH1 and RRH2 belonging to the same VNO. From the CQI sent by the UE, the Local Controller1 (LC1) located at RRH1 (c.f. Fig. 4.2) identifies that the UE is experiencing interference in the DL direction from the neighboring RRH2. To resolve the interference problem, LC1 requests (using the radio link denoted by the dotted line) Local Controller2 (LC2) situated in RRH2 to decrease its transmission power in the DL direction. RRH2 responds positively by lowering its DL transmission power. LC1 again checks the CQI from the UE and observes that it has not improved above the threshold level. According to its *action logic sequence (specified during the programming phase)* it notifies the Central Controller (CC), located at the wireless data-center, about the interference and requests a change in DL for the particular UE. Having the global view of the whole network the CC selects a DL frequency for the RRH1 that will not interfere with the used frequency of any of its neighbors. RRH1 changes the DL frequency for the UE and it continues communication with RRH1 with satisfactory QoS. Cells (RRHs) of different VNOs can also communicate among themselves to resolve such interference issues. But in that case, the communication will involve the FlowVisor and the CCs of the VNOs.

4.4.2 Mobility management

Handover of a UE from one BS to another involves interaction among several network nodes, considerable amount of state transfer, also in some cases service disruption for brief time and/or transmission of redundant data. For example, in a LTE network, a handover request from a UE is sent to a local mobility anchor (i.e. S-GW, c.f. Fig. 0.1) that handles changes in user location and stores user states. If the UE needs to be switched to the jurisdiction of another S-GW, the mobility management entity (MME) has to be informed. The MME administers user reachability and also is responsible for S-GW and P-GW selection. The MME selects a S-GW for the

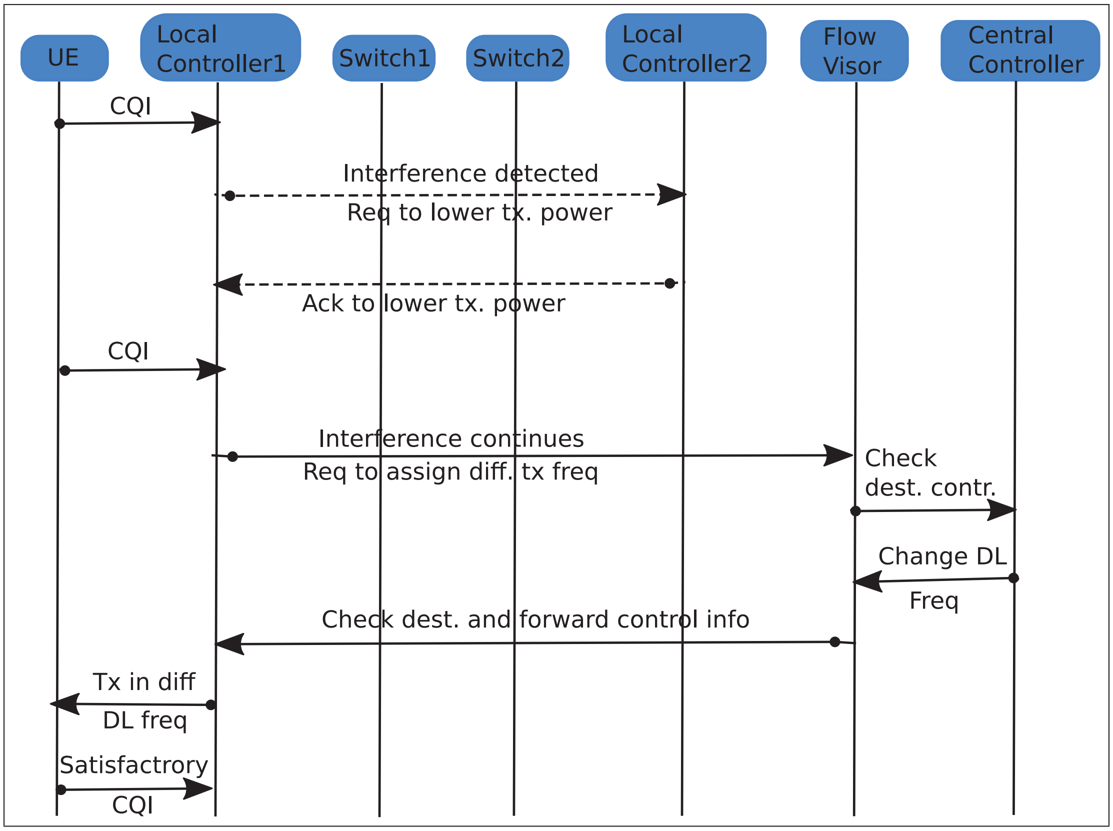


Figure 4.5 Interference management message exchange between Local Controllers belonging to the same VNO

UE and the user states and data are transferred from the old S-GW to the new one. The whole process takes a considerable amount of time as it involves control and data plane information to be transferred between multiple physical devices. This results in degradation of QoE for the UE as the continuous data transfer between the eNB and the UE is disrupted. This problem is greatly resolved in a HetNet cloud model where the different nodes (MME, S-GW, P-GW, eNB, etc.) are implemented in software in a centralized resource-pooled location and where a centralized controller with a global view of different nodes can transfer data among involved nodes in solid-state and/or wire speeds.

Moreover, present day cellular networks are characterized by a high penetration of smaller pico and femto cells (Chandrasekhar *et al.*, 2008). Due to the smaller size of the coverage radius of a BS, users experience frequent handovers. In a HeNet cloud, the RRHs of the femto cells are equipped with a *local controller* that can take certain administrative tasks like handover management and radio resource provisioning among its neighbor femto-BSs. The top layer controller which has a global view of the network can instruct the local controllers of neighboring femto-BS in a certain geographic area, to compute forwarding rules for handover management among the considered femto-BSs. Hence, hand-off of users among the local femto-BSs can be managed without the intervention of the high layer nodes (MME, S-GW, etc.) which will make the hand-off process faster and subsequently provide better user experience. The forwarding rules in the switching fabric can also be installed pro-actively to further expedite the handover process.

4.4.3 Traffic offloading in a HetNet eco-system

In today's heterogeneous network environment, users are practically submerged in a wide variety of wireless accesses, each with different access mechanism and charging policies. Users have access to multiple networks (cellular, fixed) at the same time. Also mobile devices (smart phones, tablets, etc.) run a plethora of applications having different quality of service (QoS) requirements. Some applications have strict delay limits for low volume of traffic (e.g. voice service); for some applications, a certain amount of delay is acceptable but the required data rate is very high such as for bulky file transfers, watching videos in You Tube, streaming movies, watching sports channels, etc. Normally cellular networks have higher charging policies for data traffic as the service provisioning involves the use of licensed radio spectrum and use of expensive special purpose hardware equipment. On the contrary, public WiFi networks (in coffee shops, campus networks, shopping malls, bus and train stations, airports) offer internet access at a relatively low cost or for no cost at all. This situation spurs the opportunity for cost saving for both network operators and UEs. Studies (Lee *et al.*, 2013) have shown that users spend almost 70% of their on-line time in WiFi coverage zones.

Through a collaborative effort from both the VNO and UEs, delay-tolerant high data rate user applications can be offloaded to WiFi networks while more delay-sensitive traffic can be provisioned by the cellular network. This will enable the VNOs to save expensive radio resources which are already ossified, and at the same time users will be able to save considerable amount of money for data traffic offloaded to WiFi networks that would otherwise have traversed the cellular network. Using the high level abstraction provided by Pyretic (Reich *et al.*, 2013), traffic routing of users can be controlled per user application (even per flow) granularity. Different charging policies can be applied between the involved parties as described in Sec 3.4.5. As an example for traffic offloading in a heterogeneous network, let us consider the message exchange diagram in Fig. 4.6. A UE served by VNO-A (a cellular network) comes in the coverage range of a fixed (WiFi) network, VNO-B. In real life, the situation is similar to, when a cellular user enters a shopping mall or coffee shop that has public Hot-Spot (either free or paid). The LC-A of VNO-A receives traffic from the UE and from the UDP port number (e.g. 8011) of the packet it immediately identifies that the UE is streaming video traffic. Serving the video traffic using cellular spectrum is very costly and is a unreasonable wastage of radio frequency when the UE has access to WiFi network that does not use licensed wireless spectrum. A reasonable economic choice would be to offload the video traffic to the WiFi network. Hence, the LC-A sends request to the CC-A to initiate offloading of UE's video traffic to VNO-B (WiFi Hot-Spot). The control message is intercepted transparently by the FlowVisor (c.f. Fig. 4.2) which directs the message to CC-A, recognizing that it is the correct destination from the virtual header fields. CC-A sends a handover request to CC-B for traffic offloading which is again intercepted by the FlowVisor and directed to CC-B. Upon receiving the request, CC-B sends a positive acknowledgement to CC-A granting the handover and installs new flow-rules in its associated switch, Switch-B, and instructs LC-B to take necessary steps (e.g. selection of DL/UP frequencies, transmission power, etc.) for the UE association to the network. On the other hand, CC-A installs new flow-rules in Switch-A to re-direct the traffic to VNO-B and also instructs LC-A to remove the resource reservation for the UE. In this way the video traffic handover completes. If the UE is using multiple services, e.g. using VoIP at the same time, this voice traffic flow can be handled by the cellular VNO-A to provide better quality of experience (QoE). In this way, user traffic can be offloaded per-flow/per-application basis in the HetNet Cloud architecture.

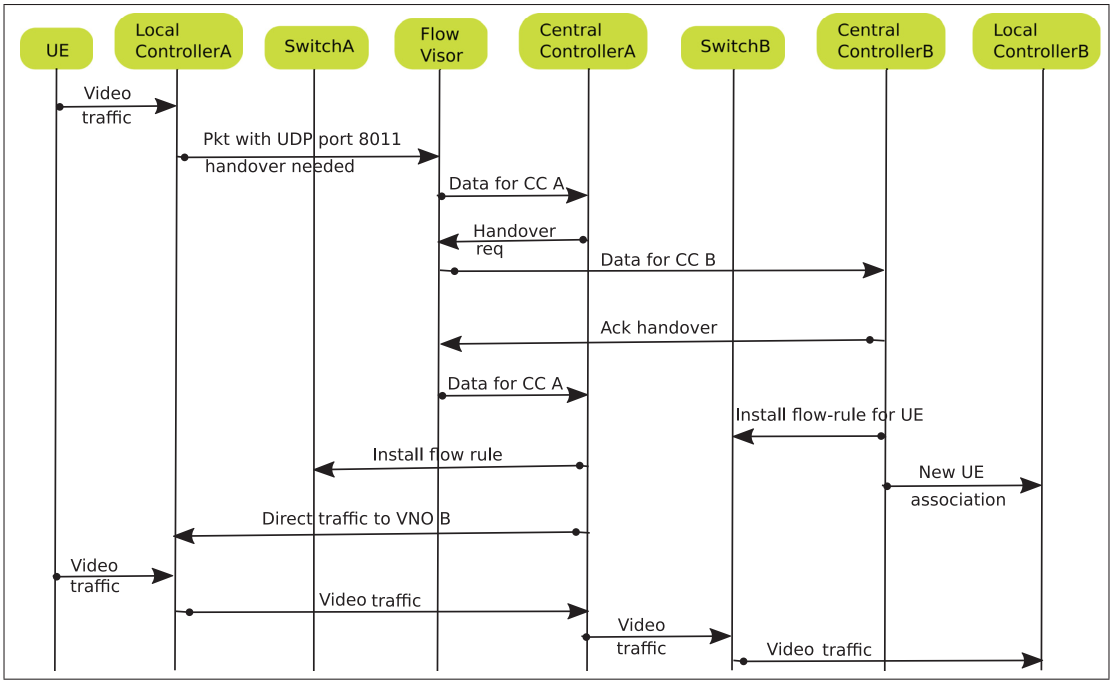


Figure 4.6 Message exchanges for traffic offloading between VNOs

4.4.4 Secured network

A secure communication system is a prerequisite for any network and wireless networks are no exceptions. With the advent of modern communication technologies, security threats have also increased and for the *ubiquitous and open* communications channels, wireless networks are particularly vulnerable (Ding *et al.*, 2014). Different solutions have been proposed for addressing security issues in a software-defined wired network; for example, Ethane (Casado *et al.*, 2007) allows network programmers to write fine grained network policies to bolster security. Some other notable works are Resonance (Nayak *et al.*, 2009), FRESCO (Shin *et al.*, 2013), NetFuse (Wang *et al.*, 2013), CloudWatcher (Shin and Gu, 2013), etc. In a wireless network, threats exists in the forms of malicious users intruding a network to sniff out important credentials from legit users; making a denial-of-service (DoS) attack to disrupt the network operation by exhausting network resources; in a wireless network, a DoS attack targeting on radio spectrum can be very severe as it will starve out the user of radio resources, eventually bringing the communication down. To alleviate these problems, different security applications, such as access controller, intrusion detection system (IDS), deep packet inspection (DPI) system, etc. can be built and using Pyretic's sequential and parallel composition operators, these applications can be combined together to build a secure wireless network.

In this regard, Kinetic (kin), a domain specific language built on top of Pyretic can be very useful. It is a SDN control system that enables network programmers to implement control programs which can dynamically change network behavior triggered by various network events. In Kinetic, a finite state machine (FSM) abstraction is used to express dynamic network policies that change network behavior based on various network events, e.g. intrusion, anomaly detection, etc.

4.4.5 Internet of things (IoT)

In a constantly evolving networked eco-system, it is predicted that in a near-future different devices will be connected together realizing a all-connected network, which is also dubbed as network of everything or internet of things (IoT). In such a context, device-to-device (D2D) communication is emerging as a hot topic of research (Asadi *et al.*, 2013), (Cai *et al.*, 2014). In a HetNet Cloud architecture, provisioning of IoT network is very convenient. A dedicated virtual network slice can be created (same as a VNO/SP) that will administer the interconnection among different connected systems. The protocol requirement in a D2D communication network is different from a traditional communication system as it requires processing of lower volume of data in infrequent epochs. Also the transmission bandwidth requirement is different, as it requires a comparatively lower bandwidth for data transmission and reception. Moreover, devices are located in a variety of wireless environment, requiring different transmission capabilities, hence, selecting the best mode to transfer data is a critical issue. In a software-defined virtual network, a central controller has global view of the underlying connected devices, hence it can select the optimal transmission mode for a particular device pairs. Also, in a software-defined environment it is much easier to implement customized network protocols for proper D2D communication in different scenarios.

4.5 Service differentiation in heterogeneous wireless networks

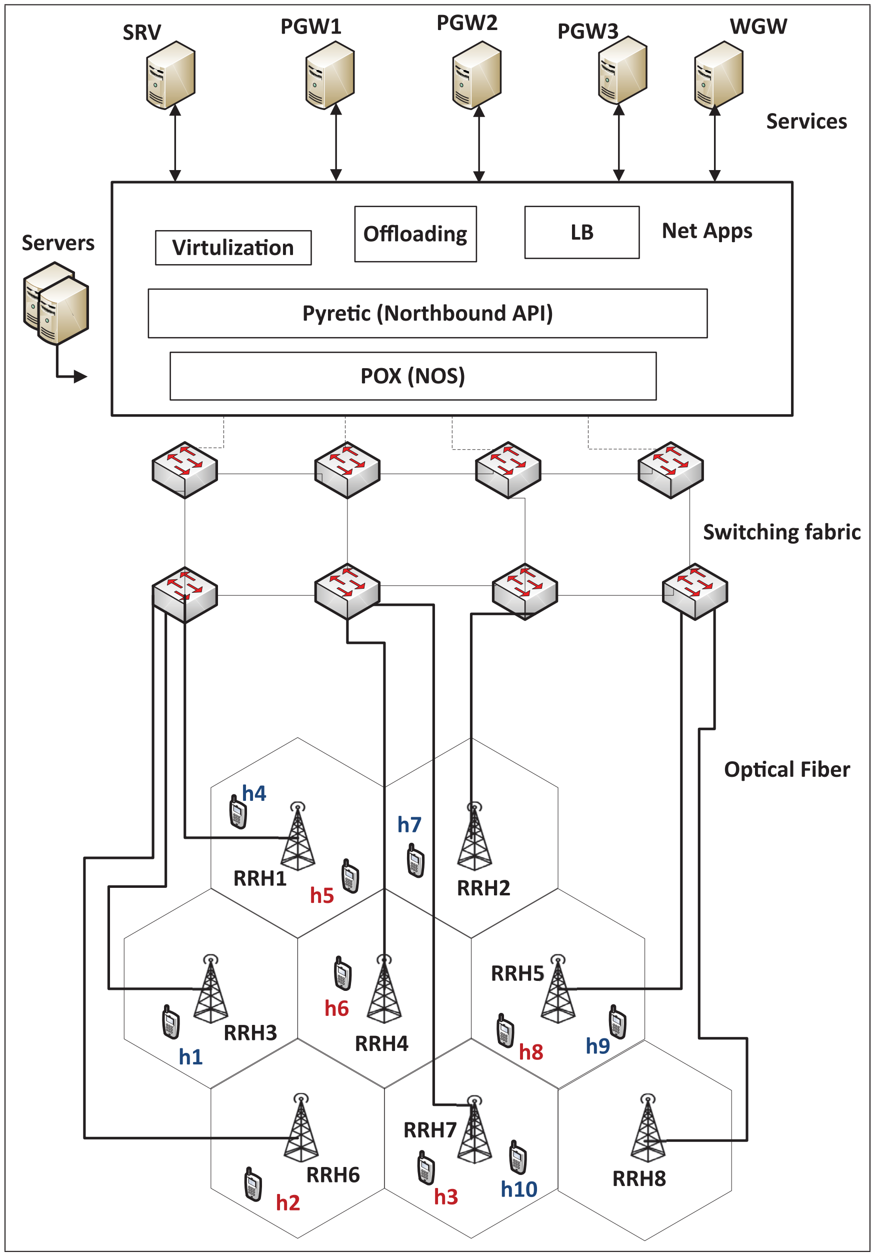


Figure 4.7 Virtaul wireless networks emulation scenario

As a proof of concept, we have implemented two VNOs in the Mininet (min) emulation platform. These VNOs are implemented as two isolated slices sharing the same physical resources, e.g. computing & storage nodes, network switches, RRHs, etc. We envision a NFV implementation for the operators, where various network processing nodes (e.g. PGW, SGW, MME, PCRF, HSS, baseband processing units, etc.) are implemented as software modules in data center servers. The schematic of the emulation structure is shown in Figure 4.7. In this experiment, we studied service differentiation provisioning for virtual wireless networks in a HetNet cloud model. We studied how various mobile services can be provided with differentiated QoS depending on the application requirement and also the user subscription category. More specifically, we studied load balancing for users of VNO1 that have different subscription categories (prioritized and normal) and also the offloading of delay tolerant traffic from VNO1 to VNO2. As performance metrics, we measured round trip transmission delay (RTTD) and achievable throughput while implementing the traffic offloading and load balancing. Network applications, i.e. virtualization (slicing), offloading and load balancing were written using Pyretic (Monsanto *et al.*, 2013), a domain specific programming language (DSPL), which is a northbound API, that uses POX (Mccauley) as the network operating system (NOS). While the southbound API, OpenFlow (McKeown *et al.*, 2008), populates the forwarding table of the underlying programmable switches (open virtual switch (OVS)) to forward traffic from the respective VNOs. UEs of VNO2 have degraded link quality than that of VNO1, hence their lower throughput.

