REINVENTING THE PROTOCOL STACK FOR
NEXT-GENERATION MOBILE NETWORKS

By

PRASAD PRAKASH NETALKAR

A dissertation submitted to the
School of Graduate Studies
Rutgers, The State University of New Jersey
In partial fulfillment of the requirements
For the degree of
Doctor of Philosophy
Graduate Program in Electrical and Computer Engineering

Written under the direction of
Dipankar Raychaudhuri
and approved by

______________________

______________________

______________________

______________________

New Brunswick, New Jersey
May, 2023
ABSTRACT OF THE DISSERTATION

Reinventing the Protocol Stack for Next-Generation Mobile Networks

by Prasad Prakash Netalkar

Dissertation Director:
Dipankar Raychaudhuri

The next-generation wireless network (5G/6G) aims to provide deep coverage, high capacity, and low latency for emerging applications ranging from Augmented Reality (AR), Virtual Reality (VR), emergency networks, and other mission critical 5G services. However, 5G/6G introduces new challenges at multiple layers of the protocol stack due to the lack of interoperability and heterogeneity. There is a need to fundamentally revisit and redesign the traditional protocol stack to support these newer services in the future network. In this thesis, we will address problems arising from 5G heterogeneous deployments specifically those related to spectrum management, mobility management, and the transport layer used for mmWave services. The proposed solutions are based on spectrum models, cross-layer analytics, and mobility protocols using the named-object network architecture and associated distributed algorithms.

Next generation wireless services and applications will increasingly rely on dynamic spectrum access (DSA) methods that can manage spectrum resources rapidly and efficiently. In this context, chapter 2 discusses a novel spectrum management architecture and algorithm design that leverages Spectrum Consumption Models (SCMs) which offer a mechanism for RF devices to announce their intention to use the spectrum and their needs in terms of spectrum protection. In both our experimental setup and simulation environment, RF devices use SCMs to determine compatibility with existing devices and to dynamically configure their transmission parameters. A novel SCM-based deconfliction algorithm and spectrum access methods are developed for large scale wireless network environments and evaluated using a custom simulation platform.
in terms of computation time, the efficiency of spectrum allocation, and the number of device reconfigurations due to interference. The simulation results and experimental evaluation on the ORBIT/COSMOS testbed validate the benefits of using SCMs and their capabilities to perform fine grained spectrum assignments in dynamic and dense communication environments.

Chapter 3 describes a novel distributed mobility management (DMM) scheme based on the “named-object” information centric network (ICN) architecture in which the routers forward data based on unique identifiers which are dynamically mapped to the current network addresses of a device. The work proposes and evaluates two specific handover schemes namely, hard handoff with rebinding and soft handoff with multi-homing intended to provide seamless data transfer with improved throughput during handovers. The proposed handover schemes are evaluated with respect to RTT and throughput using system simulation along with proof-of-concept implementation on the ORBIT testbed. Experimental results are presented to validate the proposed ICN-inspired mobility management protocol.

Finally, in chapter 4, we conclude with a focus on the transport layer necessary for future wireless networks. The proposed end-to-end protocol uses cross-layer feedback with in-network transport proxy for the fast delivery of data over mmWave channels arising in emerging 5G networks. Recent measurement studies of mmWave channels in urban micro cellular deployments show considerable fluctuations in received signal strength along with intermittent outages resulting from user mobility. This results in significant impairment of end-to-end data transfer speed when conventional TCP is used to transport data over such mmWave channels. To overcome this challenge, a new transport protocol capable of adapting to rapidly fluctuating MAC layer throughput whilst achieving high throughput efficiency is required for future large scale mmWave deployment. A specific transport layer protocol called “mmCPTP” is proposed and evaluated using emulation and experimental evaluation. Significant performance gains over TCP are demonstrated.
Completing this thesis would not have been possible without the support, guidance, and encouragement of many individuals who have played a significant role in my academic and personal life. I want to express my sincere thanks to all those who made this journey a memorable and enriching experience.

Firstly, I would like to thank my Ph.D. advisor, Prof. Dipankar Raychaudhuri, for his unwavering support and patience throughout the graduate program. His vast knowledge and expertise in the field have been invaluable in shaping my understanding and approach to real-world research problems. I’m grateful for the countless hours we spent reviewing and discussing new research topics and for providing me with constructive feedback on writing technical papers and presenting my work confidently. I’m deeply indebted to him for his inspiring and generous support in building and cherishing every moment of my graduate career.

I would also like to thank my Ph.D. qualifier, proposal, and defense committee members, Prof. Roy D. Yates, Prof. Narayan Mandayam, Prof. Richard Martin, Prof. Predrag Spasojevic, and Prof. Carlos E Caicedo Bastidas for taking the time to review my work and providing me with valuable insights and suggestions, without which this thesis would not have been complete. Special thanks to Ivan Seskar for having doors always open despite his busy schedule to answer any technical queries related to the testbed and steering my research towards more practical and industry-oriented solutions.

This endeavor would not have been possible without the mention of my collaborators - Prof. Gil Zussman, Prof. Carlos E Caicedo Bastidas, Dr. Igor Kadota, and Dr. Dragoslav Stojadinovic for endless discussions on the spectrum sharing project. I’m forever indebted to our long collaboration, imparting me with various technical skills, knowledge, and expertise on IEEE standards. I would like to extend my sincere
thanks to Dr. Jiachen Chen, and Dr. Sumit Maheshwari for their invaluable help in formulating my research proposal, debugging simulation, and prototyping experiments. I look forward to continuing our collaboration and hope our paths cross again.

I would also like to extend my gratitude to my mentors at Qualcomm, InterDigital, and Samsung Research America for their constant guidance during my internship program. I learned a lot from them on 3GPP specifications, industry-driven standardization, prototyping, field testing, and software tools and techniques. Thanks should also go to Prof. Neelawar Shekar Vittal Shet and my mentors at New York University and PES University, Bangalore to instigate my interest in research and technology.

Special thanks to my colleagues and peers for providing me with a supportive and friendly environment. I will never forget our countless discussions, travel, and fun-packed outdoor activities. Thanks again for making my stay at Rutgers a memorable and special one. Also, I would like to acknowledge staff members at WINLAB, ECE, and Rutgers Global for their timely assistance.

Finally, I would like to extend my heartfelt thanks to my family back in India for their love, support, and encouragement. My parents have instilled in me a strong work principle and a passion for learning. Their unwavering belief in me has gotten me this far, and for that, I will be forever grateful. I would also like to thank my wife, Nisha, for her love, patience, and understanding during the times I really needed it. She put her own needs aside to support me. Many thanks to Rajaneesh, Suma, and Ishan for all the help and encouragement. They have been like a second family for me in the US. I’m also thankful to Vatsala Rao, Girija, Dr. Pooja Netalkar, Dr. Sanath Kumar, Ganesh Rao, Vidya Rao, Karthik Rao, Nagaraj Revankar, Arati Revankar, and Chirag Revankar for their unconditional love and making my winter vacations so memorable.

**Funding:** I’m also thankful to National Science Foundation (NSF) “COSMOS” grant CNS-1827923, “COSMIC” grant OAC-2029295, and “SII-NRDZ” grant AST-2232456 for supporting this work.
Dedication

To my mother Sneha Netalkar, my father Prakash Netalkar, and my grandparents-
Vatsala Rao, Late Ratnakar Rao, Late Radhabai Netalkar and Late Sheshagiri Netalkar
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>iv</td>
</tr>
<tr>
<td>Dedication</td>
<td>vi</td>
</tr>
<tr>
<td>List of Tables</td>
<td>xi</td>
</tr>
<tr>
<td>List of Figures</td>
<td>xiii</td>
</tr>
<tr>
<td>1. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1. Problem statement</td>
<td>3</td>
</tr>
<tr>
<td>1.1.1. Increased need for dynamic spectrum access methods</td>
<td>3</td>
</tr>
<tr>
<td>1.1.2. Poor network performance due to mobility/handoffs</td>
<td>4</td>
</tr>
<tr>
<td>1.1.3. Poor transport performance due to mmWave intermittency</td>
<td>5</td>
</tr>
<tr>
<td>1.2. Contribution and organization</td>
<td>6</td>
</tr>
<tr>
<td>2. Spectrum Consumption Model Assisted Dynamic Spectrum Access</td>
<td>8</td>
</tr>
<tr>
<td>2.1. Introduction</td>
<td>8</td>
</tr>
<tr>
<td>2.2. Spectrum consumption models</td>
<td>9</td>
</tr>
<tr>
<td>2.3. Compatibility computations with SCMs</td>
<td>11</td>
</tr>
<tr>
<td>2.3.1. Compatibility of a single transmitter-receiver pair</td>
<td>12</td>
</tr>
<tr>
<td>2.3.2. Compatibility for multiple devices</td>
<td>12</td>
</tr>
<tr>
<td>2.4. Spectrum management architecture and protocol</td>
<td>13</td>
</tr>
<tr>
<td>2.4.1. System architecture</td>
<td>13</td>
</tr>
<tr>
<td>2.4.2. Control plane description - CIL Protocol</td>
<td>14</td>
</tr>
<tr>
<td>2.5. SCM generation and processing</td>
<td>15</td>
</tr>
<tr>
<td>2.5.1. Transmitter characterization</td>
<td>16</td>
</tr>
</tbody>
</table>
2.5.2. Receiver characterization ............................................. 17

2.6. Dynamic spectrum access with SCMs ............................................. 18

2.6.1. Spectrum deconfliction algorithm ............................................. 18

2.6.2. LCS - Sequential spectrum access ............................................. 21

2.6.3. LND - Distributed spectrum access ............................................. 21

Node status/state ................................................................. 23

Message exchange types ......................................................... 23

Timer types .......................................................... 24

Protocol details ................................................................. 24

2.7. Performance evaluation .............................................................. 27

2.7.1. Sequential spectrum access .............................................................. 29

Spectrum usage ................................................................. 30

Computation time ................................................................. 31

Sequential spectrum access enhancement ............................................. 33

Channel separation ................................................................. 34

2.7.2. Comparison of Sequential and Distributed spectrum access ... 35

Fixed topology scenario ................................................................. 35

Complex topology scenario ................................................................. 37

2.8. DSA experimentation on ORBIT/COSMOS testbed ......................... 43

2.9. Related work ........................................................................ 46

2.10. Summary ............................................................................ 47

3. Network Assisted Distributed Mobility Management ......................... 48

3.1. Introduction ........................................................................ 48

3.2. Name Based Approach For Mobility Support ............................................. 49

3.3. Seamless Handover Techniques .............................................................. 51

3.3.1. Hard Handoff with Rebinding .............................................................. 51

3.3.2. Soft Handoff with Multihoming .............................................................. 54

3.4. Performance Evaluation ........................................................................ 55
3.4.1. ORBIT/COSMOS testbed experimentation ........................................ 56
   Rebinding for Hard Handoff ....................................................... 56
   Multihoming for Soft Handoff ................................................... 57
   Impact of Router Memory on Handover ......................................... 58
   Comparison with Existing Handover Schemes .................................... 59

3.4.2. City Scale Simulation ............................................................. 59
   Throughput Evaluation .............................................................. 63
   RTT Evaluation ........................................................................... 65
   Load at Intermediate Router ....................................................... 66

3.5. Related Work .............................................................................. 67

3.6. Summary ....................................................................................... 68

4. Cross-layer Assisted Pull based Transport for mmWave Networks ....... 70
   4.1. Introduction ............................................................................. 70
   4.2. TCP Performance over mmWave ............................................... 71
   4.3. Design Objectives .................................................................... 74
   4.4. mmCPTP - Transport Protocol .................................................. 75
      4.4.1. Design Concepts ............................................................... 75
      4.4.2. Design Details ................................................................. 76
   4.5. Protocol Overview .................................................................... 78
      4.5.1. ICN based future internet architecture ................................. 79
   4.6. Implementation Details ............................................................. 81
   4.7. Performance Evaluation ........................................................... 83
      4.7.1. Evaluation with 5G NR stack ............................................. 84
         System setup ........................................................................... 84
         Single UE ................................................................................ 85
         Multiple UEs and fairness ......................................................... 86
      4.7.2. End-to-end simulation ....................................................... 88
         System setup ........................................................................... 88
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content delivery over mmWave channel</td>
<td>88</td>
</tr>
<tr>
<td>Impact of RTT and loss rate on throughput</td>
<td>89</td>
</tr>
<tr>
<td>Trade-offs between multiple Pull techniques</td>
<td>90</td>
</tr>
<tr>
<td>4.7.3. Real-time ORBIT emulation</td>
<td>90</td>
</tr>
<tr>
<td>4.7.4. Real-time emulation on COSMIC testbed</td>
<td>91</td>
</tr>
<tr>
<td>4.8. Related Work</td>
<td>93</td>
</tr>
<tr>
<td>4.9. Summary</td>
<td>95</td>
</tr>
<tr>
<td>5. Conclusion</td>
<td>96</td>
</tr>
<tr>
<td>5.1. Design trade-offs and integration</td>
<td>97</td>
</tr>
<tr>
<td>5.2. Looking ahead</td>
<td>98</td>
</tr>
<tr>
<td>Acknowledgment of Previous Publications</td>
<td>99</td>
</tr>
<tr>
<td>References</td>
<td>100</td>
</tr>
</tbody>
</table>
## List of Tables

2.1. Key Parameters of Algorithm [1] .............................................. 20

2.2. Comparison between sequential and distributed spectrum access methods 22

2.3. Simulation Parameters ......................................................... 28

2.4. Performance counters of sequential spectrum access averaged over 100 trials for different network sizes ................................................................. 30