In the emulation setup, VNO1 is a MVNO providing mobile network service, whereas VNO2 is a WiFi service operator, providing internet access to users through unlicensed spectrum. Varied radio link qualities for the two types of networks are realized by implementing more lossy links for the WiFi network. From QoS point of view, VNO1 guarantees better service quality via its dedicated licensed spectrum and high performance servers connected by high capacity network links. As show in Figure 4.7, users h2, h3, h5, h6 and h8, marked red, belong to VNO1, where h2 and h3 are prime customers and users h1, h4, h7, h9, h10, marked in blue, are served by VNO2. In the wireless data center, connection between the servers and switches are of 1GB capacity, no transmission delay and loss are assumed for these links. Server hosting PGW2 VM (for serving regular clients from VNO1) is connected via 800 Mbps link having 2% packet loss, while for the WGW VM (to serve delay-tolerant traffic), the link is 600 Mbps, with 0.5 ms delay and 2 % packet loss. These links are configured in such a manner so as to simulate a differentiated QoS. Connections between switches and between switches and RRHs are of 1GB capacity. The fiber length from WDC to the RRH is 2 km, hence a 0.01 ms of transmission delay is assumed, as typical delay for radio transmission over optical fiber is 5μs/km.

Users of VNO1 have simultaneous access to both the mobile network and the WiFi network. Given the omni presence of WiFi networks in our everyday ICT eco-system, e.g. WiFi networks in campuses, offices, shopping malls, airports, stadiums, etc., it is a reasonable assumption. For the service differentiation evaluation, delay sensitive traffic (e.g. file transfer, video streaming, etc.) from the users of VNO1 directed to the PGW1 (default server for data traffic for the UEs of VNO1) are offloaded to the WGW server, that belongs to the VNO2. This helps saving licensed spectrum that can be used for providing services having tighter QoS requirement, e.g. services producing more delay-sensitive traffic. Also, in case of VNO1, traffic from privileged users (h2, h3) are directed to the server (PGW3) capable of providing better QoS from an achievable throughput and RTT delay point of view. Table 4.1 shows the RTT delay and throughput for different service differentiation cases, when the users are static. RTTD are measured in ms and the throughput in Mbps. The 'Regular' column shows delay and throughput when traffic from users are forwarded to the server 'SRV'. The 'Offloading' is the measure when delay-tolerant traffic from VNO1 is offloaded to VNO2 and the 'Load balancing' shows the result of differentiated services for privileged (h2, h3) and regular (h5, h6, h8) users. Minimum and average delays are shown in the table. For the control information exchange between the controller and switches, the transmission time for the first packet is pretty high which in turn increases the average packet delay; in fact, the long term average delay is lower than the noted average delay in Table 4.1. No offloading or load balancing is assumed for VNO2.

We implemented a random mobility model for the users of VNO1 and VNO2. Table 4.2 shows the maximum (for the first packet) and the average packet RTTD, including the average achieved throughput. The delay depends on the connected RRH and the links' qualities to the service nodes. BH stands for 'before handoff' and AH denotes 'after handoff' RRH

Table 4.1 Delay and achievable throughput for static users

UE	Regular		Offloading		Load balancing	
	RTTD [min - avg]	Th [Mbps]	RTTD [min - avg]	Th [Mbps]	RTTD [min - avg]	Th [Mbps]
VNO1 cor. users	0.2 - 46	95.2	1.6 - 11	27.3	0.12 - 35	95.7
VNO1 reg. users	0.36 - 28	94.7	1.32 - 12	28.8	0.21 - 39	79.8
VNO2 users to SRV	0.36 - 51	31.2	-	-	-	-
VNO2 users to WGW	1.45 - 50	10.3	-	-	-	-

connections of users. According to the 3GPP standard, preferable delay for LTE voice and video is <150 ms and maximum allowable delay is <400 ms. From the average RTTD values in Table 4.1 and 4.2, we can see that the HetNet cloud architecture is very well able to satisfy these requirements.

4.6 Challenges

There are certain challenges that need to be addressed for successful realization of a HetNet cloud model; in this section we will briefly discuss some of them.

Balancing network complexity and flexibility

It is an important design consideration. A flow-level (Sherwood *et al.*, 2009) virtualization allows for network virtualization at the packet flow level but it is not possible to make any PHY and MAC layer modifications. On the other hand, virtualizing the radio chain (Bansal *et al.*, 2012) allows the creation of new wireless protocols by using various PHY and MAC layer processing blocks but it has no provision for modular building of network applications. Hence,

Table 4.2 Delay and achievable throughput for mobile users

UE	Regular		Offloading		Load balancing	
	RTTD [max - avg]	Th [Mbps]	RTTD [max - avg]	Th [Mbps]	RTTD [max - avg]	Th [Mbps]
VNO1 cor. user before handover	1931 - 148	94.8	919 - 48	23.9	775 - 39	95.7
VNO1 cor. user after handover	1738 - 129	95.0	794 - 44	29	924 - 47	95.6
VNO1 reg. user before handover	1989 - 156	94.9	894 - 47	27.9	802 - 43	78.4
VNO1 reg. user after handover	2457 - 221	95	518 - 28	25.9	750 - 40	79.7
VNO2 user before handover	2142 - 172	35.5	-	-	-	-
VNO2 user after handover	1429 - 97	34	-	-	-	-

a design compromise is needed between the depth of network virtualization and achievable flexibility for building virtual networks.

VM placement

Placement of VMs is a critical issue, especially as some wireless applications are very delay sensitive and as maintaining very low RTTD is crucial. Hence, in a distributed WDC model, locations of the DCs are very important. Moreover, from green communication point of view, the WDCs should be established in places that have access to renewable energy sources.

Network security

While SDN facilitates building various security applications for networks, it has its fair share of security holes in the form of, malicious traffic flows, switch vulnerability, compromise of controller or control plane communication channels, etc (kreutz *et al.*, 2015). Proper measures should be taken to tackle such network vulnerabilities; for example, controller replication, so that a backup controller might take control of the operation in case the primary controller fails and an auto healing mechanism is initiated to recover from security attacks.

Standardization of APIs

It is important for integration and synchronous functioning of different network devices and applications built by various vendors and operators. OpenFlow (McKeown *et al.*, 2008) is the de-facto southbound API maintained by open networking foundation (ONF) consisting of leading industry and academic partners. Similarly, standardization of east-westbound and northbound APIs are also necessary for controller platform integration and facilitating modular application building.

Backward compatibility

Backward compatibility is significant for any new technology. SDN and cloud computing technology are supposed to be gradually included to the existing production networks. Hence, it is of utmost importance that these networks operate smoothly with the existing networks.

4.7 Conclusions

In this chapter, we presented the HetNet cloud model that implements virtual wireless networks atop shared substrate of physical infrastructures. We have used a northbound API for building modular network applications, like virtualization, traffic offloading and load balancing, and compose them together to achieve complex network functionalities, e.g. service differentiation in virtual hetnet wireless networks. Emulation results show that, the HetNet cloud can achieve very low RTTD and high data throughput while ensuring service differentiation per user per application basis. Critical technical challenges for realizing such a virtual network model have also been discussed. Investigating the deployment challenges in NFV implementation in the context of HetNet cloud is the subject of our ongoing research.

CHAPTER 5

DEPLOYMENT OF FULL DUPLEX MULTI-CELL SYSTEMS FOR DENSE URBAN AND RURAL ENVIRONMENTS

5.1 Introduction

Cellular networks are subjected to an exploding increase in data traffic which can be largely attributed to a plethora of data hungry smartphone applications. Cellular data traffic is predicted to increase by as much as 11 fold during the time period form 2013 to 2018 (Cisco, 2014). Due to this surging traffic demand with accompanied existing spectrum scarcity, cellular operators are looking for a transmission solution with improved and sustainable spectrum efficiency. In such a scenario, a full duplex (FD) system can provide a promising solution with its ability to almost double the network capacity by using the same frequency channel simultaneously for both uplink (UL) and downlink (DL) transmissions.

In a traditional system, a radio transceiver typically operate in half duplex (HD) mode, i.e., it either transmits or receives at any particular time epoch or frequency but it can not transmit and receive at the same time in the same frequency. Radio transmission is done either using time division duplex (TDD) or frequency division duplex (FDD) mode where transmission and reception take place at separate time slots or frequencies. In the TDD case, the transmitter (Tx) and the receiver (Rx) of a radio transceiver operate in the same frequency but uses different time slots. Whereas in the FDD case, the Tx and the Rx can operate simultaneously but in separate frequency channels. The major impediment for the bidirectional communication of a transceiver is the leakage power from the Tx to the Rx which is referred to as self interference (SI). The transmission power is almost a million time stronger than the received power which make the decoding of the received signal very difficult if not impossible. Hence, to make bidirectional transmission and reception possible i.e., to enable a FD communication, it is imperative to reduce the SI to a level where decoding of the received signal is possible (Hong and et al., 2014).

In recent time, significant progress has been made to reduce the SI to a great extent by a combination analog and digital cancellation techniques. Such techniques reduce the SI to such lower values that decoding of the weak received signals become possible. SI cancellation using multiple antennas was studied in (Bliss and et al., 2007), (Choi and et al., 2010), (Khandani, 2010), (Haneda and et al., 2010) where SI is cancelled taking advantage of antenna position and directionality. Studies on single antenna system (Knox, 2012), (Bharadia and et al., 2013) show significant suppression of SI where (Bharadia and et al., 2013) has reported to cancel SI by as much as 110 dB. Duarte et al. (Duarte and et al., 2014) has reported to cancel SI from 70 dB to 100 dB for multi antenna systems. In their implementation two antennas were used in each FD node. Jain et al. (Jain and et al., 2011) showed that a 73 dB cancellation is achieved by using signal inversion and digital cancellation for a 10 MHz OFDMA signal. To enable FD communication at the link-level, reducing SI is sufficient but for FD communication in a cellular level, additional interference components need to be removed.

5.1.1 FD single cell deployment

Fig. 5.1 shows a single cell TDD deployment for HD and FD systems. In the HD case (the figure at the left), there is no DL-to-UL or UL-to-DL interference because DL and UL transmissions take place in different time slots. But for the FD case (the figure at the right), the UL transmission from UE1 suffers from the SI, I_{SI} from the transmission radio chain of the BS. The UL user UE1 also generates Ul-to-DL interference, I_{UD} to the DL user UE2. For this reason, in addition to sufficient self interference cancellation (SIC), careful user selection in both UL and DL directions is important for co-channel FD operation in a cell (Goyal and et al., 2014). Di et al. (B.Di and et al., 2014) used a resource allocation method using matching theory for subcarrier allocation among transmitting and receiving nodes in a single cell FD system. A comparison of multi-antenna FD capacity was done against a HD MIMO system in (Barghi and et al., 2012). It was shown that under certain conditions the FD system gain can exceed the MIMO gain. A method dividing the cell interference regions into different segments and allocating frequency resources to the segments was proposed in (Shao and et al., 2014). Modification of CSMA/CA MAC protocols for FD operation was proposed in (Sahai and et al., 2011) (Sing and et al., 2011). The lack of synchronization in UL and DL transmissions in a FD multi-cell system gives rise to even more complicated interference scenario. In the following subsection we analyze the interference in a FD multi-cell system.

5.1.2 FD multi-cell deployment

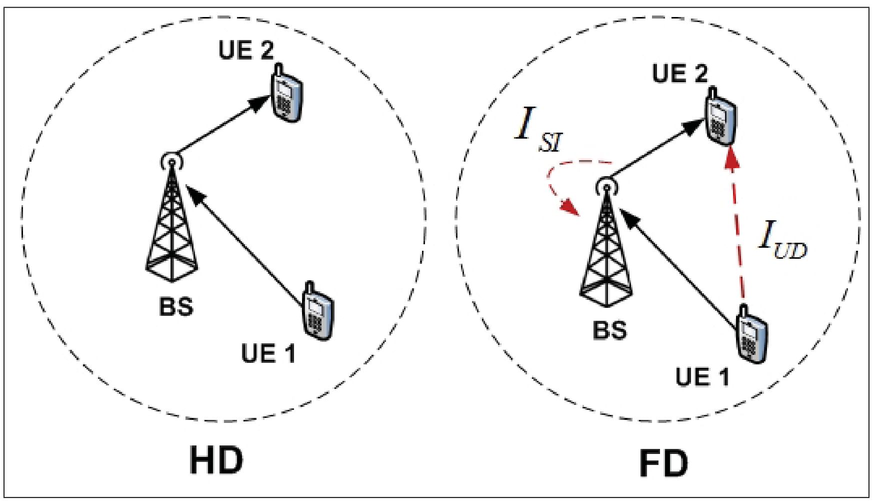


Figure 5.1 HD and FD single cell scenarios

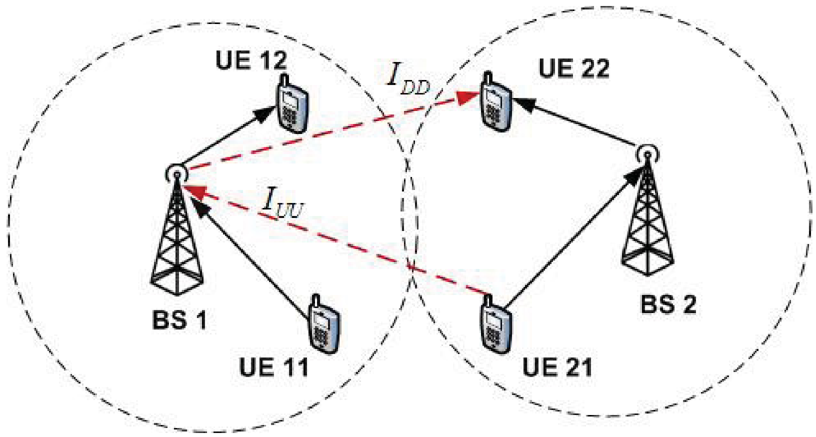
Interference patterns in a multi-cell scenario for a HD and a FD TDD deployment are shown in Fig. 5.2. In a HD implementation, the transmission in the UL and the DL are aligned which means that all the BSs transmit in the DL at the same subframe and receives transmissions from their connected users in the UL at the same subframe. Hence, the possible sources of interference are: UL-to-UL interference and DL-to-DL interference. This is illustrated in Fig. 5.2a where UE11 and UE12 are connected to BS1 and UE21 and UE22 are connected to BS2. During the DL transmission, BS1 transmits to UE12 and causes interference I_{DD} to the UE22. Similarly, in the UL transmission subframe, the cell edge user UE21 sends its signal to BS2 and in doing so it creates interfering signal I_{UU} to the uplink transmission of the neighboring cell user UE11.

The interference scenario becomes much more complicated for the FD system. The possible source of interference in a FD miulti-cell deployment are:

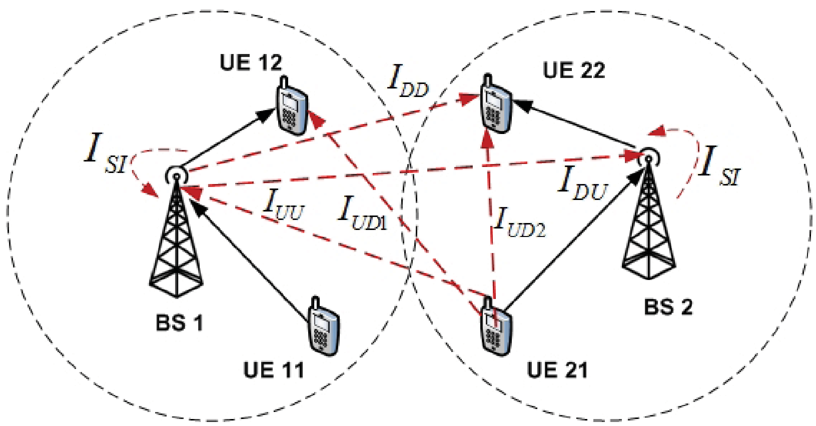
- self interference (SI) between the Tx and Rx chain of a BS, I_{SI};
- inter-cell UL-to-UL interference I_{UU};
- inter-cell and intra-cell UL-to-DL interference I_{UD};
- the DL-to-UL interference I_{DU} among neighboring BSs;
- inter-cell DL-to-DL interference I_{DD}.

Fig. 5.2b depicts these FD interferences for same BS and UE setting as in Fig. 5.2a. Here, we can see that both BSs generate SI I_{SI} between their Tx and Rx chains. BS1 generates DL-to-DL interference, I_{DD} to user UE22 and DL-to-UL interference I_{DU} to BS2. UE21 generates inter-cell UL-to-UL interference I_{UU} to UE11. It also generates intra-cell and inter-cell UL-to-DL interferences I_{UD2} and I_{UD1} to UE22 and UE12, respectively. So, it can be seen that a complex array of interference occurs in a multi-cell FD deployment and cancelling only the SI is not enough to harvest the promising gain of FD system (Sabharwal and et al., 2014), (Sultan and et al., 2015) and this will be evident in our analysis and obtained results in the subsequent sections.

Research in FD multi-cell systems has gained more traction in recent time (Huawei, 2015), (Chung and et al., 2015). The DUPLO (DUPLO, 2012) project is investigating the FD system for cellular small cell deployment; a joint UL-DL beamforming was designed for single cell deployment in (Nguyen and et al., 2014). In (Shen and et al., 2013) a scheduling algorithm for multi-cell deployment is proposed that selects UEs in UL and DL directions. The algorithm assumes fixed transmission power in the UL and the DL and it ignores the interferences among BSs and among the UEs. The results obtained from such assumptions do not give insight to the real deployment scenarios and inter-cell and inter-UE interferences are in fact



a) HD scenario



b) FD scenario

Figure 5.2 HD and FD multi-cell scenarios

significant sources of performance degradation which will be evident from the results shown in this chapter. In (Choi and et al., 2013), a FD multi-cell system has been analyzed employing UE selection and fixed UL and DL transmission powers. They have proposed to handle interference among BSs by null forming in the elevation angle of the BS antennas. Simeone et al. (SImeone and et al., 2014) proposes an analytical model for calculating achievable rate in a FD cloud radio access network (C-RAN) model. A user selection algorithm for a FD system was proposed in (H.H.Choi, 2014) where total cancellation of interference among BSs was assumed. While it would be ideal to totally cancel inter-BS interference, in a practical system deployment it is not possible to have perfect inter-BS interference cancellation. Goyal and et al

(Goyal and et al., 2013) presents a FD multi-cell system assuming full cancellation of SI with significant performance gain in FD DL. But expecting full cancellation of SI is way beyond the current state of the art which provides a SI cancellation of 110 dB. Spectral efficiency for a small cell FD system is provided in (Alves and et al., 2014), the authors consider that both the BSs and the UEs are FD capable.

The aforementioned works give interesting insights on FD multi-cell system performance. But the inference drawn from them does not properly reflect the practical FD deployment scenario due to various simplified assumptions. While some of the works assumed perfect SI cancellation, some ignored inter-BS and inter-UE interferences which are very critical in FD systems particularly for dense deployment scenarios. Moreover, interference dynamics of FD multi-tier cellular networks are also not captured from such simplified models. In this chapter, we take a more practical approach to analyze FD multi-cell networks where we consider all possible interference that might occur in such networks. For this reason, we consider the well accepted dense urban model of Madrid city (Agyapong and et al., 2013) for analyzing a multi-cell multi-tier FD network that consists of macro and pico base stations (BSs). We also investigate the FD network performance for a single-tier homogeneous deployment of macro BSs in a rural environment. This chapter claims the following contributions:

- identification the critical challenges for real world deployment of multi-cell single-tier and multi-tier FD networks;
- analysis of the FD performance trade-offs for a dense urban multi-tier cellular network. We have used the Madrid grid model proposed by METIS project (Agyapong and et al., 2013) that consists of macro and pico cells;
- analysis of the impact of co-located BS interference in FD performance for a single-tier homogeneous network deployment that consists of macro cells;

- proposal for joint intelligent user selection and power control algorithms for C-RAN and distributed RAN (D-RAN) deployments for the considered scenarios that enables reaping the gain that FD promises.