2.5. Presence of aggregate interference averaged over 100 trials for different network sizes ................................................................. 31

2.6. Global compatibility error (%) and average computation time per link (in seconds) due to the use of the distance (D) parameter in sequential spectrum access (Algorithm 2), averaged over 30 random trials. ........ 34

2.7. Performance evaluation of SD algorithm and other spectrum access methods considering fixed topology scenario ......................................................... 36

2.8. Comparison of LND step count with the chromatic number of standard graph coloring algorithms from Python NetworkX package, for a network of different link configurations (p=200m, 100 trials). ............ 39

2.9. Global compatibility error (%), system throughput, mean channel usage, and average computation time (in seconds) per link for various spectrum access methods, averaged over 30 random trials. ...................... 42

2.10. Experimentation Settings ......................................................... 44

3.1. Packet drop in IP based Handover .............................................. 59

3.2. System Simulation Parameters ................................................ 60

3.3. Comparison of different Mobility Management Schemes ........ 67

4.1. InterDigital EdgeLink TCP performance on COSMOS .................. 72

4.2. System Simulation Parameters ................................................ 85
# List of Figures

1.1. 5G Heterogeneous Network Challenges ............................................. 4
1.2. 4G and 5G NR Shannon throughput performance ............................... 5
2.1. Spectrum Consumption Model (SCM) types ........................................ 11
2.2. Overview of decentralized spectrum management architecture .............. 13
2.3. CIL - Network Interaction Language protocol details .......................... 15
2.4. Transmitter spectrum mask and receiver underlay mask ....................... 16
2.5. Topology used for receiver characterization ...................................... 17
2.6. LND - High Level Overview .......................................................... 22
2.7. LND - Timing diagram of Distributed Spectrum Access ........................ 26
2.8. ABM simulation framework ............................................................ 28
2.9. Histogram showing the number of channels used by Algorithm 2 to deconflict networks of different sizes ............................................. 29
2.10. Histogram showing the number of channels used among sequential algorithm variants (100 links & 100 trials) ........................................... 32
2.11. Average time required per link among sequential algorithm variants and the baseline algorithm to deconflict spectrum use ............................................. 33
2.12. Histogram showing the number of channels used - D constrained vs. non-constrained sequential algorithm (100 links & 100 trials) .......................... 34
2.13. Histogram showing the number of channels used by the sequential algorithm to deconflict using 2 MHz channel separation .................................... 35
2.14. Fixed topology setup ........................................................................ 36
2.15. Distributed spectrum access (100 links & 1 trial) - (a) Convergence for different peer values (b) Histogram depicting WDs in each state over time (p=200m) ........................................ 38
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.16.</td>
<td>Average step count and computation time per link for sequential and distributed spectrum access to deconflict networks of different size (100 trials).</td>
</tr>
<tr>
<td>2.17.</td>
<td>Histogram depicting channel allocation in distributed spectrum access for networks of different link configuration (100 trials).</td>
</tr>
<tr>
<td>2.18.</td>
<td>Experiment scheme – Timeline of networks joining the coordinated spectrum use environment.</td>
</tr>
<tr>
<td>2.19.</td>
<td>List of compatibility tests between WD3’s devices (Tx3 and Rx3) and existing Tx/Rx devices.</td>
</tr>
<tr>
<td>2.20.</td>
<td>Fosphor visualization showing the spectrum occupancy as the first ($t_1$), second ($t_2$) and third ($t_3$) network start operating.</td>
</tr>
<tr>
<td>3.1.</td>
<td>Association vs signal strength plot</td>
</tr>
<tr>
<td>3.2.</td>
<td>Hard handoff with rebinding</td>
</tr>
<tr>
<td>3.3.</td>
<td>Timing diagram for hard handoff</td>
</tr>
<tr>
<td>3.4.</td>
<td>Soft handoff with multihoming</td>
</tr>
<tr>
<td>3.5.</td>
<td>Timing diagram for soft handoff</td>
</tr>
<tr>
<td>3.6.</td>
<td>Homogeneous hard handoff with rebinding in WiFi</td>
</tr>
<tr>
<td>3.7.</td>
<td>Homogeneous soft handoff with multihoming in WiFi</td>
</tr>
<tr>
<td>3.8.</td>
<td>Heterogeneous soft handoff between WiFi and LTE</td>
</tr>
<tr>
<td>3.9.</td>
<td>Packet loss due to queue size during handover</td>
</tr>
<tr>
<td>3.10.</td>
<td>HetNet flat core architecture</td>
</tr>
<tr>
<td>3.11.</td>
<td>HetNet deployment with real user mobility traces</td>
</tr>
<tr>
<td>3.12.</td>
<td>Throughput for various UE access schemes</td>
</tr>
<tr>
<td>3.13.</td>
<td>Average RTT and throughput performance</td>
</tr>
<tr>
<td>3.14.</td>
<td>Throughput performance for flowrate 30 Mbps</td>
</tr>
<tr>
<td>3.15.</td>
<td>Multihomed throughput performance</td>
</tr>
<tr>
<td>3.16.</td>
<td>RTT for various scheduling schemes</td>
</tr>
<tr>
<td>3.17.</td>
<td>Impact of load at intermediate router on RTT</td>
</tr>
<tr>
<td>3.18.</td>
<td>InterDigital mmWave Tx/Rx experimentation setup</td>
</tr>
<tr>
<td>4.1.</td>
<td>InterDigital mmWave Tx/Rx experimentation setup</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

The next-generation wireless network (5G/6G) aims to provide high bandwidth and low latency support for upcoming IoT devices, mobile vehicular networks, emergency networks and various other emerging services such as Augmented Reality (AR) [1], Virtual Reality (VR) [2], etc. With the expected rise in mobile data traffic by 4.4x between 2021 to 2027 (368 EB per month) [3], a network infrastructure supporting such a surge in traffic is required. Telecom operators are now focusing on dense deployment [4,5] of cells involving heterogeneous radio access technologies (RAT) not just limited to LTE, WiFi, 5G (sub-6GHz) and 5G mmWave [6–8] to support such growth in demands. The 5G heterogeneous deployments introduces various new challenges at multiple layers of the protocol stack associated with high signal interference, frequent handoffs/handovers, and transport layer service impairments arising due to a lack of interoperability and heterogeneity.

Spectrum being a finite physical layer resource has motivated many researchers to develop new technologies and standards to effectively share it among multiple users for improved data rate, reduced interference, and efficiency. For the emerging 5G NR (New Radio) systems with new technologies such as carrier aggregation and dual-connectivity (or multihoming), dynamic spectrum access (DSA) is expected to be a key enabler to support the increase in user traffic demands and requirements [9]. Significant effort has also been put to develop co-existence techniques to address interference and heterogeneity [10–12]. Recent advancements in 5G/6G wireless technologies, standards, and policies enable DSA methods on a more granular basis than previously possible.

Another challenge to be addressed is that of providing seamless mobility among various heterogeneous networks for supporting real-time applications such as VoIP [13]
(Voice over IP), streaming services, and AR/VR. Multihoming currently in use provides seamless connectivity but introduces an additional challenge of resolving multiple network addresses associated with User Equipment (UE). Moreover, TCP/IP based networks are primarily designed to support static users and thus, perform poorly for mobile users due to variations in wireless link quality.

In addition, the opening up of new frequency bands in the mmWave spectrum raises the prospect of data transfer to mobile wireless devices at ultra-high speeds of the order of Gbps. However, existing end-to-end (E2E) transport protocols such as TCP do not perform well with mmWave link layers [14]. Previous measurement studies [15] confirm the fact that mmWave bands are highly susceptible to blockages and a mobile user may experience rapid and intermittent interruptions in the application layer services [16]. These fluctuations in the received mmWave signal strength are caused by blockages due to buildings or human body movements and various other environmental factors such as oxygen/rain absorption [17,18]. While mmWave radio channels have access to abundant bandwidth, they suffer from rapid fluctuations in signal strength which may lead to a significant slow-down in the actual data transfer speed available to applications running on 5G mobile devices.

In summary, the challenges discussed above motivate us to understand the requirements of future networks, and the need to fundamentally revisit and redesign traditional protocol stacks, based on the new techniques of spectrum models, cross-layer analytics, mobility protocols using the named-object network architecture, and associated distributed algorithms as explored in this thesis. For beyond 5G networks, it is essential to have dense heterogeneous deployments with multiple radio access technologies (RATs) and numerous cells to enable better spectrum utilization, seamless mobility, and high-speed data transfer. However, the emerging issues that arise from these networks must be addressed timely for large-scale deployment.
1.1 Problem statement

In this section, we describe in detail the problems occurring at various layers of the traditional protocol stack specifically related to spectrum management (at the physical layer) [19, 20], mobility management (at the network layer) [21], and impairments in the data transfer speed (at the transport layer) [22] due to mmWave intermittency as addressed in this thesis.

1.1.1 Increased need for dynamic spectrum access methods

Spectrum sharing has been extensively studied for over a decade and, recently, there has been special interest in the coexistence of LTE and WiFi, with several proposed solutions for LTE in unlicensed spectrum. Accordingly, more and more effort was focused on techniques to improve the coexistence of these networks in the same spectrum, some of which use collaborative methods and additional protocols for networks to coordinate their spectrum usage.

In [23], the authors present an in-depth survey on recent 5G spectrum sharing (SS) techniques that use DSA, categorizing them in terms of their architecture (i.e., centralized or distributed), spectrum allocation behavior (i.e., cooperative or non-cooperative), and method (i.e., dynamic exclusive, open access, and hierarchical). The survey discussed various SS techniques which use concepts from game theory, information theory, stochastic modeling, and several database assisted algorithms. There are also additional challenges related to real implementation, standardization, privacy, compatibility determination methods, and system architecture design. Several other centralized spectrum management architectures have been proposed in the past such as the Spectrum Access System (SAS) developed for 3.5 GHz CBRS band [24, 25]. The potential benefit of using centralized architectures are significant, but for a larger deployment, they lack (a) scalability, (b) have a single point failure, and (c) provide insufficient local autonomy.
Taking inspiration from the DARPA’s Spectrum Challenge (SC2-2016), NSF’s Spectrum Innovation Initiative (SII-2021), and ongoing standardization by the IEEE DySPAN community [26], we believe that there is a need for a spectrum management architecture and algorithm design which uses standard techniques to perform granular DSA in time, frequency and geographic locations [27]. In addition, the designed architecture should also support the exchange of meta-data on the local wireless environment with the peers such that the networks are not just limited by local sensing information but also has a better regional visibility to realize such sharing and protect current and future passive devices [28].

1.1.2 Poor network performance due to mobility/handoffs

The dense heterogeneous networks create newer possibilities to enhance user quality of service (QoS) using seamless access techniques such as multihoming and offloading [29]. However, there exist challenges associated with frequent handovers, address resolution, traffic load balancing, and high signal interference. Several SDN based centralized and Distributed Mobility Management (DMM) architectures have been proposed [30–32] in the past. But, for a larger network deployment, these solutions face scalability and controller management issues. Specifically, the architectures described above face challenges in: (a) configuring gateways, (b) setting-up tunnels between various RATs either on per UE basis or operator specific policies, (c) lack of network support for efficient handover decisions, and (d) packet loss and increased handover latency.

Figure 1.1: 5G Heterogeneous Network Challenges
TCP interprets wireless link rate fluctuations and errors as congestion, triggering a mechanism to slow-down the rate and react aggressively to any packet loss events. Multipath TCP (MPTCP) [33] for multihoming built on top of TCP faces similar issues by invoking congestion control mechanism on these multiple subflow paths and performs poorly in a highly mobile environment [34]. In addition, extensions to the TCP/IP architecture such as Mobile IP [35] also suffer from a number of unresolved issues relating to triangular routing and dual connectivity [36]. Hence, mobility management remains an important network design issue to be addressed (see Fig. 1.1) in next generation 5G mobile core network motivating for a scalable DMM scheme enabling seamless connectivity with low control overhead and supporting interoperability.

1.1.3 Poor transport performance due to mmWave intermittency

The unique PHY layer characteristics of mmWave represent a design challenge to the upper layers of the network protocol stack, motivating a redesign of E2E transport protocols [37]. In particular, several studies have shown that the most commonly used reliable transport protocol, TCP, performs poorly over mmWave links [18, 38]. Using TCP for mmWave results in frequent timeout followed by the slow growth of congestion window [15,39] due to frequent switching and handoffs between the line of sight (LoS), non line of sight (nLoS), and sub 6 GHz paths. Such behavior often confuses the application layer logic [40,41], for example, the bit-rate adaptation of video streams can further decrease the overall quality of service (QoS) and throughput. The authors
evaluated the performance of a commercial 5G mmWave non-standalone (NSA) Verizon network and found that the handover due to signal fluctuations is a major performance issue that needs to be addressed in mmWave systems. A user frequently switches between various 4G/5G paths as shown in Fig. 1.2, an example Shannon bit-rate trace of a mobile user experiment conducted in downtown Minneapolis street with 5G capable phone [42]. The traces were obtained by considering LTE/4G bandwidth of 10 MHz and NR/5G bandwidth of 150 MHz, sampled every 1s. These events cause TCP to react as if the channel were congested, by timing out and gradually increasing the bit-rate following the additive increase, multiplicative decrease (AIMD) procedure thus motivating us to revisit traditional transport protocol and put forward a new solution to overcome the problems of TCP probing and wastage of bandwidth during temporary outages in 5G mmWave networks.