The rest of the chapter is organized as follows: in Section 5.2 we analyze the power control mechanisms and provide user selection and scheduling algorithms for FD networks for a C-RAN as well as a D-RAN deployment. System model for Madrid grid deployment and FD performance analysis results are provided in Section 5.3. In Section 5.4, system model and FD performance analysis for a single-tier heterogeneous network are provided. Finally, concluding remarks and potential future research directions are presented in Section 5.5.

5.2 User selection and scheduling

The throughput gain in a FD system depends on couple of networking settings e.g., distance among BSs, user distribution, mobility, channel propagation condition, DL & UL transmission power levels, etc. For this reason, a FD system should work opportunistically, i.e., when the network condition is favorable a BS should operate in FD mode and in case of an unfavorable condition, the BS should switch back to HD mode. Hence, the operating mode of the FD system can be attributed to a hybrid FD mode rather than a pure FD mode. In our analysis, we have assumed the BSs are FD capable while the user equipment (UEs) operate in HD mode. In this chapter, we have used proportional fair centralized and distributed schedulers that maximize the geometric mean (i.e., the exponential sum log throughput) of the scheduled users at any given time slot. The advantage of using the geometric mean as the scheduling metric is that besides ensuring maximum throughput gain it also tries to achieve fairness when scheduling users. The scheduler employs proportional fair schedule and pair mechanism for scheduling users. First it randomly selects a user then it schedules a second user such that the geometric mean of users' rate is higher than it was for the single user. In a similar fashion it continues to schedule the next user and so on. The scheduler objective function can be defined as

$$max\left(\sum_{n=1}^{N}\sum_{u=1}^{U_n}[\log(R_{n,u}^{DL})(t)+\log(R_{n,u}^{UL})(t)]\right) \tag{5.1}$$

$$\text{s.t.}\begin{cases} P_{n,u}^{DL}(t)=P_n^{DL,max}, \textit{when a BS has user} \\ P_{n,u}^{DL}(t)=0, \textit{when a BS has no user} \\ P_{u,n}^{UL}(t)=P_u^{UL,max}, \textit{when a user is scheduled} \\ P_{u,n}^{UL}(t)=0, \textit{when a user is not scheduled} \\ R_{n,u}^{DL}(t)\times R_{n,u}^{UL}(t)=0, n=\{1,2,...,N\} \end{cases} \tag{5.2}$$

where N is the number of BSs, U_n is the number of scheduled users in serving cell (i.e., BS) n, $R_{n,u}^{DL}(t)$ and $R_{n,u}^{UL}$ are the average DL and UL data rates, respectively, for user u at the serving cell n at time slot t and $1 \leq u \leq U_n$. The first four constraints enforce power control scheme for BSs and UEs. A binary power control scheme has been employed where a BS is active (and transmits at full power $P_n^{DL,max}$) if it has any user connected to it otherwise it is switched off. Similarly, if a UE is scheduled at a certain time period it transmits at full power $P_u^{UL,max}$ otherwise it is switched off. The fifth constraint enforces the half-duplex mode of operation for the UEs, i.e., at any given time slot t, they can either transmit to the BS they are attached to or receive transmission from the BS but not do the both simultaneously.

The instantaneous DL rate of a user can be expressed as

$$\begin{aligned} R_{n,u}^{DL}(t) &= \log_2\left(1+SNIR_{n,u}^{DL}(t)\right) \\ &= \log_2\left(1+\frac{P_n^{DL}(t)H_{n,u}^{DL}(t)}{\sigma^2+\sum_{i=1,i\neq n}^{N}P_i^{DL}(t)H_{i,u}^{DL}(t)+\sum_{i=1}^{N}\sum_{j=1,j\neq u}^{U_{ij}}P_j^{UL}(t)H_{j,u}^{UL}(t)}\right) \end{aligned} \tag{5.3}$$

where the nominator of the SNIR, $SNIR_{n,u}^{DL}(t)$ consists of the DL transmission power of the BS n,$P_n^{DL}(t)$ and the DL channel gain $H_{n,u}^{DL}(t)$ between the BS n and the DL user u; the denominator of the $SNIR_{n,u}^{DL}(t)$ composes of the noise power σ^2, iner-cell interference from the neighboring BSs, $\sum_{i=1,i\neq n}^{N}P_i^{DL}(t)H_{i,u}^{DL}(t)$ and the interference from the other users in the system, $\sum_{i=1}^{N}\sum_{j=1,j\neq u}^{U_{ij}} P_j^{UL}(t)H_{j,u}^{UL}(t)$.

Similarly, the instantaneous UL rate of a user can be expressed as

$$\begin{aligned} R_{n,u}^{UL}(t) &= \log_2\left(1+SNIR_{n,u}^{UL}(t)\right) \\ &= \log_2\left(1+\frac{P_{n,u}^{UL}(t)H_{n,u}^{UL}(t)}{\sigma^2+P_n^{DL}(t)g+\sum_{i=1,i\neq n}^{N}P_i^{DL}(t)H_{i,u}^{DL}(t)+\sum_{j=1}^{N}\sum_{j=1,j\neq u}^{U_{ij}}P_j^{UL}(t)H_{j,u}^{UL}(t)}\right) \end{aligned} \tag{5.4}$$

where the nominator of the SNIR, $SNIR_{n,u}^{UL}(t)$ consists of the UL transmission power of the users and the channel gain between the user u and the BS n; in the denominator, the first term is the noise power, the second term is the self interference, i.e., the product of the BS's DL power and the self interference cancellation (SIC) gain g, the third term is the inter cell interference among the neighboring BSs and the last term is the UL interference power from the neighboring users.

The goal of the scheduler is to select UEs in UL and DL directions in a way so that the utility in equation (5.1) is maximized. First, it schedules a user (either in UL or in DL) that has the highest utility value. Then it tries to schedule another user such that the achieved utility is larger than it was in the previous step when the first user was scheduled. And it continues to schedule more users until the achieved system utility increases compared to the previous step. The scheduler stops to schedule users once the system utility decreases after scheduling a new user.

5.2.1 Selecting users

The goal of the user selection process is to select users in an optimal manner such that it maximizes the system performance. In the HD case, as the DL transmission nodes (i.e., the BSs) have known locations, it is possible to accurately compute the interference from neighboring BSs. Hence, it is convenient to estimate the channel gains with respect to the neighboring BSs for each DL UE. And this does not require the information regarding the scheduling decision of the neighboring BSs. For this reason, it is possible to make optimal UE selection decision in the DL. On the contrary, for the UL scheduling case, interference from the neighboring cells is not possible to compute unless the scheduling decisions of the cells are available. For the FD case, as the DL and UL take place simultaneously it is not possible to optimally compute the channel gains in either directions without complete scheduling information of the neighboring BSs. We have implemented two different scheduling algorithms: a centralized scheduler that has global channel state information and a distributed scheduler at each BS that only has the local channel information of its own cell. The scheduling algorithms are described in the following subsections.

5.2.2 Centralized Scheduling

In a C-RAN case, a centralized scheduler can have a global view of the system i.e., it has information about the user distribution, BS power levels, channel information, etc. Thus this centralized scheduler is able to schedule users intelligently to favor FD modes for the BSs which will increase the overall system throughput. The algorithm 5.1 shows the steps followed in the user selection process by a centralized scheduler.

In each scheduling epoch, the vector B contains all the active BSs in the network, the vector W is the weight vector having the size of the number of active users which is initialized to 1. A matrix ϕ is used that will contain the ids of the scheduled users in different scheduling epoch. The scheduler schedules a user either in the UL or in the DL direction depending on the weighted sum rate maximization of the scheduled users. The weight vector W is updated in a way that make sure all the active users are scheduled either in the UL or in the DL direction (Line 7). For each BS, the algorithm finds a DL/UL user (say $\phi(i)$) that has the highest metric calculated by $GetMetric()$ (Line 11), the metric calculation is given in algorithm 5.3. In the very first iteration this selected user is the first scheduled user, hence its metric is assigned as the maximum metric (Line 16). Then the user is added in the scheduled user list and its associated weight is updated such that in the next scheduling iteration it is not selected again (Line 17 to Line 18). In the next iteration, the scheduler runs through all the BSs, and for each

Algorithm 5.1 User Selection: Centralized Scheduling

1: $B \leftarrow \{1, 2, 3, ..., N_B\}$
2: $W \leftarrow 1$, weight vector initialized to 1
3: $\mu \leftarrow$ constant
4: $\nu \leftarrow$ constant
5: $\phi \leftarrow \{\phi(t_1), \phi(t_2), ..., \phi(t_T)\}$
6: $M_{max} \leftarrow 0$
7: **while** $W > \mu$ **do**
8:     **for** $b \leftarrow B(1)$ to $B(N_B)$ **do**
9:         **for** $p \leftarrow \{\ u, d\}$ **do**
10:             **for** $k \leftarrow 1$ to K_p **do**
11:                 $M(k) \leftarrow GetMetric(k, W)$
12:         $\{\ \phi(i), M(b)\} \leftarrow \{\ arg\ max_{d\varepsilon\{u,d\}}\ M\ \}$
13:         **if** $M(b) \leq M_{max}$ **then**
14:             break
15:     **if** $M(b) > M_{max}$ **then**
16:         $M_{max} \leftarrow M(b)$
17:         $\phi(t_i) \leftarrow \phi_k(i)$
18:         $W(k) \leftarrow W(k)/\nu$, update the weight so that the user is not scheduled in the next iteration

unscheduled user, it calculates the metric for that user and the user scheduled in the previous scheduling epoch. If the new metric ($M(b)$) is higher than the highest metric in the last run (M_{max}), the new user is selected and included in the schedule list $\phi(t_i)$. The corresponding weight of the newly selected user is updated (Line 16 to Line 18) and the whole process runs again until all the users are scheduled. This gives a bunch of schedules of users.

5.2.3 Distributed scheduling

Algorithm 5.2 shows the distributed scheduling algorithm where each BS individually takes the scheduling decision of the UEs connected to it without being concerned of the neighboring BSs. The scheduling is run for a number of transmit time intervals TTI_{tar}. Like the centralized scheduling case a weight vector W is initialized to ensure all the users are scheduled. The average user rate R_{Φ}^{avg} is initialized to a arbitrary small value that is used to update the W in each TTI. At each TTI, each BS schedules users either in the UL or the DL direction from all the

Algorithm 5.2 User Selection: Distributed Scheduling

1: $W \leftarrow 1$; weight vector initialized to 1
2: $\nu \leftarrow 0.95$
3: $R_{\Phi}^{avg} \leftarrow R_{th}$
4: **for** $TTI \leftarrow 1$ to TTI_{tar} **do**
5: $\quad \Phi \leftarrow \{\phi(1), \phi(2), ..., \phi(B)\}$
6: $\quad$ **for** $bs \leftarrow 1$ to $N(B)$ **do**
7: $\qquad W_{bs} \leftarrow W$ *(UEs of bs)*
8: $\qquad M_{max}^{bs} \leftarrow 0$
9: $\qquad$ **for** $p \leftarrow \{ u, d \}$ **do**
10: $\qquad\quad$ **for** $k \leftarrow 1$ to K_p^{bs} **do**
11: $\qquad\qquad M_{bs}(k) \leftarrow GetMetric(k, W_{bs})$
12: $\qquad \{ \phi(bs), M(b) \} \leftarrow \{ arg\ max_{d\varepsilon\{u,d\}}\ M_{bs} \}$
13: $\qquad$ **if** $M(b) \leq M_{max}^{bs}$ **then**
14: $\qquad\quad$ break
15: $\qquad$ **else**
16: $\qquad\quad M_{max}^{bs} \leftarrow M(b)$
17: $\qquad \Phi(bs) \leftarrow \Phi(bs) + \phi(bs)$
18: $\quad R_{\Phi} = GetRate(\Phi)$
19: $\quad R_{\Phi}^{avg} = R_{\Phi}^{avg}\nu + (1-\nu)R_{\Phi}$
20: $\quad W = 1/R_{\Phi}^{avg}$
21: $R_{final} = R_{\Phi}^{avg}/TTI_{tar}$

users connected to it. ϕ (Line 18) contains the scheduled users of all the BSs for that particular TTI. A weighted sum rate maximization technique is used to select a certain group of users for transmission. For each BS, the UEs connected to the BS are scheduled based on the maximization of the metric (sum of their weighted achievable rate) (Line 7 to 17). Depending on the metric maximization, a BS can at most schedule two users, one in the UL and the other in the DL or it will schedule only one UE either in the UL or in the DL direction. After iteration over one BS, the scheduled users for a particular BS $\phi(bs)$, is added to the global schedule list Φ (Line 17). Once, the UEs for all the BSs are scheduled for a particular TTI, their achievable rate R_{Φ} is calculated from the $GetRate(\Phi)$ function (Line 18). The average UEs' rate is then updated using a sliding window method with a window value of ν (Line 19). The weight vector is then updated as inverse proportional of the UEs' rate, this is to make sure that the UEs that were not scheduled at the current TTI get higher priority to be scheduled in the subsequent one.

At the end, the final UEs' rate is the average rate over all the TTIs (Line 21).

Calculation of users' metric in a certain scheduling instant is shown in algorithm 5.3. A channel matrix consisting of channel information from all the transmitters (i.e., the active cells and users) to all the receivers (i.e., the active cells and users) is calculated (Line 3). Then for each user, channels are calculated from all the transmitters to it (Line 7). This encompasses all the interfering signals i.e., for a UL user, interference from the neighboring BSs, interference from intra-cell and inter-cell UL users. For a DL user the interference signal encompasses DL signals from neighboring BSs, UL signals from intra-cell and inter-cell users. Then the user rate is calculated from the signal-to-noise-and-interference-ratio (SNIR) (Line 9 to 10). Finally, the metric for the scheduled users are calculated by multiplying the users' rate with their corresponding weights (Line 12).

Algorithm 5.3 GetMetric(active users, W)

1: $S_{tx} \leftarrow$ Set of active transmitters
2: $S_{tx} \leftarrow$ Set of active receivers
3: $H(S_{Rx}, S_{tx}) \leftarrow$ Channel matrix for all Tx and Rx
4: $N \leftarrow$ Noise power
5: **for** $Rx \leftarrow S_{Rx}$ **do**
6: $\quad h_{Rx} \leftarrow (Rx, S_{Tx}(Tx_{Rx}))$
7: $\quad h_{Rx-int} \leftarrow H(S_{Rx}(Rx), S_{Tx})$
8: $\quad R_{Rx-int} \leftarrow h_{Rx-int} h'_{Rx-int}$
9: $\quad SIR \leftarrow h_{Rx} h'_{Rx} / R_{Rx-int}$
10: $\quad SNIR \leftarrow SIR/N$
11: $\quad R_{Rx} \leftarrow min(log_2(1 + SNIR), 6)$
12: $metric \leftarrow \sum(R_{rx}W)$

System-Level Performance Analysis of a Multi-Cell Full Duplex System

We have studied two different cellular deployment scenarios: one is a dense urban multi-cell heterogeneous network and the other is a multi-cell homogeneous network. For the dense urban model we have studied the Madrid grid model developed by the Metis (Agyapong and et al.,

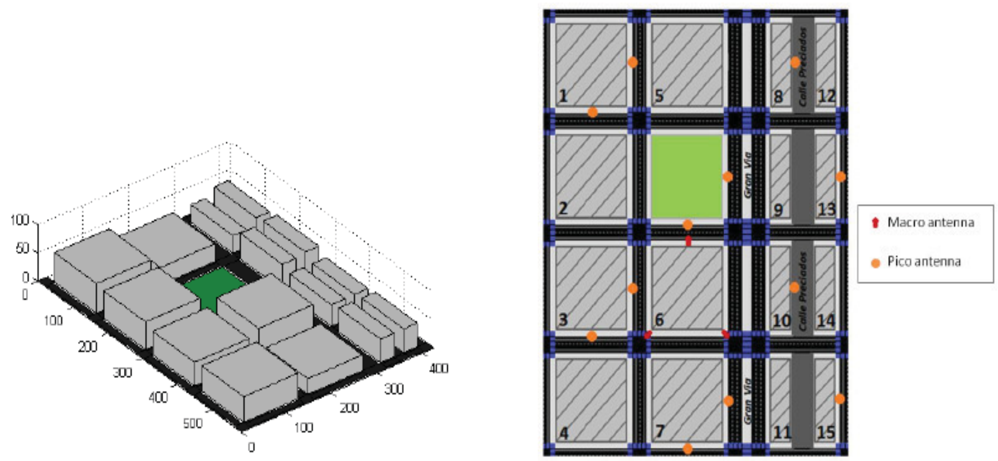
2013) project that consists of macro and pico cells. For the homogeneous model, we have considered a hexagonal grid of cellular cells consisting of macro BSs.

5.3 Dense Urban Model: Madrid Grid (MG)

The urban environment of the Madrid grid model is shown in Fig. 5.3 (Agyapong and et al., 2013). The building layout of the grid can be seen in Fig 5.3a, it consists of building (with entrances) that has different dimension and heights, roads, bus stops, park, side walk and crossing lanes. This model captures the typical propagation environment of a modern city. Fig 5.3b shows the layout of the buildings with BS placements. In this model each macro BS has three sectors and there are 12 pico BSs per macro BS. The red arrow shows the locations of the macro antennas and the orange dots represents the pico BS locations. The antenna radiation pattern of the macro and pico BSs are shown in Fig. 5.4. It can be observed that the radiation beam of antenna-1 of the macro BS (Fig 5.4a) is very wide, this is because there is an open ground in front of the antenna-1 (between buildings 5 and 6 in Fig 5.3b). For this reason, the antenna beam propagates without any obstacle. But the antenna-2 and antenna-3 are placed at the intersection of roads that are surrounded by buildings (cf. at building 6 in Fig. 5.3b), hence their radiation beams are quite narrow (cf. Fig. 5.4b and Fig. 5.4c). Fig. 5.4d shows the radiation of pattern of a pico antenna. The simulation parameters for the Madrid grid model is listed in Table 5.1.