1.2 Contribution and organization

In this thesis, at first, we address the issue of spectrum access in dense heterogeneous deployment with the introduction of a spectrum management architecture and protocol design which uses Spectrum Consumption Models [43] (SCMs) and Collaborative Interaction Language (CIL) [44] developed for DARPA SC2 to facilitate autonomous and dynamic selection of spectrum resources for each network in order to establish non-interfering communication links between the devices of each network. The SCMs describe the characteristics and boundaries of spectrum usage of the devices in each network and this kind of use of SCMs is a first of its kind in a civilian communications environment. Next, we propose a novel SCM-based Spectrum Deconfliction (SD) algorithm that takes into consideration aggregate interference and makes use of frequency and power adjustments to deconflict spectrum use. Further, we propose two spectrum access methods, a) Sequential and b) Distributed that use the deconfliction algorithm to perform spectrum assignment at scale. The proposed algorithm is validated using a custom built simulation platform against the scalability and feasibility of SCM-based DSA for coordinating spectrum use in dynamic and dense communication environments. Further, we describe a spectrum access framework built on ORBIT/COSMOS [45][47]
testbed which supports and facilitates coordinated use of spectrum resources between collaborative wireless networks involving SCMs and CIL. The simulation results and the experimental validation demonstrate the feasibility of SCM based spectrum use deconfliction technique in performing fine-grained spectrum assignments.

In chapter 3, we focus on improving the network performance of mobile users specifically aimed towards providing seamless handovers in heterogeneous networks as proposed by beyond 5G architecture. The named-object based network architecture which helps in providing seamless mobility along with specific protocol designs for handover is presented first. Next, we propose two novel handover schemes, hard handoff with rebinding and soft handoff with multihoming using the above architecture. The proposed handoff schemes are evaluated with a proof-of-concept implementation in the ORBIT/COSMOS testbed and further validated using system simulation by considering real user mobility traces. When compared to IP based handover, our solution solves the mobility problem without the resetting of end-to-end sessions and tunnels during the handover. Our proposed method can achieve seamless connectivity with zero packet loss given sufficient router storage capacity. The results demonstrate the benefits of using named-object based network techniques to enable superior handover performance in terms of packet loss, latency, throughput, and interoperability.

Finally, in chapter 4, we describe our latest work in solving challenges associated with poor transport performance of mmWave channels in urban micro cellular deployments. At first, we describe the behavior of TCP over mmWave channel under a lab environment and identify its problems as well as list the key requirements of a transport protocol to support such intermittent links. Next, we present mmCPTP - a novel Cross-Layer Pull based Transport Protocol for 5G mmWave Networks, which specifically uses concepts from Information Centric Networking (ICN) and cross-layer feedback support for faster delivery of data over mmWave channels associated with emerging 5G services. The proposed scheme is implemented in a 5G NR stack and is further validated using NS3 E2E simulation and testbed implementation against various TCP and Indirect-TCP (I-TCP) versions over highly intermittent mmWave channel. Results presented, show significant performance gains for mmCPTP over TCP.
Chapter 2

Spectrum Consumption Model Assisted Dynamic Spectrum Access

2.1 Introduction

The evolution of wireless communication services and technologies increasingly relies on enhanced algorithms for spectrum management that enable heterogeneous devices to share limited spectrum resources and coexist harmoniously. In 2016, DARPA started the Spectrum Collaboration Challenge (SC2), engaging multiple teams of researchers with the common goal of developing dynamic spectrum access (DSA) algorithms that allow multiple networks to collaboratively adapt in real-time to fast-changing congested spectrum environments. The networks were called Collaborative Intelligent Radio Networks (CIRN), and the interaction language developed by DARPA was named Collaborative Interaction Language (CIL) [44]. More recently, the National Science Foundation (NSF) has started the Spectrum Innovation Initiative (SII). The goal of the initiative is to build an ecosystem for research related to dynamic and agile spectrum utilization.

As a part of our efforts to address the challenges that have been identified by these initiatives, in this chapter, we propose and develop standardized ways to coordinate and synchronize large-scale spectrum access using Spectrum Consumption Models (SCMs). SCMs offer a standardized mechanism [51] for RF devices to “announce” or “declare” their intention to use the spectrum (in the case of transmitters) or their needs in terms of spectrum protection (in the case of receivers and passive devices). This declaration (i.e., the SCM) can simplify spectrum use coordination when compared to DSA algorithms that rely solely on sensing to avoid interference between several RF transmitters and receivers [52,53]. The major contributions are as follows:
• We introduce a spectrum management architecture and protocol design leveraging SCM and CIL for efficient spectrum access between multiple networks.

• We propose a novel *SCM-based Spectrum Deconfliction (SD) algorithm* that takes into consideration aggregate interference and makes use of frequency and power adjustments to deconflict spectrum use.

• Using the proposed SD algorithm, we develop sequential and distributed spectrum access methods and evaluate their performance based on scalability and feasibility for coordinating spectrum use in dynamic and dense communication environments.

• Finally, we demonstrate a spectrum access framework built on the NSF PAWR COSMOS testbed [47, 54] that perform DSA coordination automatically using SCMs from three different wireless networks.

### 2.2 Spectrum consumption models

SCMs provides an information model that can capture the boundaries of the use of spectrum by RF systems so that their compatibility (i.e., non-interference) can be arbitrated by efficient and standardized computational methods [51, 55, 56]. The information captured in SCMs allows for the efficient determination of aggregate interference levels and aggregate compatibility interference between many devices.

The IEEE 1900.5.2 standard [55] for modeling spectrum consumption specifies 11 constructs for an SCM:

1. **Reference power:** This value provides a reference power level for the emission of a transmitter or the allowed interference in a receiver. It is used as the reference power value for several other SCM constructs (i.e., spectrum mask, underlay mask, and power map).

2. **Spectrum mask:** Data structure that defines the relative spectral power density of emissions by frequency.
3. **Underlay mask**: Data structure that defines the relative spectral power density of allowed interference by frequency.

4. **Power map**: Data structure that defines a relative power flux density per solid angle.

5. **Propagation map**: Data structure that defines a path loss model per solid angle.

6. **Intermodulation mask**: Data structure that defines how co-located signals generate intermodulation products in a transmitter or receiver.

7. **Platform name**: A name or list of names of platforms that are attributed to a particular site (i.e., ship, airplane, etc.). They are useful in identifying when multiple systems are co-located.

8. **Schedule**: Construct that specifies the time in which the model applies (start time, end time). Periodic activity can also be defined.

9. **Location**: The location where an RF device may be used. Several types of locations and trajectory/orbit descriptions are supported.

10. **Minimum power spectral flux density**: A power spectral flux density that, when used as part of a transmitter model, implies the geographical extent to which receivers in the system are protected.

11. **Policy or protocol**: A named protocol or policy with parameters that define behaviors supported by a device or systems that allow different systems to be co-located and to coexist in the same spectrum.

These constructs can be used to build different types of SCMs that follow an aggregation hierarchy, as shown in Fig. 2.1. It is worth noting that depending on the type of model and its purpose, not all constructs are required. Fig. 2.1 shows the relationships between different types of SCMs as defined in the IEEE 1900.5.2 standard. A transmitter model captures the extent of RF emissions of an active radio device, including but not limited to: spectral emission mask, propagation map, antenna radiation pattern, possible locations of the device, and times of operation. A receiver
model conveys what is harmful interference to an RF device, providing a limit to the aggregate interference that transmitter devices can cause to a receiver in the temporal, spatial, and spectrum dimensions. System models are a collection of transmitter and receiver models that collectively capture the spectrum use of an RF system. An SCM set is a collection of system, transmitter, and receiver SCMs. SCM sets can be used to structure lists that describe the spectrum that is available for use (Spectrum authorization sets), identify constraints to spectrum use (Spectrum constraint sets), and list the spectrum being consumed (used) by a group of systems and devices (Collective consumption set).

### 2.3 Compatibility computations with SCMs

In addition to defining the constructs for SCMs, the IEEE 1900.5.2 standard specifies a method for computing the compatibility of spectrum use between devices and/or systems that have expressed the boundaries of their spectrum use via SCMs [51, 55]. Depending on the locations of the devices for which compatibility is to be assessed and if there is overlap in their spectrum use operations (in time and frequency), the information conveyed by their SCMs related to transmitter spectrum masks, receiver underlay mask, reference power, power map, and propagation map, among other constructs, determine the details of a link budget computation.
2.3.1 Compatibility of a single transmitter-receiver pair

The Compatibility Test (CT) using SCMs aims at determining if a transmitter model is compatible with a receiver model. CT begins by checking if the SCMs overlap in both time and frequency. If they do not overlap, there’s compatibility, if they do overlap, then the evaluation continues. The IEEE 1900.5.2 standard describes how to compute the power spectral flux density (PSFD) from the transmitter at the location of the receiver and the corresponding maximum allowed interference power for the receiver. In case the transmitter does not exceed the receiver’s maximum allowed interference power, the devices are declared compatible. The difference between the maximum allowed interference and the transmitter’s power at the receiver’s location is the power margin. The power margin can be used to determine by how much a transmitter’s power could be increased – if necessary (e.g., to expand coverage) – while still being compatible with the receiver. Alternatively, in case the devices are declared incompatible, the power margin can be used to determine the amount of attenuation necessary at the transmitter to achieve compatibility [57].

2.3.2 Compatibility for multiple devices

When several transmitters and receivers interfere with one another, the CT is based on the computation of the aggregate interference caused by the transmitters under consideration at a particular receiver. Aggregate compatibility is achieved when the aggregate interference at every receiver under consideration is below its maximum allowable interference power [57]. The IEEE 1900.5.2 standard provides guidelines and a method to compute aggregate compatibility using SCMs, for scenarios with many receiver and transmitter devices. When the locations of transmitters and receivers are static, the compatibility test for a single transmitter-receiver (Tx/Rx) pair can be easily extended to the multiple transmitters and multiple receivers case. When there is mobility, before evaluating compatibility, the most constraining configuration for a particular receiver needs to be found. This configuration should be a feasible positioning and configuration of all devices in the scenario that maximizes the aggregate interference on that
receiver 57, 58. In this work, we focus on static devices. Mobility related issues are left for future work.

2.4 Spectrum management architecture and protocol

In this section, we describe our proposed spectrum management architecture along with a detailed description of CIL protocol used for coordination between multiple independent networks.

2.4.1 System architecture

The spectrum management architecture illustrated in Fig. 2.2 is composed of four functional planes:

- cloud based spectrum service plane,
- wireless domain control plane,
- wireless data plane, and
- monitoring and measurement plane.

The sensors, radio devices, and networks on the data plane are represented by Wireless Domain (WD) controllers in the control plane. In particular, a WD controller can represent one or more wireless networks operating in a single administrative domain. SCMs

Figure 2.2: Overview of decentralized spectrum management architecture.
from individual RF devices in each of the wireless networks are aggregated at the corresponding WD controller. The WD control plane is based on DARPA’s CIL, allowing for the exchange of messages containing SCMs between the domains as well as to control radio nodes in the data plane. In addition to peering of SCM exchanges enabled by the control plane, a cloud based spectrum service layer is introduced to accommodate hierarchical control with the benefits of centralized optimization involving complex AI/ML algorithms. The cloud service layer provides spectrum management, monitoring, and marketplace capabilities to which WD’s can subscribe. The data, control, and cloud service planes are further supported by an independent spectrum monitoring infrastructure intended to provide accountability for actual spectrum use. The monitoring plane collects and aggregates sensor data which is then passed up to a spectrum analytic service in the cloud layer for further processing. The analytic/monitoring application in the cloud disseminates this information to WD controllers using information centric PUB/SUB techniques.

2.4.2 Control plane description - CIL Protocol

The network interaction language for our spectrum management architecture was built on top of DARPA’s implementation of CIL. CIL was originally designed as a PUB-SUB message queuing system and in our framework, we introduced several adaptations and improvements to support SCMs. Each network has a designated node, a WD controller, which can use the interaction language to communicate with other network’s WDs. For high performance and efficiency of the message queuing service, Google’s Protocol Buffers are still used to convert all supported messages into binary blobs. The message types were adapted to support SCM exchange and compatibility reports between the WDs. The compatibility computations are included as a component of the interaction language. First, with each new network joining the framework, it only sends/receives SCM information from the networks in the same interference domain and performs compatibility computations to determine the optimal transmission parameters. The details of the protocol used by the interaction language involving SCMs are shown in Fig. 2.3. Some of the message exchanges of the CIL protocol are as follows:
Figure 2.3: CIL - Network Interaction Language protocol details.

- **Register()**: Generated by WD controller to register with collaboration server or regional aggregator
- **Inform()**: Regional aggregator informs newly joined WD about existing WDs in the network
- **Notify()**: Regional aggregator notifies existing WDs about the newly joined WD
- **SCM_request()**: Message to request SCMs from peer WDs
- **SCM_response()**: Reply message to send SCM to the requester WD
- **CT_report()**: Sends compatibility test report to peer WDs
- **Calibrate_radios()**: Message to calibrate SDRs with respective gain, frequency, modulation etc within each WD
- **Leave()**: Generated by WD when exiting the system

### 2.5 SCM generation and processing

The interactions in our control plane will rely on the exchange of SCM messages that characterize the transmitters and receivers of each WD. The required constructs for
the transmitter models are: reference power, spectrum mask, power map, propagation map, schedule, and location. For the receivers, the spectrum mask is not required but an underlay mask is (see Fig. 2.4). For our analysis, we used a set of NI USRPs as transmitters and another set as receivers. The schedule and location of each transmitter and receiver device was assumed to be well known. For simplicity, all transmissions used BPSK modulation with a channel bandwidth of 1 MHz. Additional details on the characterization of the devices and the processing of their SCMs are mentioned in the following subsections.