Table 5.1 Simulation parameters for MG model

Parameter	**Value**	**Unit**
Sector/Macro	3	-
Number of Picos/Macro BS	12	-
Maximum Macro BS power	43	dBm
Maximum Pico BS power	24	dBm
Maximum UE power	23	dBm
Thermal noise density	-174	dBm/Hz
Transmission mode	SISO	-



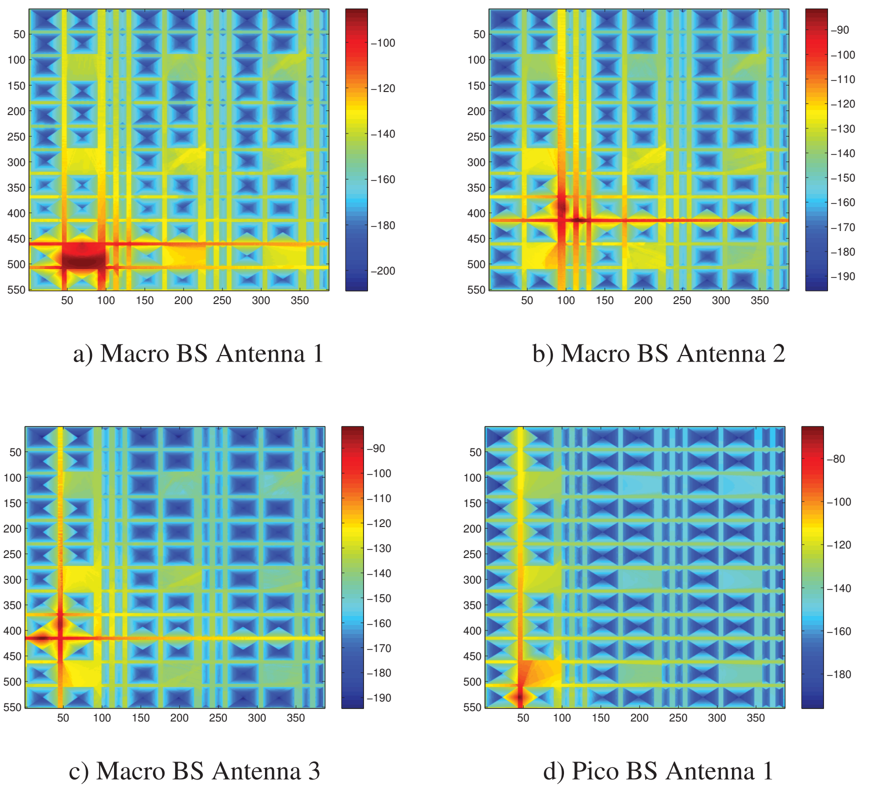
a) Building layout b) Grid layout

Figure 5.3 Madrid grid model layout

5.3.1 Result analysis

The propagation model for the Madrid grid considers channels between BS to UE, BS to BS as well as the UE to UE. The model also considers outdoor-to-indoor and indoor-to-outdoor propagation. It also considers the fully indoor propagation basically for indoor UE-to-UE channel. In the simulation, UEs were dropped in the considered simulation area on an average 10 UEs/BS. The UEs connect to a BS depending on the maximum received signal strength (i.e., encountered lower pathloss). The UEs then were scheduled for transmission by the scheduler(s).

For the centralized scheduling case, the scheduler selected users belonging to the active cells (i.e., the BSs that have connected users) based on their achievable data rate. For the HD case, one user per active cell is selected. For the FD case, for each active cell, the scheduler schedules a user for transmission either in the UL or in the DL. Based on algorithm 5.1 it schedules another user in the opposite direction. It should be noted that, if the inclusion of a user is not favorable to the resulting system performance, the scheduler might schedule only one user either in the UL or in the DL or it might not schedule any user at all for that particular scheduling



a) Macro BS Antenna 1

b) Macro BS Antenna 2

c) Macro BS Antenna 3

d) Pico BS Antenna 1

Figure 5.4 Madrid grid: antenna radiation pattern for Macro and Pico BSs

epoch. Hence, some of the FD BSs might fall back to the default HD operating mode.

For the distributed scheduling case, each BS makes its own scheduling decision regarding its connected user without being concerned about the scheduling decision of its neighboring BSs. Like the centralized scheduling case, for the HD system, each BS assigns a user either in the UL or in the DL. For the FD system, each BS schedules one user at the UL and the other at the DL as long as its cell metric increases. Otherwise, it might schedule only one user at any direction. To investigate the impact of SIC in FD system performance, a range of SIC value is used. Three different interference cancellation modes have been studied: a) the case when only SI is canceled by SIC value, b) the case when in addition to the SIC an inter BS interference cancellation *IBIC* similar to the SIC is employed and c) the case when a SIC and a fixed *IBIC* of 30 dB is used. In the following subsections, we observe the impact of SIC and *IBIC* on achievable user rate and network fairness.

5.3.1.1 User rate vs. SIC

Fig. 5.5 shows the geometric mean of user rate against various SIC values for different interference cancellation scenarios. The centralized scheduling cases have been drawn in solid lines whereas distributed scheduling cases are drawn using dashed line. For all of the centralized and distributed scheduling plots the same approach has been adopted. It is evident from the figure that the centralized scheduling has much better performance in terms of achievable user rate than the distributed scheduling. This is intuitive because the centralized scheduler has a global view of the network and hence, can better schedule the users that minimizes the system interference level thereby increasing the achievable user rate.

For the centralized scheduling case, FD has better performance than the TDD. When only SIC is applied (the green solid line), the system performance is unaffected up until SIC = 82 dB, after that it increases gradually with increasing SIC and saturates when SIC = 160 dB. This shows that increasing the SIC after a certain threshold value does not provide much performance gain. In fact, when only SIC is applied, and after SIC = 140 dB, no significant performance improvement is observed. It is interesting to note that applying a constant inter BS *IBIC* of 30 dB (the red solid line) gives a significant performance boost of almost 50%. Again until the SIC = 82 dB, the system performance remains constant after that it starts to increase and saturates at around 160 dB. Now, to observe the impact of *IBIC*, the interference among BSs is canceled by the same amount as the SIC (the solid blue line). It is interesting to note that, for lower SIC, the system performance increases almost linearly with the *IBIC* until *IBIC* = 48 dB, after that the performance gain is independent of the *IBIC* and it picks up again after SIC = 80 dB with increase in the SIC and saturates at SIC = 160 dB. This shows that, *IBIC* is very significant for performance improvement in a centralized (i.e., a C-RAN) FD system and applying a constant *IBIC* (e.g., 30 dB) is sufficient to achieve a considerable performance gain.

For the distributed scheduling case (the dashed lines), the FD with only SIC (the green dashed

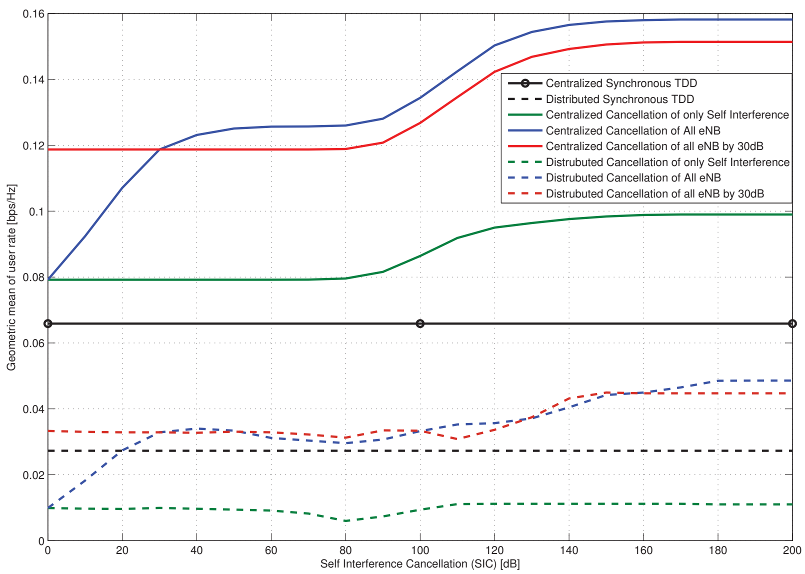


Figure 5.5 Madrid grid model: user rate vs. SIC for HD and FD systems (centralized and distributed scheduling)

line) performs worse than the TDD. There is a performance degradation of almost 68%. If a constant *IBIC* of 30 dB is applied the performance of the system improves, a throughput gain of 20% is observed when only *IBIC* = 30 dB is applied and *SIC* = 0 dB. An almost 65% increase in system performance is achieved when SIC = 148 dB and *IBIC* = 30 dB. Increasing the *IBIC* to an arbitrary amount (dashed blue line) does not provide much performance gain than the fixed *IBIC* of 30 dB. So, it is clear that for the distributed scheduling case (i.e., a D-RAN), applying only SIC while not doing any *IBIC* actually degrades the system performance. A certain amount of *IBIC* (e.g., 30 dB) is needed to observe any reasonable FD performance gain.

Fig. 5.6 shows the CDF of user rate for the centralized and distributed scheduling cases when only SIC is applied. Three different *SIC* values (0, 80 and 120 dB) are used. An *SIC* = 80 dB does not give much performance gain than *SIC* = 0dB. Gain in user rate is seen when *SIC* = 120dB, this shows that a certain threshold of *SIC* is required to achieve FD gain. It is interesting to note that, while the centralized scheduling favors the low rate users which is 70% - 90 % of the users, the distributed scheduling favors the high data rate users. The reason is, as the centralized scheduler has global information about the channel state of all the users it can schedule them intelligently to ensure fairness among them. On the other hand, the distributed scheduler in each BS tries to schedule users connected to it in a greedy fashion in order to maximize its cell throughput which might eventually generate more interference to the neighboring BSs, hence, decreasing the overall system performance.

Fig. 5.7 shows the CDF of users when an additional *IBIC* of 30 dB is applied. It is clear from the figure that the *IBIC* increases the performance of the centralized scheduling by a considerable margin. The *IBIC* boosts the system performance for higher *SIC* values. A 44% gain in user can be used for in 70% of the user when and *SIC* = 120 dB, *IBIC* = 30 dB compared to when only *SIC* = 120 dB is applied.

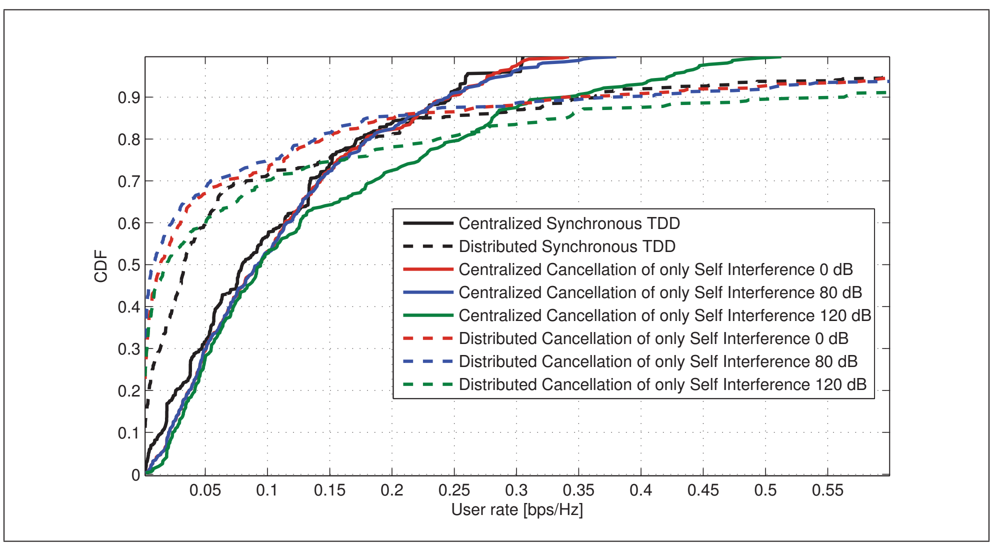


Figure 5.6 Madrid grid model: CDF of user rate when only SIC is employed

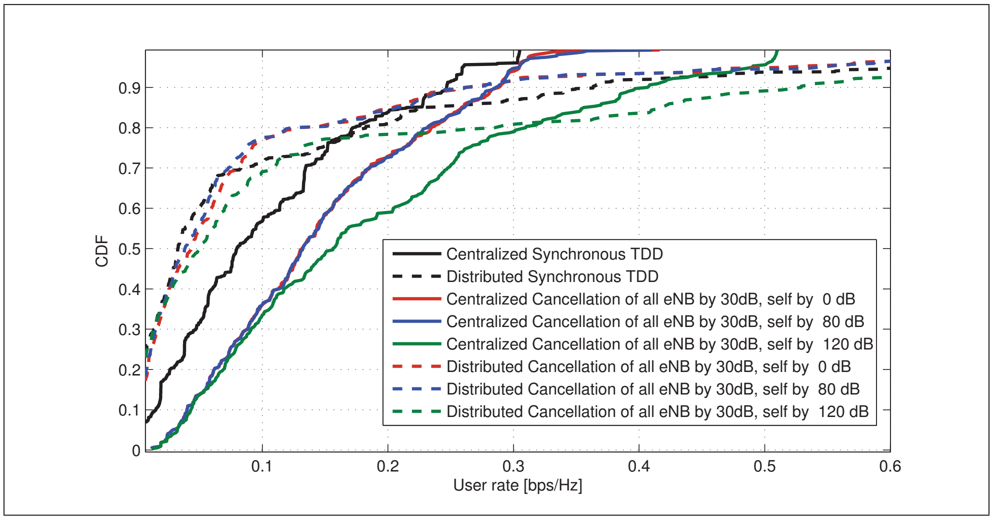


Figure 5.7 Madrid grid model: CDF of user rate when SIC + *IBIC* = 30 dB is employed

5.3.1.2 System throughput vs. fairness

To study the compromise between system throughput and fairness we analyze the sum throughput (ST) against the Jain's Fairness Index (JFI) for centralized and distributed scheduling. To explain the JFI in brief, say for example, the data rates for n users are $r_1, r_2, r_3, ..., r_n$. Then the JFI of the rates can be expressed as

$$J(r_1, r_2, r_3, ..., r_n) = \frac{(\sum_{i=1}^{n} r_i)^2}{n \sum_{i=1}^{n} r_i^2} \tag{5.5}$$

where r_i is the data rate for the i-th user. Table 5.2 lists the throughput vs. JFI performance for the TDD as well as the FD with different cancellation modes. Fig. 5.8 presents a visual representation of the data in the table. It can be observed that while the achieved ST level is similar in both scheduling cases, the centralized scheduling case provides a significant gain in system fairness. Now let us have a closer look at the ST vs. JFI performance of the scheduling models individually.

Fig. 5.9 shows the ST vs. JFI performance for centralized scheduling case. It can be seen that the FD baseline (when SIC = 0 dB and *IBIC* = 0 dB) has 6% throughput gain over the

TDD. Applying a *IBIC* = 30 dB increases the throughput gain by 31.5% while increasing the JFI by a considerable amount. It is to be noted that increasing the SIC to 80 dB does not provide any significant throughput gain. But increasing the SIC to 120 dB provides a further throughput gain of 29.5% with a slight increase in JFI. The FD system with SIC = 120 dB and *IBIC* = 30 dB provides a ST gain of 81% over the TDD.

For the distributed scheduling case (c.f. Fig. 5.10), the FD baseline performance degrades

Table 5.2 Madrid grid: JFI vs. sum throughput

System		**Centralized**		**Distributed**	
		JFI	ST [bps/Hz]	JFI	ST [bps/Hz]
TDD		0.6024	33.9828	0.1460	47.2619
FD (SIC only)	0	0.6374	36.1019	0.1213	47.2633
	80	0.6286	36.4626	0.1185	45.5573
	120	0.5700	46.1049	0.1560	62.6313
FD (SIC + *IBIC* = 30 dB)	0	0.7312	47.4588	0.1372	41.0425
	80	0.7293	47.5508	0.1347	40.1900
	120	0.6619	61.4693	0.1855	61.6282

in fairness while the ST level is almost the same. Increasing the SIC to 120 dB provides a gain of 32% while increasing system fairness at the same time. When an additional *IBIC* = 30 dB is applied the JFI increases significantly with a slight decrease in ST level. The reason is, with additional *IBIC* each BS observes a better channel condition to its UEs and thus greedily schedules its user to maximize the cell throughput. When some of the BSs schedule users at the cell edge, they generate increased interference to their neighbors and as a result the network performance degrades.

5.3.1.3 Node activity

The implemented FD system is a rather hybrid-FD system where BSs operate in FD mode opportunistically when the channel conditions are favorable for FD operation. It would be interesting to see what percentage of transmission frames operate in FD mode. Fig. 5.11

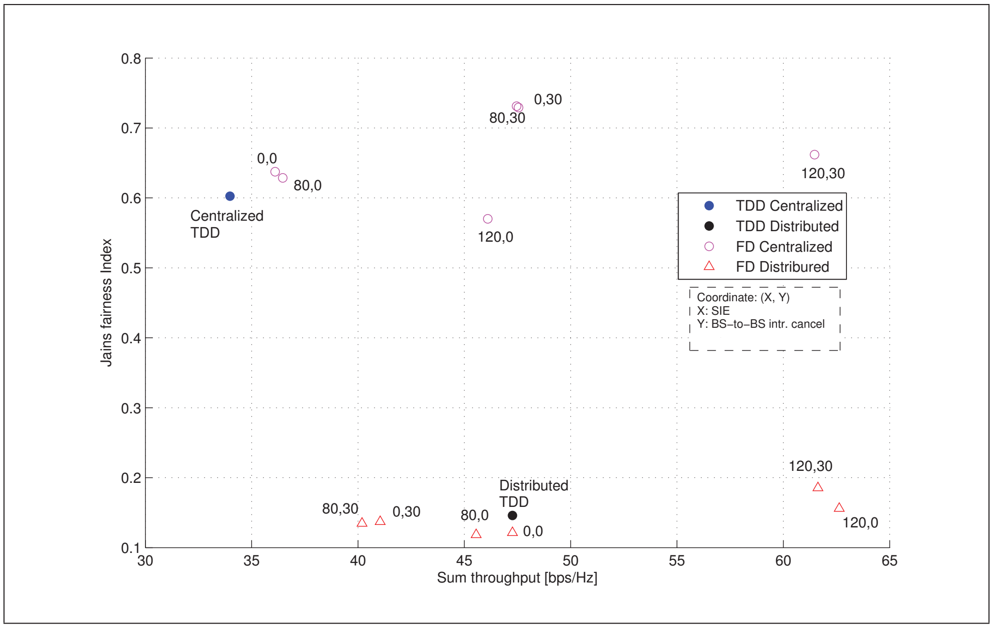


Figure 5.8 Madrid grid model: centralized and distributed scheduling sum throughput vs JFI

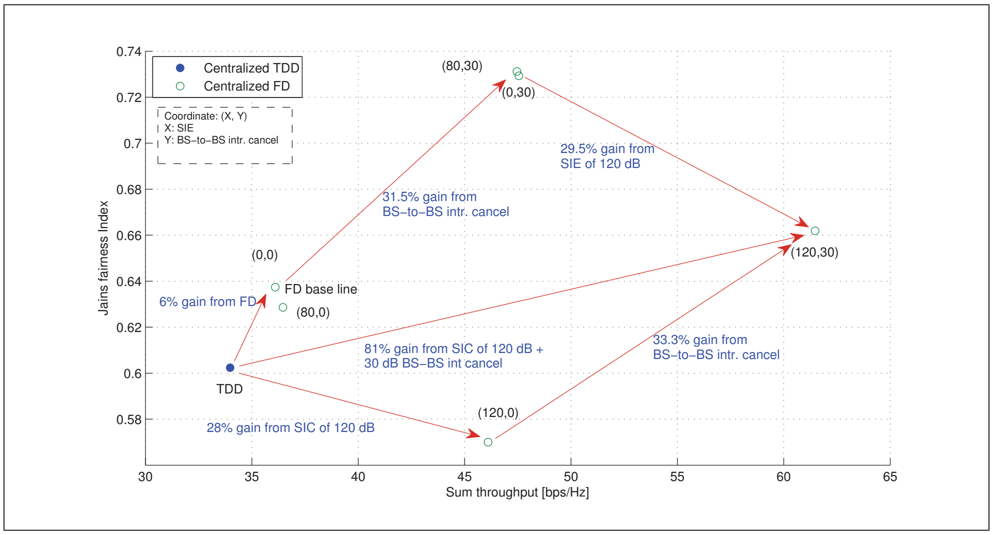


Figure 5.9 Madrid grid model: centralized scheduling sum throughput vs JFI

shows the percentage of frames with FD nodes against the SIC values. It is observed that more frames operate in FD mode as the SIC increases. When SIC = 82 dB and inter-BS interference is canceled by 30 dB, almost 50 % of the transmission frames operate in FD mode and almost

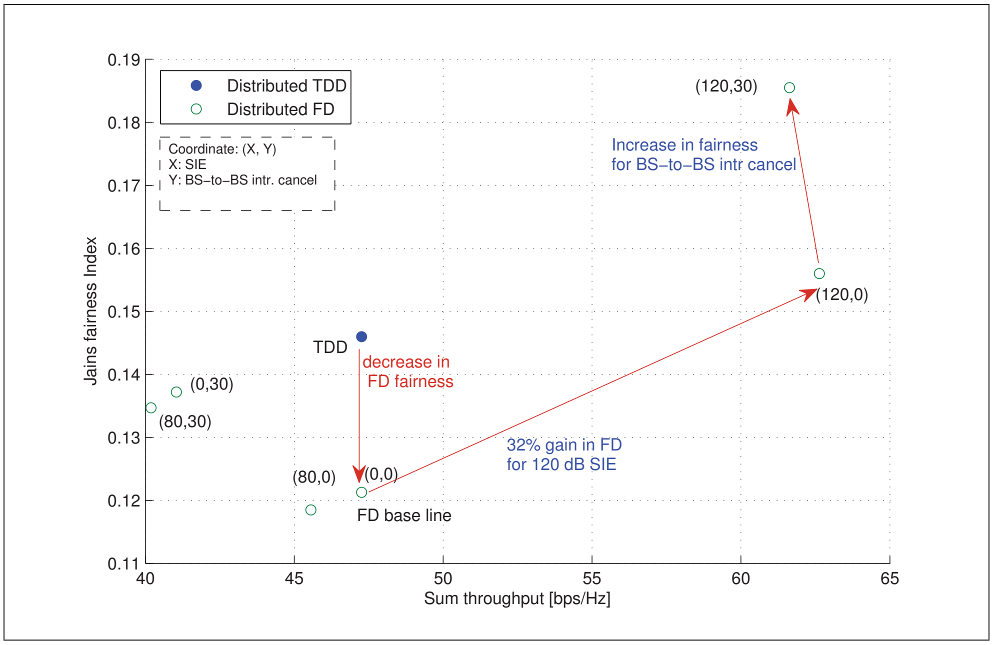


Figure 5.10 Madrid grid model: distributed scheduling sum throughput vs JFI

90 % of the frames are in FD mode when SIC = 110 dB. It is interesting to note that, even when SI is canceled the FD percentage of frames is quite high.