2.5.1 Transmitter characterization

In order to characterize and build the SCM for a transmitter (Tx) USRP, we first obtain the Power Spectral Density (PSD) plot that will help us build the spectrum mask for the device. This is performed by setting Tx gain to 10dB, resolution bandwidth to 100 KHz or lower, and amplitude 500mV. The received samples are captured at a distant receiver (Rx) setup to operate at the same central frequency as the Tx and the distance between Tx and Rx is reported. Next, we change the amplitude values at the Tx to understand how well it translates to radiated power. Moreover, we also need samples that cover at least 1.5 times the bandwidth of the expected received signal so that we can generate a spectrum mask that can capture any out-of-band emissions. The separation distance between the devices should always be at least 1 meter. Finally, we repeat the same experiment again but with a different receiver and report the distances between Tx
and Rx. Capturing the Tx’s radiated power at receivers located at different distances provides details to elaborate the propagation map construct needed in the SCMs.

2.5.2 Receiver characterization

To characterize the receiver USRP, determining the shape of its underlay mask is key. For this purpose, we fix the Tx amplitude to 500mV, Rx and Tx gain to 10dB and the center frequency at the receiver to 2 GHz. From the Tx, transmit a BPSK modulated signal while varying the central frequency and capture the PSD image and SNR value at each frequency value at the receiver. The central frequency of the transmission is varied in steps of 200 KHz from 1997 MHz to 2003 MHz. Next, we gather data to determine the allowable interference on the receiver considering Fig. 2.5 as an example topology. From Tx1, we transmit a BPSK modulated signal with a center frequency of 2 GHz and gain 10dB. At the receiver, we make sure that we can demodulate Tx1’s BPSK signal and capture the signal at the receiver while Tx1 is ON and Tx2 is OFF. Continuing with the same setup as before, while Tx1 is still transmitting, Tx2 is turned on and transmits a QPSK modulated signal (interfering signal) with a center frequency of 2 GHz. The initial Tx2 gain should be 10dB, and later we change Tx2’s gain until we get to a value which we will call $X$ where Rx1 will stop demodulating Tx1’s transmissions and if Tx2’s gain is later lowered by 0.5dB (i.e., Tx2’s gain becomes $X - 0.5$ dB), Rx1 will be able to demodulate some of Tx1 transmissions. We report the value of $X$ and generate a signal capture on Rx1 when Tx2 is operating with gain $X$ and Tx1 is ON. Finally, we also
determine the available interference on the receiver considering a displaced frequency of 1999.6 MHz at Tx2. The above described procedure is repeated using this frequency and the value of \( X \) and signal is captured.

### 2.6 Dynamic spectrum access with SCMs

In this section, we describe a novel SCM-based deconfliction algorithm and spectrum access methods that leverage the proposed algorithm to deconflict spectrum use of networks with a large number of RF devices. The algorithm leverages CIL and SCM to dynamically determine the transmission parameters of devices, particularly their central frequency and transmission power levels, in order to achieve *aggregate compatibility*.

#### 2.6.1 Spectrum deconfliction algorithm

The deconfliction of spectrum use for networks having large number of RF devices is based on CT and computation of aggregate compatibility between new and existing set of devices. For our purposes, each network is composed of a single transmitter-receiver pair that has a designated wireless domain (WD) controller, which interacts with other WD controllers for spectrum access. We propose a novel **SCM-based Spectrum Deconfliction (SD)** mechanism as described in Algorithm 1 with key terms defined in Table 2.1. We further propose two spectrum access methods that use the SD algorithm, namely, 1) LCS - Logically Centralized Sequential spectrum access and, 2) LND - Local Neighborhood based Distributed spectrum access. In sequential spectrum access, we compute the aggregate interference from all Txs at each Rx and monitor it using a centralized variable \( Rx^{*}_{interf} \), which is later used to obtain the ideal power margin of new Tx (line 18 & 19 of Algorithm 1), whereas for distributed spectrum access aggregate compatibility is only ensured on a peer-to-peer basis.

The SD algorithm running at the corresponding WD controller requests for the SCMs from all existing (i.e. for sequential) or neighboring (i.e. for distributed) link pairs and initializes the following parameters: compatibility score \( (ct\_score = 0) \), total power margin \( (total_{PM} = 0) \) and max power margin \( (max_{PM} = 0) \). The score is used
to verify the compatibility of the new link with the entire system. In contrast, the
power margins are used to verify whether any small adjustments in the power level of
the new transmitter \((T_x)_n\) can make it compatible with the existing receivers.

As the WD controller performs CTs between the new \(R_x)_n\) and existing transmitters
\((T_x)_1...T_x)_{n-1}\), if there is compatibility, the WD controller computes and updates the
total aggregate interference \((total_{TP})\) at \(R_x)_n\). With the total aggregate interference
value, an aggregate evaluation of compatibility is performed. If \(R_x)_n\) is not compatible

\begin{algorithm}
\caption{Spectrum Deconfliction (SD) involving aggregate interference, frequency, and power adjustments at \(WD_n\).}
\begin{algorithmic}[1]
\IF{LCS}
\STATE \textbf{Input:} \(WD_n\) collects SCMs from all existing devices
\STATE \textbf{Initialize:} \(R^*_x_{interf} = [\]
\ELSE \textbf{LND}
\STATE \textbf{Input:} \(WD_n\) collects SCMs from assigned peer devices
\STATE \textbf{Output:} Deconflicted \((T_x)_n\) and \(R_x)_n\) pair
\STATE \textbf{Initialize:} \(power_{MT}\)
\STATE \textbf{Initialize:} \(ct_{score} = 0, total_{TP} = 0\) and \(max_{PM} = 0\)
\FOR{\((T_x)_n\) perform CT between \((T_x)_1...T_x)_i...T_x)_{n-1}\) do
\STATE \(total_{TP} = total_{TP} + P_m\)
\IF{compatible}
\STATE \(ct_{score} = ct_{score} + 1\)
\ELSE \(ct_{score} == n - 1\) \AND \(total_{TP} \leq P_{allowRx}_n\)
\STATE \textbf{Initialize:} \(ct_{score} = 0\) and \(R^*_x_{interf,n} = [\]
\STATE \(R^*_x_{interf[n]} = total_{TP}\)
\FOR{\((R_x)_n\) perform CT between \((R_x)_1...R_x)_j...R_x)_{n-1}\) do
\STATE \(R^*_x_{interf,n}[j] = P_{nj}\)
\STATE \(curr_{PM} = R^*_x_{interf[j]} + P_{nj} - P_{allowRx}_j\)
\STATE \(max_{PM} = \max(max_{PM}, curr_{PM})\)
\IF{compatible \AND current_{PM} \leq 0}
\STATE \(ct_{score} = ct_{score} + 1\)
\ELSE \(ct_{score} == n - 1\)
\STATE \(R^*_x_{interf} = R^*_x_{interf} + R^*_x_{interf,n}\)
\ELSE \(max_{PM} \leq power_{MT}\)
\STATE \textbf{Adjust} \((T_x)_n\) reference power based on \(max_{PM}\)
\STATE \textbf{Update} \(R^*_x_{interf}\) based on \(max_{PM}\) and \(R^*_x_{interf,n}\)
\STATE \textbf{Verify} if the new link is reachable or not
\ELSE \textbf{link not reachable}
\STATE \textbf{Move} \((R_x)_n\) \& \((T_x)_n\) frequency and recompute CTs
\ELSE \textbf{else}
\STATE \textbf{else}
\STATE \textbf{else}
\STATE \textbf{if} \(total_{TP} > P_{allowRx}_n\) \AND \(ct_{score} == n - 1\)
\STATE \textbf{Aggregate interference detected}
\STATE \textbf{Move} \((R_x)_n\) \& \((T_x)_n\) frequency and recompute CTs
\STATE \textbf{else}
\STATE \textbf{else}
\STATE \textbf{else}
\STATE \textbf{Update} \((R_x)_n\) \& \((T_x)_n\) SCM and setup the link
\end{algorithmic}
\end{algorithm}
Table 2.1: Key Parameters of Algorithm 1