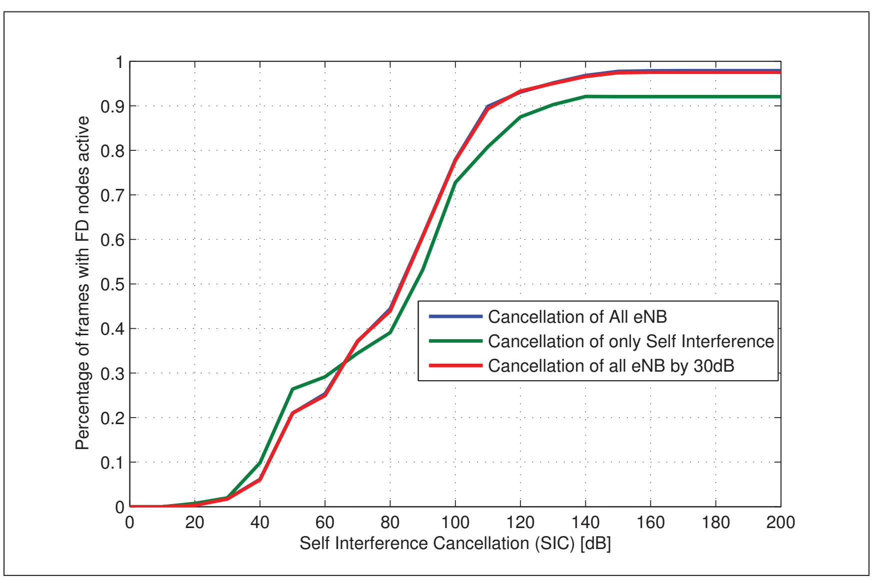


Figure 5.11 Madrid grid model: FD node activity

5.4 Hexagonal grid (HG) model

To study the FD performance for macro-only deployment, we have simulated the hexagonal grid model shown in Fig. 5.12. It consists of 19 macro BSs each having 3 antennas, hence, there is a total of 57 cells in the grid. Each BS is 500 meters apart and the antenna radiation pattern has been considered to be symmetric about the boresight. The simulation parameters are listed in Table-5.4. It is important to note that unlike the Madrid grid model the antennas of the macro BSs are co-located and hence, do not have signal isolation due to antenna placement. For this reason, it is critically important to cancel the interference among co-located antennas of the BSs in order to operate the BSs in FD mode. The impact of the co-located BS interference will be more clear from the result analysis in this section. The pathloss from BS to UE was calculated according to equation (5.6).

$$PL_{BS2UE} = 15.3 + 37.6 log_{10}(D_{BS2UE}) + S_{\sigma} \tag{5.6}$$

where D_{BS2UE} is the distance between the BS and the UE and S_{σ} is the shadowing correlation which is expressed as $S_{\sigma} = \sigma x$, here σ is the shadowing coefficient and x is a random variation. To calculate the interference among the BSs, we used the pathloss between macro BSs in the Madrid grid model using curve fitting as shown in Fig. 5.13. The resulting macro-to-macro pathloss formulation is

$$PL_{BS2BS} = -37 D_{BS2BS} - 44 + 12.1835 \varepsilon \tag{5.7}$$

where D_{BS2BS} is the distance among neighboring BSs and ε introduces a random variation to the pathloss. As mentioned earlier, in FD systems, UE-to-UE interference is also a significant source of performance degradation. To calculate the interference among neighboring UEs we have used the Winner II channel model (Kyösti, 2007). As shown in equation (5.8) three cases have been considered, when the distance between the UEs $d < 5\ m$ they are considered to be in line of sight (LOS). When the distance between two user $5\ m < d < 100\ m$, the users are considered to be in non line of sight (NLOS). Finally, if the distance between two UEs, $d > 100$

m we assume that their signals do not interfere with each other due to lower transmission power of the users.

$$PL_{UE2UE} = \begin{cases} A1_{los} log_{10}(d) + A2_{los} + A3_{los} log_{10}(f_c/5) + x + \sigma_{los},\ for\ d < 5\ m \\ A1_{nlos} log_{10}(d) + A2_{nlos} + A3_{nlos} log_{10}(f_c/5) + x + \sigma_{nlos},\ for\ 5\ m < d < 100\ m \\ \text{inf},\ for\ d > 100\ m \end{cases} \tag{5.8}$$

The parameter values are given in Table 5.3.

Table 5.3 UE to UE pathloss parameter

Parameter	**A1**	**A2**	**A3**	f_c **[GHz]**	σ **[dB]**
LOS	18.7	46.8	20	2.4	3
NLOS	36.8	43.8	20	2.4	6

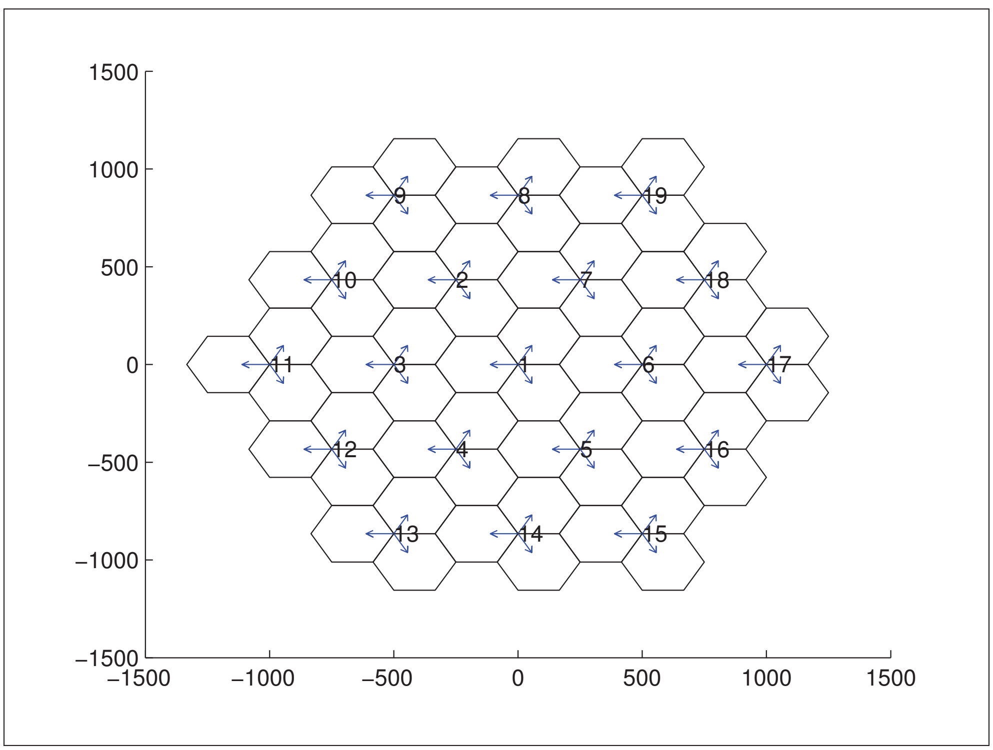


Figure 5.12 Hexagonal grid model

Table 5.4 Simulation parameter for HG

Parameter	Value	Unit
No. of macro BSs	19	-
Sectors/BS	3	-
Max. BS Tx power	43	dBm
Max. UE power	23	dBm
Thermal noise density	-174	dBm/Hz
Shadowing loss	8	dB
BS noise figure	5	dB
UE noise figure	7	dB
BS antenna gain	15	dB
BS antenna height	32	meters
UE antenna height	1.5	meters

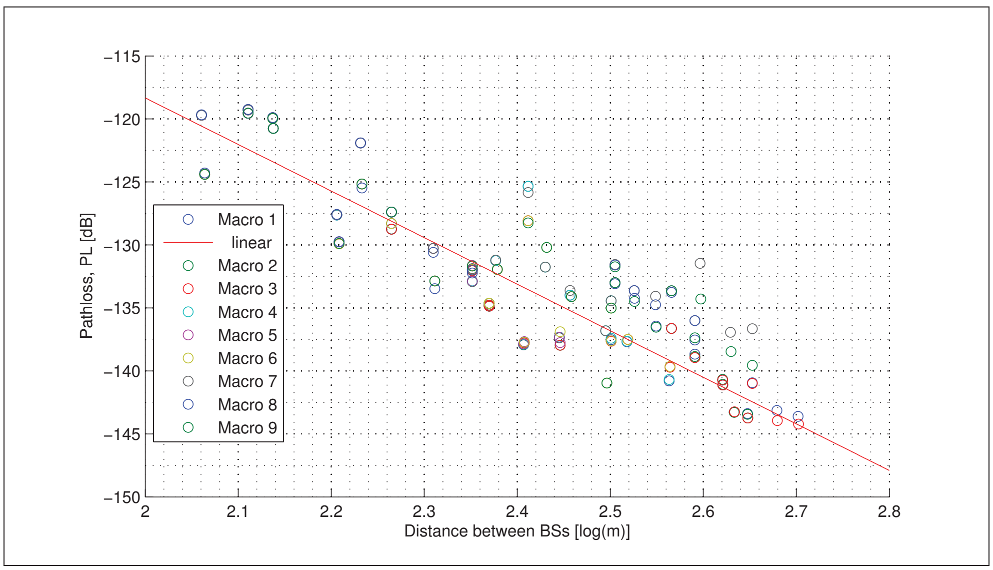


Figure 5.13 Pathloss curve fitting from Madrid grid model

5.4.1 Result analysis

5.4.1.1 User rate vs. SIC

Fig. 5.14 shows the geometric mean of user rate vs. SIC performance when co-located BS interference is canceled by 80 dB. It is evident that the centralized scheduling has much better performance than the distributed scheduling. In case of the centralized scheduling, the FD system has a scheduling gain of 60% even without canceling any self interference. For only cancelling the SIC case (green solid line), the achievable user rate is constant until SIC = 82 dB after that it increases gradually until SIC = 140 dB and then the gain is kept constant. This signifies that there is a threshold value of SIC (around 80 dB) below which only scheduling gain is visible and beyond this threshold value a gain due to increased SIC is observed. Adding an extra *IBIC* = 30 dB (solid red line) improves the throughput gain by 9.5% for lower SIC range (until SIC = 82 dB) and the gain increases up to 21% for larger range of SIC (>82 dB). Increasing the *IBIC* (solid blue line) further improves the user rate and fairness.

For the distributed scheduling case (denoted by the dashed lines), FD system with only SIC (dashed green line) has degraded performance than the TDD counterpart. Employing *IBIC* improves the system performance (dashed red and green lines) after a certain SIC threshold (82 dB). Beyond that threshold, the FD system performance is inferior to the TDD system. Hence, to achieve FD gain in distributed system, a certain SIC threshold as well as *IBIC* cancellation are necessary.

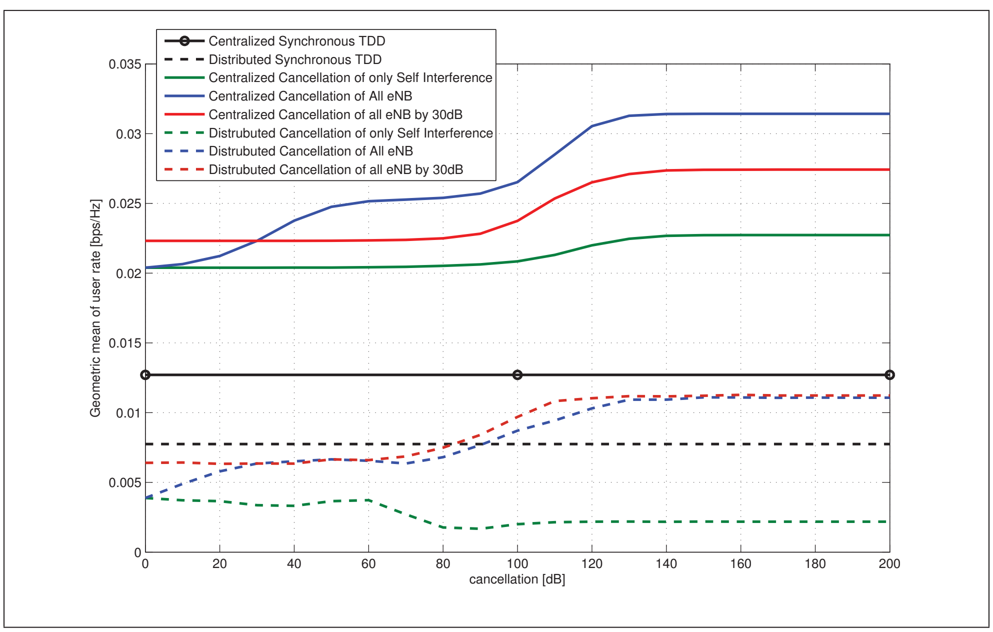


Figure 5.14 Hexagonal grid: geometric mean of user rate vs. SIC

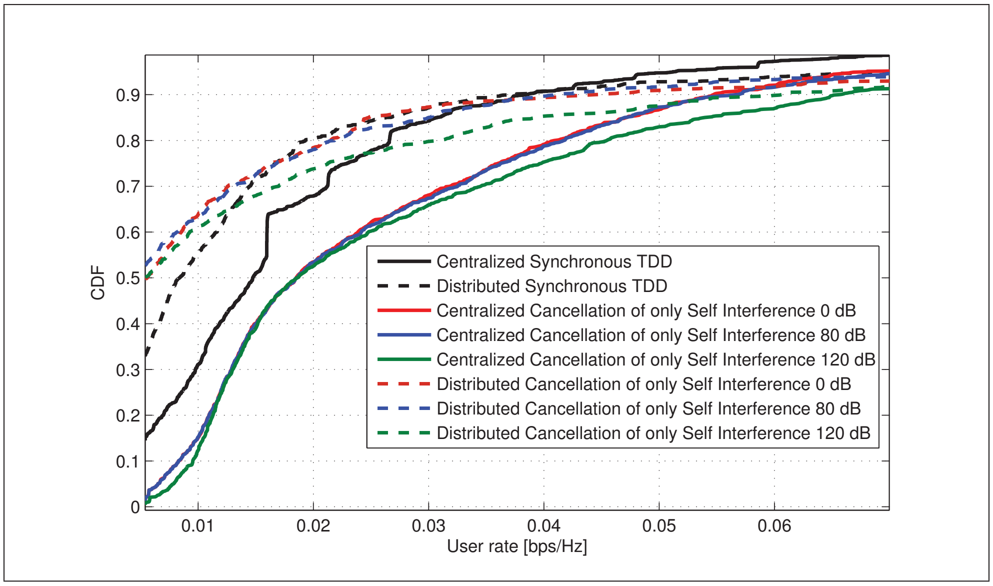


Figure 5.15 Hexagonal grid: CDF of user rate when only SIC is applied

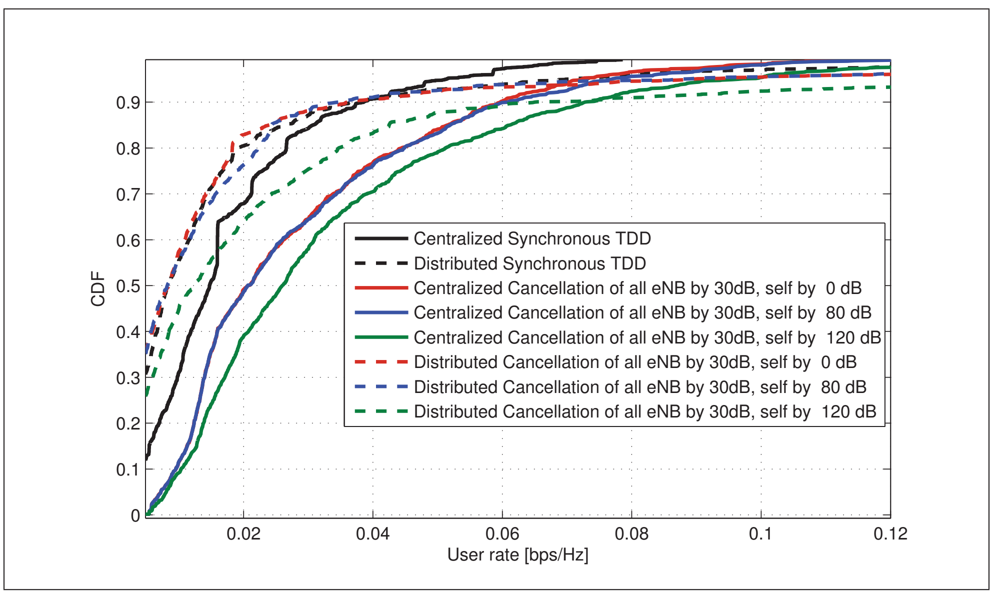


Figure 5.16 Hexagonal grid: CDF of user rate when SIC and a *IBIC* = 30 dB is applied

5.4.1.2 System throughput vs fairness

System throughput vs. JFI performance for centralized and distributed scheduling is presented in Table 5.5 when co-located BS interference is canceled by 80 dB. The same performance metric is presented in Table 5.6 when interference among co-located BSs is canceled by 120 dB. Fig. 5.17 represents a visual delineation of Table 5.5. It can be observed that the centralized scheduling provides much better fairness than the distributed scheduling for comparable ST. It is to be noted that for higher SIC and *IBIC* (120, 30) the distributed scheduling provides significant ST gain over the centralized scheduling but off course at the cost of huge reduction in system fairness. In distributed scheduling case, each BS makes their scheduling decision independent of the scheduling decisions of its neighbours. If a BS schedules a UE at its cell edge, the DL transmission of the BS to the UE generates increased interference to its neighbouring BSs which in turn, reduces the global throughput level. Next we have a closer look at centralized and distributed scheduling performances individually.

Table 5.5 Hexagonal grid: JFI vs. sum throughput (co-located antenna interference cancelled by 80 dB)

System		**Centralized**		**Distributed**	
		JFI	ST [bps/Hz]	JFI	ST [bps/Hz]
TDD		0.5807	21.4512	0.0512	28.0169
FD (SIC only)	0	0.6094	30.3458	0.0582	31.2844
	80	0.6024	30.7005	0.0646	26.7055
	120	0.5590	34.0769	0.0622	43.8472
FD (SIC + *IBIC* = 30 dB)	0	0.6251	30.8768	0.0427	34.8174
	80	0.6124	33.4717	0.0446	34.7253
	120	0.5974	39.9093	0.0758	50.2231

Fig. 5.18 shows the JFI vs. ST for centralized scheduling. The points in the plots are identified by 3-coordinates, the first one refers to the co-located BS interference cancellation, the second one denotes the SIC and the third one represents *IBIC*. It is seen that, the FD base line (point

Table 5.6 Hexagonal grid: JFI vs. sum throughput (co-located antenna interference cancelled by 120 dB)

System		Centralized		Distributed	
		JFI	ST [bps/Hz]	JFI	ST [bps/Hz]
TDD		0.5775	20.9743	0.0696	27.8332
FD (SIC only)	0	0.6784	33.3011	0.0715	25.2011
	80	0.6617	33.9119	0.0761	25.1975
	120	0.6697	41.0527	0.1143	33.2600
FD (SIC + *IBIC* = 30 dB)	0	0.6986	34.3173	0.0639	21.9808
	80	0.6809	34.9447	0.0701	22.0570
	120	0.6921	42.2761	0.1053	28.7944

(0,0,0)) has 32.4% loss in ST than the TDD though it has a higher JFI. Applying a co-located BS interference cancellation of 80 dB increases the ST of the FD system by 2.11 fold. Additional gain in ST and JFI is observed by a *IBIC* of 30 dB. An SIC = 120 dB further increases the ST gain by an additional 21.4%. A SIC = 120 dB and *IBIC* = 30 dB provides 86% FD gain compared to the TDD case. Increasing the co-located BS interference cancellation to 120 dB (the point (120,120,30)) further increases the ST and JFI gain.

The distributed scheduling performance for JFI vs. ST is shown if Fig. 5.19. It can be seen that the FD baseline has a 12% loss in ST in addition to a decrease in JFI when compared to the TDD case. Employing a co-located BS interference cancellation of 80 dB provides a ST gain of 26.7%. A further 40% gain in ST is achieved from a SIC of 120 dB. The combined impact of SIC = 120 dB and *IBIC* = 30 dB provides a ST gain of 79.3% with a significant improvement in JFI when compared to the TDD case.

5.4.1.3 Node activity

Fig. 5.20 shows the percentage of FD frame with SIC when co-located BS interference is canceled by 80 dB. It is observed that for low SIC ($\leq$ 50 dB) the percentage of FD frame is quite low (below 30%). If only SIC (green line) is applied, 50% of the frames operate in FD mode when SIC = 100 dB. But as SIC is increased ($\geq$ 130 dB) the FD frames does not increase

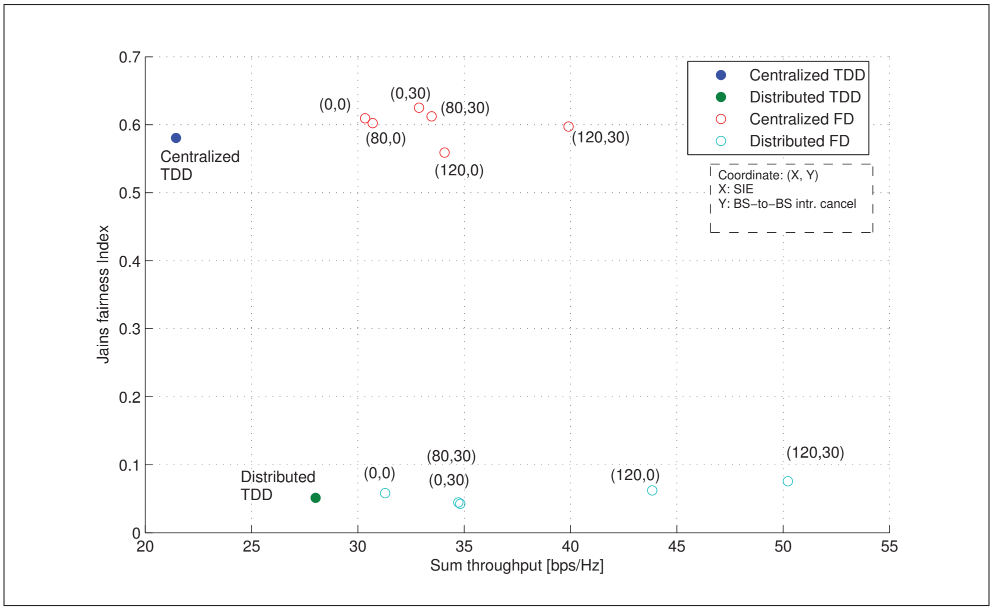


Figure 5.17 Hexagonal grid: centralized and distributed scheduling sum throughput vs. JFI

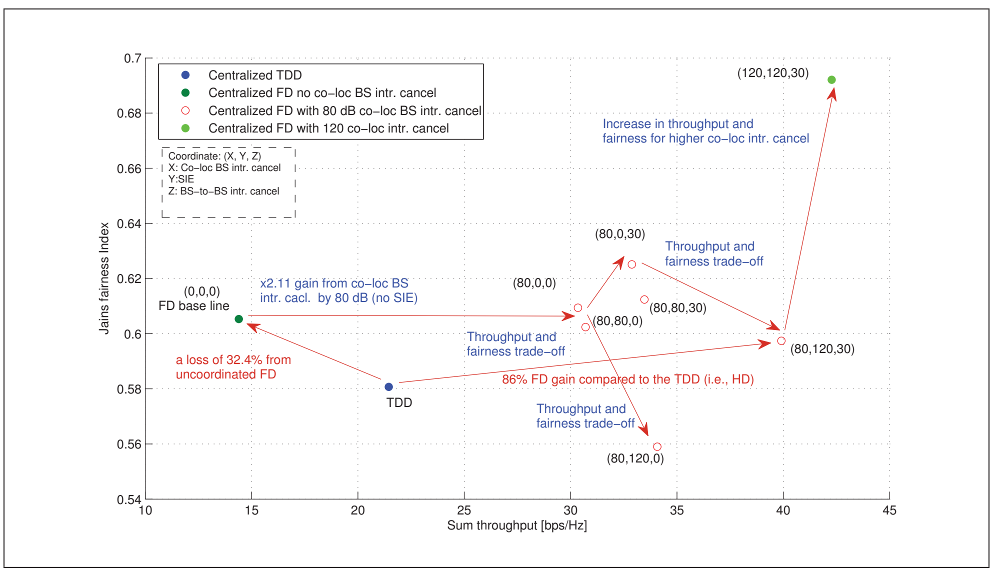


Figure 5.18 Hexagonal grid: centralized scheduling sum throughput vs. JFI

beyond 70%. The reason is though SI is canceled by the high SIC values, the interference from co-located BSs acts as the bottleneck for further increase in FD operation. When additional

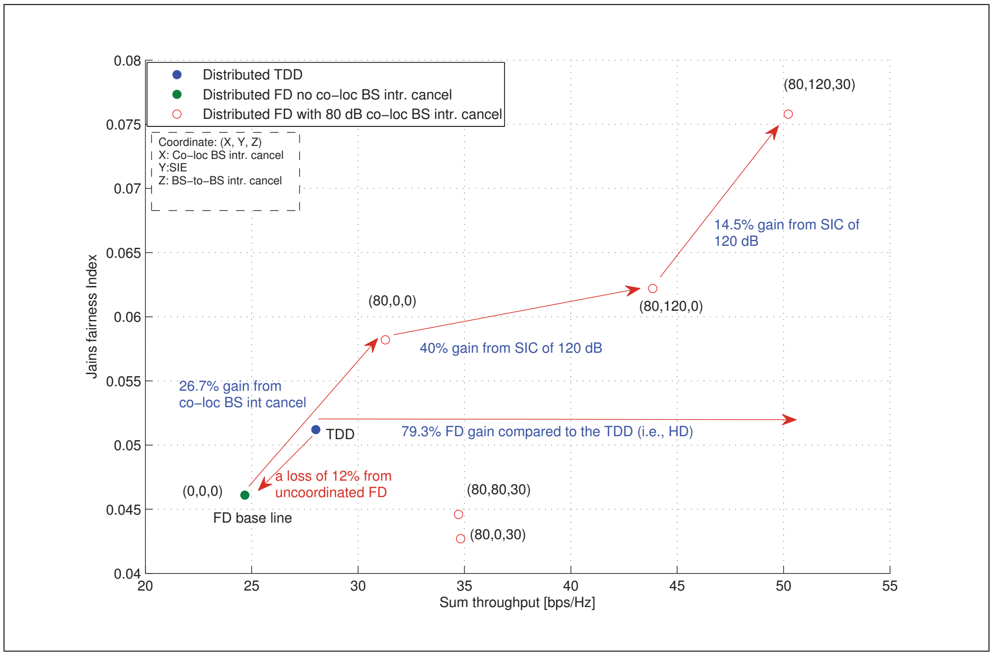


Figure 5.19 Hexagonal grid: distributed scheduling sum throughput vs. JFI

inter-BS interference cancellation of 30 dB is applied, the FD frame percentage grows to as much as 90%.

5.5 Conclusions

In this chapter, multi-cell performance of FD cellular systems has been investigated. Two deployment scenarios have been considered, namely a dense urban multi-tier network with macro & pico BSs and a homogeneous multi-cell network consisting of only macro BSs. The deployments are investigated for both C-RAN and traditional D-RAN models. Considering FD BSs and HD UEs, scheduling algorithms are presented that use joint user selection and binary power control mechanisms. It is observed that for successful implementation of a FD network, BSs need to be aware regarding the scheduling and power control decision of the neighboring BSs. For this reason, in each of the investigated network models the centralized scheduling model has much better performance than the distributed scheduling model.

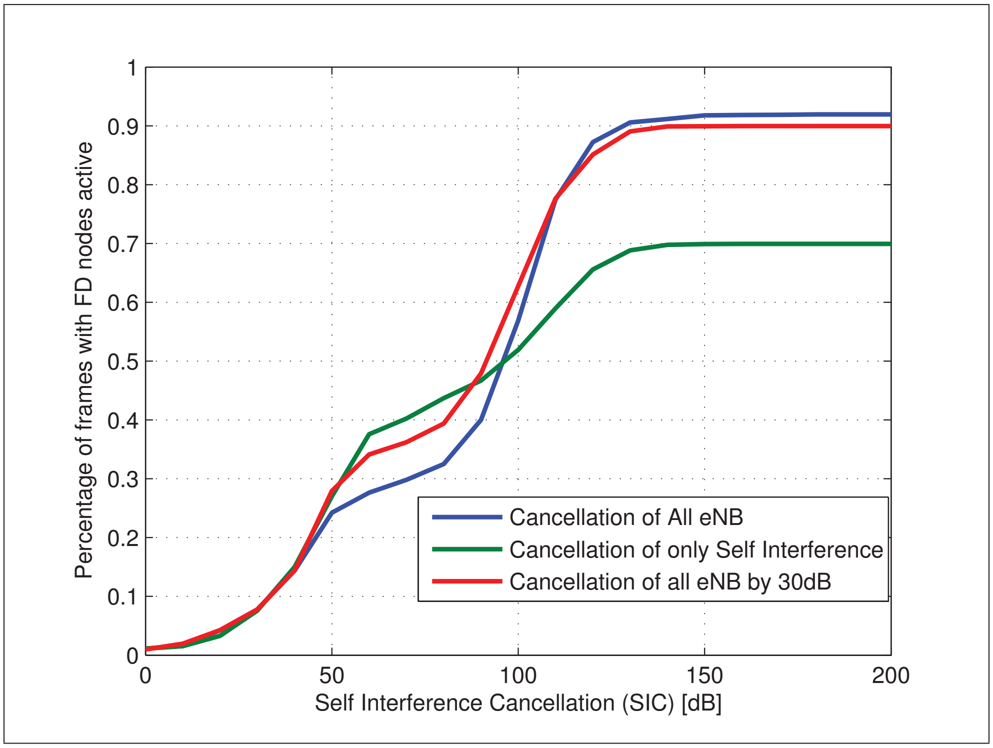


Figure 5.20 Hexagonal grid model: FD node activity

For the multi-tier dense urban model, canceling inter BS interference is very significant. For the centralized scheduling case, only employing a SIC = 120 dB gives a throughput gain of 28% whereas adding an additional 30 dB inter-BS interference can give a throughput gain of 81%. The distributed scheduling case gives a FD throughput gain of 32%, an additional *IBIC* though does not give much throughput gain but improves the network fairness performance. For the single tier macro-only deployment, cancellation of interference among co-located BS is crucial. For the centralized scheduling case, removing co-located BS interference by 80 dB results in 2.11 fold increases in system throughput. Adding an additional SIC of 120 dB and *IBIC* of 30 dB gives a throughput gain of 81% compared to the HD counterpart. Increasing the co-located BS interference cancellation to 120 dB provides a further improvement in both system throughput and fairness. For the distributed scheduling case, a throughput gain of 79.3% is attained by canceling co-located BS interference by 80 dB and employing SIC = 120 dB and *IBIC* = 30 dB.

Though the FD system promises to provide the performance boost that is highly sought after in the wake of current cellular data explosion, a number of challenging issues need to be resolved for a successful realization of FD networks. Current state of the art FD radio can achieve a SIC of 110 dB but to realize a multi-tier FD cellular network this is not going to be enough, as an additional SIC of 10-20 dB is required as was shown in the system-level simulation results. Intelligent selection of UEs need a higher degree of cooperation among BSs which results in high volume of backhaul traffic. Efficient techniques need to be devised to handle such significant backhaul traffic. In our on going efforts, we are investigating UE-to-UE interference cancellation techniques which is very critical for FD system. We are also extending our work on intelligent backhauling for multi-cell FD networks.

CONCLUSION

In Chapter 2, three frameworks for wireless access network virtualization have been proposed. The first one is a special-purpose hardware-based model, referred to as Locally Virtualized Network (LVN), where a hypervisor is used to slice super base stations (SBSs) to create multiple virtual base stations (VBSs). The second is a data center based model, referred to as Clustered/Remote Virtualized Network (CVN/RVN), where the signal processing gear is pooled in centralized data centers and fiber-distributed RRHs are used to provide radio access to users. A third model, referred to as hybrid virtualized network (HVN), is a proper combination of the aforementioned models (LVN and CVN/RVN) designed to offer the potential to balance network cost and QoS with greater flexibility than the previous two models (LVN and CVN/RVN). The proposed virtualization frameworks are quite different in terms of their network cost and the achievable QoS. Hence, to compare the suitability of the models for specific deployment scenarios, a new multi-criteria utility function has been developed that accounts for network cost & QoS trade-offs to enable the design and optimization of wireless access virtualization architectures that best comply with the investment and service-level requirements of network operators (and/or service providers).

In this chapter, for the analysis TDD mode of operation was considered. The use of TDD requires tight coordination and synchronization among network equipment in the same coverage area. For this reason, in TDD, BSs operating in the same coverage area need to be synchronized with each other within the frame granularity. The switching electronics in the BS and UE need time to toggle between the Tx/Rx modes. To facilitate this operation, a guard period (GP) is allocated in a special subframe to compensate for the switching time and the propagation delay. The GP has to be sufficiently long to accommodate the propagation delay and the hardware switching time to properly enable the DL/UL transition. This GP plays a crucial role in achievable QoS of the virtualization frameworks.

The choice of a certain framework essentially is based on a given compromise between the corresponding network cost and the achievable QoS. The LVN can reduce cost to some extent but its implementation complexity increases due to the pooling of (virtual) network nodes and the introduction of a hypervisor. The CVN/RVN is the most cost-effective solution due to its usage of inexpensive general purpose IT hardware for baseband signal processing. But the inclusion of optical fibers in its network architecture places limitations on the achievable QoS due mainly to additional RTTD for radio transmission over fiber optic cables. The HVN is a more balanced approach to network cost and QoS optimization. The impact of the PHY and the MAC layer parameters on the CPC size has been assessed in this chapter.

The optimal size of a CPC depends on many parameters such as the system bandwidth, the coverage radius of the macro base stations, the network architecture (i.e., whether it is homogeneous or heterogeneous), etc. One of the most critical parameters affecting the CPC size is the GP value of an OFDMA subframe. When the primary concern is QoS (i.e., less emphasis on cost), smaller CPCs should be preferred. But when the operational budget is constrained, network designers should favor relatively larger CPCs with relatively wider coverage areas. A CPC of 1 to 3 km radius in a coverage area of 20 km radius is preferred for a wide range of w_c values. Interestingly, in the extreme case when there is no budget restriction (i.e., $w_c = 1$), the optimal CPC size is with a 10 km radius, meaning that a RVN (i.e., a single CPC covering the whole area) can never be an optimal design choice. It is worth mentioning that MAC layer parameters like GP can be optimized along with the cost-QoS trade-off in a CVN/RVN model.

The CVN has better utility performance than RVN for some GP value. The maximum network utility is achieved with $GP = 4$ symbol periods (when $\alpha = 1.4$) because it balances both the cost and QoS in the most efficient manner. When $GP = 1$ in the RVN case, the network utility is severely penalized because just one symbol period is not large enough to account for radio propagation delays over a fiber distance of 20 km for adequate OFDM DL-UP synchronization. Hence the RVN architecture can never be a favorite choice, because the network's QoS is severely penalized due to the RVN's inability to properly resolve PHY (resolving transmission channel severity issues) and MAC (DL-UL synchronicity) layer issues.

In all the considered scenarios, HVN has the best utility behavior. For lower or higher w_c values, the LVN and the CVN approach ultimately match the HVN in utility performance at either end of the w_c range, respectively, but never outperform it. Acknowledging both facts that HVN offers lower cost than the LVN at lower w_c values and higher QoS than the CVN at higher w_c values, it stands up unambiguously as the best network design choice. The value of w_c is a subjective design choice that depends on a given MVON's/SP's investment constraints and intended services.

Chapter 3 lays out the blueprint of an end-to-end programmable HetNet. To cope with the novel service requirements of future 5G HetNet, it has been argued in the thesis that, programmability should be ensured at each layer of the network architecture. So that, the VNOs can deploy their customized networks that might require change in any later of the network, e.g., novel routing algorithms, application specific charging policies, new signal processing algorithms, etc. Potential business cases and research challenges have also been identified in this chapter.

In this chapter, the convergence of virtualized heterogeneous wireless network infrastructure has been put forth to facilitate abstraction of physical resources, hence paving the way for their efficient utilization. Two key requirements have been identified which future service providers will need; they are programmability and elasticity of their networks that will provide them enough flexibility & control over the network substrate and make them able to scale up/down their network resources to meet customer demands. In this respect, an end-to-end programmable, cloud-based solution for heterogeneous wireless networks called HVWN has been presented. It provides programmability in both network core and access by employing SDN and programmable radio technologies. To meet the service requirements of different kinds of networks, HVWN uses cloud-based resource pools in distributed WDC as well as virtualized APs that use general purpose hardware and in-situ signal processing. VNOs can lease appropriate resources from the InPs to deploy their customized virtual networks. Different layers of the HVWN have been discussed in detail. Business cases for virtual wireless networks as well as the critical research issues and challenges to address in realizing such a programmable virtualized heterogeneous networks have also been identified.

In Chapter 4, provisioning of differentiated service in software-defined heterogeneous wireless networks has been studied. In particular, the implementation viability of NFV using a software-defined paradigm has been investigated in this chapter. It has been shown that in a virtualized wireless HetNet, control layer functionalities e.g., mobility management, load balancing, data offloading, etc. can be implemented has high-level network policies in a software-defined paradigm. This facilitates providing differential services on a common physical substrate which a major goal of future 5G networks. It has been proposed to utilize the *spare bits* of the OpenFlow (McKeown *et al.*, 2008) packet structure to implement virtual fields for identifying virtual network components. Extensive system-level simulation results show that such SDN-based implementation of a virtual wireless HetNet is able to meet the critical performance characteristics of carrier networks. The use of northbound APIs to facilitate provisioning of differentiated services in a virtual wireless HetNet has been advocated in this chapter. Wireless networks need various applications to run simultaneously to achieve full network operability. These applications range from traffic routing, mobility management, resource scheduling, policy enforcement, billing functionalities etc. Domain specific programming languages like Pyretic (Reich *et al.*, 2013) (built on top of POX (Mccauley) controller platform) make building modular network programs an ease. It has been proposed to use northbound API like Pyretic (Reich *et al.*, 2013) to build modular applications for virtual wireless networks. The parallel and sequential composition operators of the language makes it possible to compose complicated network applications by composing (in parallel or in series) more simpler applications. Using the abstract packet model in Pyretic, OpenFlow (McKeown *et al.*, 2008) packet header fields can be extended to include virtual fields, that can be used to associate packets with high level meta data. In (Monsanto *et al.*, 2013), Monsanto et al. gives a comprehensive description of the usage of Pyretic language model.

It has been proposed in this thesis to extend the abstract packet model in Pyretic to implement virtual wireless networks through *abstract topology* (Monsanto *et al.*, 2013). The spare bits (e.g. VLAN, MPLS fields) in an OpenFlow packet (notice Fig. 4.3a) are used for specifying virtual networks, virtual network node and wireless spectrum to be used for transmitting that particular packet. VNOs are identified by a *VNO* id in the virtual field, these ids are unique as VNOs should be uniquely identifiable. Virtual nodes (switches, BSs, APs, middle boxes, etc.) are identified with a *virtual switch (VSW)* id. These ids are unique to a InP but different InPs can use the same VSW id, as it is locally significant. For flexible allocation of radio resources a *Radio Spectrum (RS)* id is used to specify the transmission frequency for a VNO. This gives a great flexibility in being able to do wireless resource allocation on a per packet (per flow) granularity which will facilitate to tackle different radio propagation problems, like interference management, traffic offloading, etc. The use of northbound API in facilitating different network management issues e.g., interference management, traffic offloading have been discussed in detail. Through system-level simulations it has been shown that in a virtual wireless HetNet differentiated services can be provided based on not only the application type but also the subscription level of users. And the SDN-based network architecture is able to meet the critical performance requirement of carrier networks.

Future work

In chapter 2, we analysis is for *green field* deployment of virtualized networks. As future work, we would study the *grey field* deployment of virtual wireless networks, i.e., when virtual networks are deployed gradually replacing the traditional networks. In order to make the analysis tractable, a rather simplified model has been assumed for network performance analysis. In future we shall consider advanced PHY-MAC technologies such as coordinated multi point (CoMP), joint resource scheduling and processing among neighbouring BSs, interference management for a centralized control plane architecture, etc.

In chapter 3 an end-to-end programmable architecture for HVWN has been presented and a simplified version of the model was implemented in chapter 4. In this implementation, wireless channel models of software-defined virtual networks were simulated with very simplified assumption. In future, we will implement a detailed wireless propagation model consisting of large and small scale fading, multipath propagation, interference among BSs, BSs and UEs, etc. This will give a more accurate result on real world deployment of virtualized networks.

As mentioned in chapter 5, FD systems have high capacity backhaul requirements. As future work we will develop efficient coding techniques for enabling high capacity backhaul network. We shall also devise algorithms to mitigate UE-to-UE interference in FD multi-cell networks which will improve the system performance to a significant extent.

To conclude, in this thesis we have three virtualization frameworks. And we have developed a composite utility model that takes into account the virtual frameworks' network CAPEX & OPEX and achievable QoS to compare them (chapter 2). This utility model can serve as guideline for network designer to choose a virtualization framework for a particular deployment model. Based on the results from the analytical modeling it became evident that cloud-based wireless networks will be a significant part of future virtual network deployments. In this regards, we have proposed an end-to-end programmable, cloud-based network architecture for deploying virtual heterogeneous wireless networks on a common physical substrate using SDN and cloud computing technologies (chapter 3). We have implemented a simplified version of the network in Mininet emulation platform and investigated differentiated service provisioning in software-defined virtual heterogeneous networks (chapter 4). Simulation results show that such architecture is able to meet the strict performance requirements of carrier networks. Finally, addressing the importance of FD system to alleviate the spectrum scarcity problem of virtual wireless networks, we have investigated the implementation challenges of multi-cell FD networks in single-tier and multi-tier networks. We have developed novel algorithms for C-RAN and D-RAN deployment of multi-cell networks for successful roll-out of FD cellular networks that enables to harness the doubling spectral efficiency gain of FD system. It has been shown that the algorithms enables to achieve significant performance gain in FD multi-cell networks.

BIBLIOGRAPHY

"Apprenda". https://apprenda.com/. [Online; accessed 12-Nov-2015].

"Amazon Web Services". https://aws.amazon.com/. [Online; accessed 12-Nov-2015].

"Click Modular Router". http://read.cs.ucla.edu/click/click. [Online; accessed 07-Dec-2015].

"Floodlight Controller". *available at: http://www.projectfloodlight.org/floodlight/*.

"iJOIN project, Revised definition of requirements and preliminary definition of the iJOIN architecture". *INFSO-ICT-317941 iJOIN, D5.1, version 1.0*.

"Google Compute Engine (GCE)". https://cloud.google.com/compute/. [Online; accessed 12-Nov-2015].

"iJOIN project, Revised definition of requirements and preliminary definition of the iJOIN architecture". *INFSO-ICT-317941 iJOIN, D5.1, version 1.0*.

"Kinetic: Verifiable Dynamic Control". *available at http://resonance.noise.gatech.edu/*.

"Mininet: An Instant Virtual Network on your Laptop (or other PC)". *available at: http://mininet.org/*.

"Microsoft Azure". https://azure.microsoft.com/en-us/. [Online; accessed 12-Nov-2015].

"OFELIA, A First report on implementation testing and operation of Federated Islands". *Deliverable 2.4, available at: http://www.fp7-ofelia.eu/assets/Public-Deliverables/OELIAD2.4-First-report-on-implementation-testing-and-operation-of-Federated-Islandsfinal.pdf*.

a. "Open Networking Foundation (ONF)". *available at https://www.opennetworking.org/*.

b. "Open Networking Foundation (ONF), Software-Defined Networking: The New Norm for Networks". *white paper, available at https://www.opennetworking.org/*.

a. "OpenAirInterface". http://www.openairinterface.org/.

b. "OpenDaylight project". *available at: http://www.opendaylight.org/*.

c. "OpenvSwitch". http://openvswitch.org/. [Online; accessed 07-Dec-2015].

"SoftCOM: Reshaping the Future of Network Architecture". *Huawei White Paper, available at http://www.huawei.com/ilink/en/solutions*.

2009. "Network Virtualization, New generation network architecture AKARI conceptual design". *version 2.0*.

2011. "C-RAN: The Road Towards Green RAN".

2013. ". *Juniper Networks, Contrail Architecture, White Paper*.

2013. "ETSI, Network Function Virtualization (NFV); Architectural Framework". *ETSI GS NFV 002, version 1.1.1*.

2013. "North Bound Interfaces Working Group (NBI-WG) Charter, version 1.1". *Open Networking Foundation (ONF)*.

3GPP TS 36.211, v.-. "Evolved Universal Terrestrial Radio Acess (E-UTRA); Physical Channels and Modulation (Release 9)".

Agyapong, P. and et al. 2013. "Mobile and wireless communications Enablers for the Twenty-twenty Information Society (METIS)". *Seventh Framework Program, Deliverable D6.1*.

Ahokangas, P., M. Matinmikko, S. Yrjola, H. Okkonen, and T. Casey. 2013. ""Simple rules" for mobile network operators' strategic choices in future cognitive spectrum sharing networks". *IEEE Wireless Communications*, vol. 20, n° 2.

Akyildiz, I. F., W.-Y. Lee, M. C. Vuran, and S. Mohanty. 2006. "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey". *Elsevier Computer Networks*, vol. 50, n° 13.

Akyildiz, I. F., P. W. A. Lee, M. Luo, and W. Chou. 2014. "A roadmap for traffic engineering in SDN-OpenFlow networks". *Elsevir Computer Networks*, vol. 71.

Akyildiz, I. F., P. Wang, and S.-C. Lin. 2015. "SoftAir: A software defined networking architecture for 5G wireless systems". *Elsevier, Computer Networks*, vol. 85, p. 1-18.

Alves, H. and et al. 2014. "On the average spectral efficiency of interference limited full duplex networks". In *Proc. of the 9th International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CROWNCOM)*.

Asadi, A., Q. Wang, and V. Mancuso. 2013. "A survey on device-to-device communication in cellular networks". *CoRR*.

Bansal, M., J. Mehlman, S. Katti, and P. Levis. 2012. "OpenRadio: A Programmable Wireless Dataplane". In *proc. of the HotSDN*.

Barghi, S. and et al. 2012. "Characterizing the throughput gain of single cell MIMO wireless systems with full duplex radios". In *Proc. of the International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*.

B.Di and et al. 2014. "Radio resource allocation for full duplex OFDMA networks using matching theory". In *Proc. of IEEE Conference on Computer Communications Workshops (INFOCOM WS)*.

Bermana, M., J. S. Chaseb, L. Landweberc, A. Nakaod, M. Otte, and et al. 2014. "GENI: A federated testbed for innovative network experiments". *Elsevier Computer Networks*, vol. 61.

Bernardos, C. J., A. D. L. Oliva, P. Serrano, and et al. 2014. "An Architecture for Software Defined Wireless Networking". *IEEE Wireless Communications*, vol. 21, n° 3.

Bhanage, G., I. Seskar, R. Mahindra, and D. Raychaudhuri. 2010a. "Virtual Basestation: Architecture for an Open Shared WiMAX Framework". In *Proc. of the VISA*.

Bhanage, G., I. Seskar, R. Mahindra, and D. Raychaudhuri. 2010b. "Virtual Basestation: Architecture for an Open Shared WiMAX Framework". In *proc. of the VISA*.

Bhanage, G., D. Vete, I. Seskar, and D. RAychaudhuri. 2010c. "SplitAP: Leveraging Wireless Network Virtualization For Flexible Sharing Of WLANs". In *Proc. of the IEEE Globecom*.

Bharadia, D. and et al. 2013. "Full Duplex Radios". In *Proc. of the ACM SIGCOMM*.

Bhaumik, S., S. P. Chandrabose, M. K. Jataprolu, G. Kumar, and et al. 2012. "CloudIQ: A Framework for Processing Base Stations in a Data Center". In *proc. of the ACM MobiCom*.

Bliss, D. W. and et al. 2007. "Simultaneous transmission and reception for improved wireless network performance". In *Proc. of the IEEE/SP 14th Workshop on Statistical transmission and reception for improved wireless network performance*.

Borth, D. E. 1989. "TDMA communications system with adaptive equalization". *Patent. US4852090A, URL: https://www.google.com/patents/US4852090*.

Bosch, P., A. Duminuco, F. Pianese, and T. L. Wood. 2011. "Telco Clouds and Virtual Telco: Consolidation, Convergence, and Beyond". In *proc. of the 6th IFIP/IEEE International Workshop on Broadband Convergence Networks*.

Bosshart, P., G. Gibb, H.-S. Kim, G. Varghese, N. McKeown, and et al. 2013. "Forwarding metamorphosis: fast programmable match-action processing in hardware for SDN". In *proc. of the ACM SIGCOMM*.

Cai, Y., F. R. Yu, and C. Liang. 2014. "Resource Sharing for Software Defined D2D Communications in Virtual Wireless Networks with Imperfect NSI". In *proc. of the IEEE Globecom*.

Casado, M., M. Freedman, J. Pettit, J. Luo, N. McKeown, and S. Shenker. 2007. "Ethane: taking control of the enterprise". In *proc. of the ACM SIGCOMM*.

Chandrasekhar, V., J. G. Andrews, and A. Gatherer. 2008. "Femtocell Networks: A Survey". *IEEE Communications Magazine*, vol. 46, n° 9.

Chiosi, M., D. Clarke, P. Willis, and et al. 2012a. "Network Function Virtualization: An Introduction, Benefits, Enablers, Challenges & Call for Action". In *SDN and OpenFlow World Congress*.

Chiosi, M., D. Clarke, P. Willis, and et al. 2012b. "Network Function Virtualization: An Introduction, Benefits, Enablers, Challenges & Call for Action".

Choi, J. I. and et al. 2010. "Achieving single channel, full duplex wireless communication". In *Proc. of the MobiCom*.

Choi, Y.-S. and et al. 2013. "Simultaneous transmission and reception: algorithm, design and system level performance". *IEEE Transaction on Wireless Communication*, vol. 12, n° 12, p. 5992-6010.

Chowdhury, N. M. K. and R. Boutaba. 2010. "A survey of network virtualization". *Computer Networks*, vol. 54, n° 5, p. 862-876.

Chun, B., D. Culler, T. Roscoe, A. Bavier, L. Peterson, and et al. 2003. "PlanetLab: an overlay testbed for broad-coverage services". *ACM SIGCOMM Computer Communication Review*, vol. 33, n° 3.

Chung, M. and et al. 2015. "Prototyping real time full duplex radios". *IEEE Communicatoin Magazine*, vol. 53, n° 9, p. 56-63.

Cisco. 2014. "Cisco Visual Networking index: Forecase and Methodology 2013-2018". *Cisco White Paper*.

Dimakis, A. G., K. Ramchandran, Y. Wu, and C. Suh. 2011. "A survey on network codes for distributed storage". *IEEE*, vol. 99, n° 3.

Ding, A. Y., J. Crowcroft, S. Tarkoma, and H. Flinck. 2014. "Software defined networking for security enhancement in wireless mobile networks". *Elsevier, Computer Networks*.

Duarte, M. and et al. 2014. "Design and Characterization of a Full Duplex Multi-Antenna System of WiFi Netowrks". *IEEE Trans. Vehic. Tech.*, vol. 63, n° 3, p. 1160-1177.

DUPLO. 2012. "The EU 7th project full-duplex radios for local access (DUPLO)". *Available at http://www.fp7-duplo.eu/*.

Feamster, N., J. Rexford, and E. Zegura. 2013. "The road to sdn". *Queue*, vol. 11, n° 2.

Foster, N., R. Harrison, M. J. Freedman, C. Monsanto, J. Rexford, A. Story, and D. Walker. 2011. "Frenetic: A Network Programming Language". In *proc. of the ACM SIGPLAN International Conference on Functional Programming (ICFP)*.

Gante, A. D., M. Aslan, and A. Matrawy. 2014. "Smart wireless sensor network management based on software-defined networking".

Goyal, S. and et al. 2013. "Analyzing a full duplex cellular system". In *Proc. of the 44th Annual Conference on Information Science and Systems (CISS)*.

Goyal, S. and et al. 2014. "Improving Small Cell Capacity with Common Carrier Full Duplex Radios". In *Proc. of the IEEE ICC*.

Gude, N., T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker. 2008. "NOX: towars an operating system for networks". *ACM SIGCOMM Computer Communication Review*, vol. 38, n° 3.

Gudipati, A., D. Perry, L. E. Li, and S. Katti. 2013. "SoftRAN: Software Defined Radio Access Network". In *proc. of the second ACM SIGCOMM workshop on HOt topics in software defined networking, HotSDN*.

Hamaguchi, T., T. Komata, T. Nagai, and H. Shigeno. 2010. "Framework of better deployment for WLAN access point using virtualization technique". In *proc. of the IEEE WAINA*.

Haneda, K. and et al. 2010. "Measurement of loop-back interference channels for outdoor-to-indoor full duplex radio relays". In *Proc. of the Fourth European Conference on Antennas and Propagation (EUCAP)*.

He, Y., J. Fang, J. Zhang, H. Shen, K. Tan, and Y. Zhang. 2010. "MPAP: Virtualization Architecture for Heterogenous Wireless APs". In *proc. of the ACM SIGCOMM Conference*.

H.H.Choi. 2014. "On the design of user pairing algorithms in full duplexing wireless cellular networks". In *Proc. of the International conference on Information and Communication Technology Convergence (ICTC)*.

Hong, S. and et al. 2014. "Applications of self-interference cancellation in 5G and beyond". *IEEE Communicatoin Magazine*, vol. 52, n° 2, p. 114-121.

Hong, S., J. Mehlman, and S. Katti. 2012. "Picasso: Flexible RF and Spectrum Slicing". In *proc. of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Application*.

Huawei. 2015. "Full Duplex technology for 5G". *Huawei Innovation Research Program Journal*, p. 62-69.

Jain, M. and et al. 2011. "Practical, Real-Time, Full Duplex Wireless". In *Proc. of the 17th Annual Int'l Conf. Mobile Computing and Networking*.

Jain, S., A. Kumar, S. Mandal, J. Ong, and et al. 2013. "B4: Experience with a Globally-Deployed Software Defined WAN". In *proc. of the ACM SIGCOMM*.

Jin, X., L. E. Li, L. Vanbever, and J. Rexford. 2013. "SoftCell: Taking Control of Cellular Core Networks". *rXiv preprint arXiv:1305:3568*.

Johansson, K., A. Furuskar, P. Karlsson, and J. Zander. 2004. "Relation between Base Station Characteristics and Cost Structures in Cellular Systems". In *Proc. of the IEEE PIMRC*.

Kang, J.-M., H. Bannazadeh, and A. Leon-Garcia. 2013. "SAVI testbed: Control and management of converged virtual ICT resources". In *proc. of the IFIP/IEEE International Symposium on Integrated Network Management*.

Karagiannis, G., A. Jamakovic, A. Edmonds, C. Parada, T. Metsch, and et al. "Mobile Cloud Networking: Virtualisation of Cellular Networks". *MCN report*.

Kempf, J., B. Johansson, S. Pettersson, H. Luning, and T. Nilsson. 2012. "Moving the Mobile Evolved Packet Core to the Cloud". In *proc. of the Fifth International Workshop on Selected Topics in Mobile and Wireless Computing*.

Khandani, A. K. 2010. "Methods for Spatial Multiplexing of Wireless Two-Way Channels". *US Patent No. 7,817,641*.

Kim, K.-H., S.-J. Lee, and P. Congdon. 2012. "On Cloud-Centric Network Architecture for Multi-Dimensional Mobility". In *proc. of the MCC*.

Knox, M. 2012. "Single Antenna Full Duplex Communications using a Common Carrier". In *Proc. of the 13th Annual Wireless and Microwave Tech. Conf.*

Kobayashi, M., S. Seetharaman, G. Parulkar, G. Appenzeller, and et al. 2013. "Maturing of OpenFlow and Software-dened Networking through Deployments". *Elsevier*.

Koponen, T., S. Shenker, H. Balakrishnan, N. Feamster, I. Ganichev, and et al. 2011. "Architecting for innovation". *ACM SIGCOMM Computer Communication Review*, vol. 41, n° 3.

Kreutz, D., F. M. V. Ramos, and P. Verissimo. 2013. "Towards Secure and Dependable Software-Defined Networks". In *proc. of the ACM HotSDN*.

kreutz, D., F. M. V. Ramos, P. Verissimo, C. E. Rothenberg, A. Azodolmolky, and S. Uhlig. 2015. "Software-Defined Networking: A Comprehensive Survey". *Proceedings of the IEEE*, vol. 103, n° 1.

Kruschwitz, L. and A. Loeffler, 2005. *Discounted Cash Flow: A Theory of the Valuation of Firms*. West Sussex, England : John Wiley & Sons, Ltd.

Ku, I., Y. Lu, and M. Gerla. 2014. "Software-Defined Mobile Cloud: Architecture, Services and Use Cases". In *proc. of the International Wireless Communications and Mobile Computing Conference (IWCMC)*.

Kyösti, P. 2007. "WINNER II Chanel Models". *IST-4-027756 WINNER II, D1.1.2 V1.2*.

Lee, K., J. Lee, Y. Yi, I. Rhee, and S. Chong. 2013. "Mobil Data Offloading: How Much Can WiFi Deliver?". *IEEE/ACM Transaction on Networking*, vol. 21, n° 2.

Lehpamer, H., 2002. *Transmission Systems Design Handbook for Wireless Networks*. Boston, London : Artech House Inc.

Li, L. E., Z. M. Mao, and J. Rexford. 2012a. "CellSDN: Software-Defined Cellular Networks". *Technical Report, Princeton University*.

Li, L. E., Z. M. Mao, and J. Rexford. 2012b. "Toward Software-Defined Cellular Networks". In *Proc. of the IEEE EWSDN*.

Lin, T., J. M. Kang, H. Bannazadeh, and A. L. Garcia. 2014. "Enabling SDN Applications on Software-Defined Infrastructure". In *Proc. of the IEEE NOMS*.

Lin, Y., L. Shao, Z. Zhu, Q. Wang, and R. K. Sabhikhi. 2011. "Wireless network cloud: Architecture and system requirements". *IBM Journal*, vol. 54, n° 1.

Liu, J., T. Zhao, S. Zhou, Y. Cheng, and Z. Niu. 2014. "CONCERT: A Cloud-Based Architecture for Next-Generation Cellular Systems". *IEEE Wireless Communications*.

Loizillon, F. and et al. 2002. "Final results on seamless mobile IP servie provision economics". *IST-2000-25172 TONIC Deliverable number 11*.

Luo, T., H.-P. Tan, and T. Q. S. Quek. 2012. "Sensor OpenFlow: Enabling Software-Defined Wireless Sensor Networks". *IEEE Communications Letters*, vol. 16, n° 11.

Mccauley, J. "POX Controller". *available at: http://www.noxrepo.org/pox/about-pox/*.

McKeown, N., T. Anderson, H. Balakrishnan, G. Parulkar, L. P. J. Rexford, S. Shenker, and J. Turner. 2008. "OpenFlow: Enabling Innovation in Campus Networks". *ACM SIGCOMM Computer Communication Review*, vol. 38, n° 2, p. 69-74.

Mitola, J. and G. Maguire. 1999. "Cognitive radio: making software radios more personal". *IEEE Personal Communications*, vol. 6, n° 4.

Monsanto, C., J. Reich, N. Foster, J. Rexford, and D. Walker. 2013. "Composing Software Defined Networks". *USENIX NSDI*.

Moradi, M., W. Wu, L. E. Li, and A. M. Mao. 2014. "SoftMoW: Recursive and Reconfigurable Cellular WAN architecture". In *proc. of the CoNEXT*.

Morin, C. and I. Puaut. 1997. "A survey of recoverable distributed shared virtual memory systems". *IEEE Transaction on Parallel and Distributed Systems*, vol. 8, n° 9.

Nayak, A. K., A. Reimers, N. Feamster, and R. Clark. 2009. "Resonance: dynamic access control for enterprise networks". In *proc. of the ACM WREN*.

Nguyen, D. and et al. 2014. "On the spectral efficiency of full-duplex small cell wireless systems". *arXiv:1407.2628 [cs.IT]*.

Niebert, N., S. Baucke, I. EI-Khayat, M. Johnsson, and et al. 2008. "The way 4WARD to the creation of a future internet". In *proc. of the IEEE PIMRC*.

Nuaymi, L., 2007. *WiMAX: Technology for Broadband Wireless Access*. West Sussex, England : John Wiley & Sons.

Nunes, B., M. Mendonca, and et al. 2014. "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks". *IEEE Communications Surverys & Tutorials*, vol. 16.

OpenStack. "available at http://www.openstack.org".

Patel, C. D. and A. J. Shah. 2005. "Cost Model for Planning, Development and Operation of a Data Center". *Hewlett-Packard Development Company, L.P.*

Pentikousis, K., Y. Wang, and W. Hu. 2013. "MobileFlow: Toward Software-Defined Mobile Networks". *IEEE Communications Magazine*, vol. 51, n° 7, p. 44-53.

Pérez, D. L., I. Güvenc, G. D. L. Roche, M. Kountouris, and et al. 2011. "Enhanced Intercell Interference Coordination Challenges in Heterogeneous Networks". *IEEE Communications Magazine*, vol. 18, n° 3.

Perez, S., J. M. Cabero, and E. Miguel. 2009. "Virtualization of the Wireless Medium: A Simulation-Based Study". In *Proc. of the IEEE VTC*. p. 1-5.

pi, Z. and F. Khan. 2011. "An introduction to millimeter-wave mobile broadband systems". *IEEE Communications Magazine*, vol. 49, n° 6.

Rahman, M. M., C. Despins, and S. Affes. 2013. "Analysis of CAPEX and OPEX Benefits of Wireless Access Virtualization". In *proc. of the IEEE ICC*.

Rahman, M. M., C. Despins, and S. Affes. 2014a. "Green Wireless Access Virtualization Implementation: Cost vs. QoS Trade-offs". In *proc. of the 3rd International Conference on Eco-friendly Computing and Communication Systems (ICECCS)*.

Rahman, M. M., C. Despins, and S. Affes. 2014b. "Design Optimization of Wireless Access Virtualization based on Cost & QoS Trade-off Utility Maximization". *submitted to IEEE Transaction on Wireless Communications*.

Rahman, M. M., C. Despins, and S. Affes. 2015a. "End-to-End Programmable, Cloud-based Virtualized HetNet: Advances Made & Challenges to Address". *submitted to Elsevier Computer Communications*.

Rahman, M. M., C. Despins, and S. Affes. 2015b. "Service Differentiation in Software Defined Virtual Heterogeneous Wireless Networks". In *proc. of the IEEE International Conference on Ubiquitous Wireless Broadband (ICUWB)*.

Rahman, M. M., C. Despins, and S. Affes. 2015c. "HetNet Cloud: Leveraging SDN & Cloud Computing for Wireless Access Virtualization". In *proc. of the IEEE International Conference on Ubiquitous Wireless Broadband (ICUWB)*.

Rahman, M. M., C. Despins, and S. Affes. 2015d. "Configuration cost vs. QoS trade-off analysis and optimization of SDR access virtualization schemes". In *proc. of 1st IEEE Conference on Network Softwarization (NetSoft)*.

Rahman, M. M., A. Callard, C. D. G. Senarath, and S. Affes. 2016a. "Multi-Cell Full-Duplex Wireless Communication for Dense Urban Deployment". In *proc. of 1st IEEE Globecom (submitted)*.

Rahman, M. M., A. Callard, G. Senarath, C. Despins, and S. Affes. 2016b. "Deployment of Full Duplex Wireless Systems for Multi-Cell Dense Urban and Rural Networks.". *to be submitted to IEEE Transaction on Wireless Communications*.

Rappaport, T. S., S. Sun, and et al. 2013. "Millimeter Wave Mobile Communications for 5G Cellular: It Will Work!". *IEEE Access*, vol. 1.

Ratal, A. R., L. Jakab, M. Portolés, and et al. 2012. "LISP-MN: Mobile Networking Through LISP". *Wireless Personal Communications*, vol. 70, n° 1.

Raychaudhuri, D. and N. B. Mandayam. 2012. "Frontiers of wireless and mobile communications". *in proc. of IEEE*, vol. 100, n° 4.

Reich, J., C. Monsanto, N. Foster, J. Rexford, and D. Walker. 2013. "Modular SDN Programming with Pyretic". *;login:, The USENIX Magazine*, vol. 38, n° 5, p. 40-47.

Riggio, R., T. Rasheed, and F. Granelli. 2013. "EmPOWER: A Testbed for Network Function Virtualization Research and Experimentation".

R.Kokku, R. Mahindra, H. Zhang, and S. Rangarajan. 2012a. "NVS: A Substrate for Virtualizing Wireless Resources in Cellular Networks". *IEEE/ACM Trans. Netw*, vol. 20, n° 5.

R.Kokku, R. Mahindra, H. Zhang, and S. Rangarajan. 2012b. "NVS: A Substrate for Virtualizing Wireless Resources in Cellular Networks". *IEEE/ACM Transaction on Networking*, vol. 20, n° 5, p. 1333-1346.

Rost, P., C. M. Bernardos, A. D. Domenico, M. D. Girolamo, M. Lalam, A. Maeder, D. Sabella, and D. Wubben. 2014. "Cloud Technologies for Flexible 5G Radio Access Networks". *IEEE Communications Magazine*, vol. 52, n° 5.

Ryu. "available at http://osrg.github.io/ryu/".

Sabharwal, A. and et al. 2014. "In-band full-duplex wireless: Challenges and opportunities". *IEEE Journal on Sel. Area in Comm.*, vol. 32, n° 9, p. 1637-1652.

Sachs, J. and S. Baucke. 2008. "Virtual Radio - A Framework for Configurable Radio Networks". In *proc. of the WICON*.

Sahai, A. and et al. 2011. "Pushing the limites of full-duplex: Design and real-time implementation". *arXiv:1107.0607 [cs.NI]*.

Shao, S. and et al. 2014. "Analysis of carrier utilization in full-duplex cellular networks by dividing the co-channel interference region". *IEEE Communications Letters*, vol. 18, n° 6, p. 1043-1046.

Shen, X. and et al. 2013. "On the design of the scheduling algorithm for the full duplexing wireless cellular network". In *Proc. of the IEEE GLOBECOM*.

Sherwood, R., G. Gibb, K.-K. Yap, G. Appenzeller, M. Casado, N. McKeown, and G. Parulkar. 2009. "FlowVisor: A Network Virtualization Layer". *OPENFLOW-TR-2009-1*.

Shin, S. and G. Gu. 2013. "CloudWatcher: network security monitoring using OpenFlow in dynamic cloud networks (or: How to provide security monitoring as a service in clouds?)". In *proc. of the IEEE ICNP NPSec Workshop*.

Shin, S., P. Porras, V. Yegneswaran, M. Fong, G. Gu, and M. Tyson. 2013. "FRESCO :modular composable security services for software-defined networks". In *proc. of the NDSS*.

SImeone, O. and et al. 2014. "Full Duplex Cloud Radio Access Networks: An Information Theoretic Viewpoint". *IEEE Wireless Communications Letters*, vol. 3, n° 4, p. 413-416.

Sing, N. and et al. 2011. "Efficient and fair MAC for wirelss networks with self-interference cancellation". In *Proc. of International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*.

Singhal, S., G.Hadjichristofi, I. Seskar, and D. Raychaudhri. 2008. "Evaluation of UML based Wireless Network Virtualization". In *Proc. of the Next Generation Internet Networks, Krakow, Poland*.

Smith, G., A. Chaturvedi, A. Mishra, and S. Banerjee. 2007. "Wireless virtualization on commodity 802.11 hardware". In *Proc. of the ACM international workshop on Wireless network test-beds, experimental evaluation and characterization*.

Smith, J. E. and R. Nair. 2005. "The architecture of virtual machines". *IEEE Computer*, vol. 38, n° 5.

Soh, Y. S., T. Q. S. Quek, M. Kountouris, and S. Hyundong. 2013. "Energy Efficient Heterogeneous Cellular Networks". *IEEE Journal on Selected Areas in Communications*, vol. 31, n° 5, p. 840-850.

Sultan, R. A. and et al. 2015. "Mode selection, user pairing, subcarrier allocation and power control in full-duplex OFDMA HetNets". In *Proc. of IEEE ICC Small-Nets Workshop*.

Sundaresan, K., M. Y. Arslan, S. Singh, S. Rangarajan, and S. V. Krishnamurthy. 2013a. "FluidNet: A Flexible Cloud-based Radio Access Network for Small Cells". In *proc. of the MobiCom*.

Sundaresan, K., M. Y. Arslan, S. Singh, S. Rangarajan, and S. V. Krishnamurthy. 2013b. "FluidNet: A Flexible Cloud-based Radio Access Network for Small Cells". In *Proc. of the MobiCom*.

Suresh, L., J. S.-Zander, R. Merz, A. Feldmann, and T. Vazao. 2012. "Towards Programmable Enterprise WLANs with Odin". In *proc. of the HotSDN*.

Taleb, T. and A. Ksentini. 2013. "Follow Me Cloud: Interworking Federated Clouds and Distributed Mobile Networks". *IEEE Network*, vol. 27, n° 5.

Taleb, T., P. Hasselmeyer, and F. Mir. 2013. "Follow me Cloud: An OpenFlow-Based Implementation". In *proc. of the IEEE GreenCom*.

Taleb, T., M. Corici, C. Parada, and et al. 2015. "EASE: EPC as a Service to Ease Mobile Core Network Deployment over Cloud". *IEEE Network*, vol. 29, n° 2.

Tan, K., J. Zhang, J. Fang, H. Liu, and et al. 2011. "Sora: High Performance Software Radio Using General Purpose Multi-core Processors". *Communications of the Association for Computing Machinery*, vol. 54, n° 1.

Tan, K., H. Shen, J. Zhang, and Y. Zhang. 2012a. "Enabling Flexible Spectrum Access with Spectrum Virtualization". In *proc. of the IEEE Symposium on New Frontiers in Dynamic Spectrum Access Network*.

Tan, K., H. SHen, J. Zhang, and Y. Zhang. 2012b. "Enable Flexible Spectrum Access with Spectrum Virtualization". In *proc. of the IEEE Symposium on New Frontiers in Dynamic Spectrum Access Network*.

Vuong, Q. T. N., N. Agoulmine, E. H. Cherkaoui, and L. Toni. 2013. "Multi-Criteria Optimization of Access Selection to Improve the Quality of Experience in Heterogeneous Wireless Access Networks". *IEEE Transaction on Vehicular Technology*, vol. 62, n° 4, p. 1785-1800.

Wang, Y., Y. Zhang, V. Singh, C. Lumezanu, and G. Jiang. 2013. "NetFuse: short-circuiting traffic surges in the cloud". In *proc. of the IEEE ICC*.

Xia, L., S. Kumar, X. Yang, P. Gopalakrishnan, Y. Liu, S. Schoenberg, and X. Guo. 2011. "Virtual wifi: bring virtualization from wired to wireless". *ACM SIGPLAN*, vol. 46.

Xia, W., Y. Wen, and et al. 2015. "A Survey on Software-Defined Networking". *IEEE Communications Surveys & Tutorials*, vol. 17, n° 1.

Xiao, J., Y. Q. R. Hu, L. Gong, and B. Wang. 2013. "Expanding LTE network spectrum with cognitive radios: From concept to implementation". *IEEE Wireless Communications*, vol. 20, n° 2.

Xing, P., L. Yang, C. Q. Li, P. Demestichas, and A. Georgakopoulos. 2013. "Multi-RAT Network Architecture". *Wireless World Research Forum, White Paper, Version 2.0*.

Yang, L., R. Dantu, T. Anderson, and R. Gopal. 2004. "Rfc 3746: Forwarding and control element separation (forces) framework". *Network Working Group*, vol. 4.

Yang, M., Y. Li, D. Jin, L. Su, S. Ma, and L. Zeng. 2013. "OpenRAN: A Software-defined RAN Architecture via Virtualization". In *proc. of the ACM SIGCOMM*.

Yap, K.-K., R. Sherwood, D. Underhill, and et al. 2009. "The Stanford OpenRoads Deployment". In *proc. of WiNTECH*.

Yap, K.-K., M. Kobayashi, R. Sherwood, T.-Y. Huang, M. Chan, N. Handigol, and N. McKeown. 2010a. "OpenRoads: Empowering Research in Mobile Networks". *ACM SIGCOMM Computer Communication Review*, vol. 40, n° 1.

Yap, K.-K., R. Sherwood, M. Kobayashi, T.-Y. Huang, M. Chan, N. Handigol, N. McKeown, and G. Parulkar. 2010b. "Blueprint for introducing innovation into wireless mobile networks". In *Proc. of the second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures (VISA)*.

Yap, K.-K., R. Sherwood, M. Kobayashi, T.-Y. Huang, M. Chan, N. Handigol, N. McKeown, and G. Parulkar. 2010c. "Blueprint for introducing innovation into wireless mobile networks". In *proc. of the second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures (VISA)*.

Yazici, V., U. C. Kozat, and M. O. Sunay. 2014. "A New Control Plane for 5G Network Architecture with a Case Study on Unified Handoff, Mobility and Routing Management". *IEEE Communications Magazine*.

Zaki, Y., L. Zhao, C. Goerg, and A. Timm-Giel. 2010a. "A Novel LTE Wireless Virtualization Framework". In *Proc. of the MONAMI*.

Zaki, Y., L. Zhao, C. Goerg, and A. Timm-Giel. 2010b. "Lte wireless virtualization and spectrum management". In *proc. of the IEEE Wireless and Mobile Networking Conference (WMNC)*.

Zhao, H., R. Mayzus, and et al. 2013. "28 GHz millimeter wave cellular communication measurements for reflection and penetration loss in and around buildings in New York city".

Zhou, L., X. Wang, W. Tu, and et al. 2010. "Distributed Scheduling Scheme for Video Streaming over Multi-Channel Multi-Radio Multi-Hop Wireless Networks". *IEEE Journal on Selected Areas in Communications*, vol. 28, n° 3, p. 409-419.

Zhou, X. and L. Chen. 2014. "Demand Shaping in Cellular Networks". In *proc. of Annual Allerton Conference on Communication, Control and Computing*.

Zhu, Z., Q. Wang, Y. Lin, P. Gupta, S. Kalyanaraman, and H. Franke. 2010. "Virtual Base Station Pool: Towards A Wireless Network Cloud for Radio Access Networks". In *Proc. of the 8th ACM International Conference on Computing Frontiers*.