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_{ij}$</td>
<td>Interference power at $j^{th}$ RX from $i^{th}$ TX</td>
</tr>
<tr>
<td>$R_{x_{interf}}^*$</td>
<td>Interference at each RX from all existing TXs</td>
</tr>
<tr>
<td>$R_{x_{interf,n}}^*$</td>
<td>Interference at each RX from TX$_n$</td>
</tr>
<tr>
<td>$total_{TP}$</td>
<td>Total power at RX$_n$ from all existing TXs</td>
</tr>
<tr>
<td>$power_{MT}$</td>
<td>Power margin threshold</td>
</tr>
<tr>
<td>$max_{PM}$</td>
<td>Max power margin from all RXs</td>
</tr>
<tr>
<td>$curt_{PM}$</td>
<td>Current power margin</td>
</tr>
<tr>
<td>$P_{allowRx_n}$</td>
<td>Allowable power at RX$_n$</td>
</tr>
</tbody>
</table>

with the existing transmitters or if the $total_{TP}$ is greater than the allowable interference power at $Rx_n$ (identified as $P_{allowRx_n}$), the Tx and Rx pair are moved to a different frequency/channel and the CTs start over (more details on the frequency assignment scheme will be discussed later). On the other hand, if $Rx_n$ is compatible with the existing transmitters, the WD controller verifies whether the corresponding $Tx_n$ is compatible with the existing receivers ($Rx_1...Rx_{n-1}$). To that end, it computes the interference caused by $Tx_n$ at the existing receivers and compares it with $max_{PM}$. If $Tx_n$ is compatible with the receivers already operating in the operational area, the WD controller sets up the new link. Otherwise, the WD controller tries to achieve compatibility by decreasing $Tx_n$’s power by no more than the value of the power margin threshold ($power_{MT}$). In case the power adjustment fails to achieve compatibility, the WD controller chooses a different frequency channel for the Tx and Rx pair and the compatibility tests start over. At the end of the algorithm, once aggregate compatibility is achieved, the WD controller sets up the link and updates the SCMs of $Tx_n$ and $Rx_n$ based on the frequency and power values found to achieve global (scenario wide) compatibility so that they are ready for future CTs. Please note, in the case of distributed spectrum access, aggregate compatibility is only ensured among the peer devices, and there is no centralized variable ($Rx_{interf}^*$ and $Rx_{interf,n}^*$) as in the case of sequential spectrum access to update the contribution of the existing and new transmitter to the aggregate interference seen by each receiver.

Regarding frequency assignment: Upon entering the operational area, all RF devices start with a default center frequency $f_c$ and, if needed, move to new frequencies in $\Delta f$.
increments until a compatible frequency for the operation of the new Tx/Rx pair is found. The objective here is to achieve high spectrum efficiency by minimizing the number of different frequency channels used. Thus, other approaches to determining $f_c$ and $\Delta f$ can be used, but they are left for future research.

2.6.2 LCS - Sequential spectrum access

In this kind of spectrum access method, WDs allocate spectrum resources sequentially as described in Algorithm 2. The assumption here is, existing devices are given high priority, and new devices are added in sequence to make them compatible with the existing system. At first, the links/WDs are prioritized either based on their sequence IDs or certain policies, and the deconfliction is performed using Algorithm 1. The interference by new and existing transmitters at each of the receivers is computed during the CT and stored centrally ($Rx_{intef}^*$) at the regional aggregator. It is then used by the SD algorithm to detect the presence of aggregate interference and to compute the ideal max power margin, required to limit the power of the new transmitter. More details on the evaluation of sequential spectrum access can be found in section 2.7.1.

\begin{algorithm}
\caption{Sequential spectrum access}
\begin{algorithmic}[1]
\Statex \textbf{Input}: Sequence IDs or priorities of WDs (set $S$)
\Statex \textbf{Output}: Deconflicted Tx/Rx pairs in a system
\For {each WD$_i$ over the sequence set $S$}
\State Obtain SCMs from all existing RF devices
\State Run \textbf{Spectrum Deconfliction} algorithm
\EndFor
\end{algorithmic}
\end{algorithm}

2.6.3 LND - Distributed spectrum access

The above proposed spectrum access method performs assignment sequentially based on CTs whenever a new pair of RF devices joins the system. But for a larger deployment, it introduces very high latency and a potential backlog of other low priority nodes waiting for their turn even if they are allowed to perform deconfliction. To this end, we propose LND, a fully distributed spectrum access method that uses Algorithm 1 to perform dynamic frequency and power adjustments. The coordination between multiple independent WDs is still performed using CIL but with additional messaging types required to support the distributed framework. The WD (or node) only maintains
local information, unlike sequential spectrum access, which uses a regional aggregator to observe the interference at each receiver. Table 2.2 provides a brief comparison between sequential and distributed spectrum access methods as proposed in this work. Each node computes an interference graph (or peers) during the bootstrapping phase from the exchange of beacons with the immediate 1-hop neighbors [59]. Then based on the node status and peer table, the WD independently performs spectrum assignment as shown in Fig. 2.6 (requires three time-steps to complete deconfliction), provided they are non-interfering with their respective peer groups.

![Figure 2.6: LND - High Level Overview.](image)

The design details of the LND are as follows:
Node status/state

This determines the current status of the node (WD), hence allowing the node to either wait or perform spectrum assignment. A WD can only be in one of the following states at a given time, and they include:

- **Unassigned**: Spectrum not assigned to the WD
- **Pre-assignment**: WD wants to perform deconfliction
- **Pause**: Temporarily pause pre-assignment (if any)
- **Running**: WD currently running SD algorithm
- **Assigned**: WD has completed spectrum assignment

Message exchange types

In addition to the CIL, the list of coordination message exchange types required to perform distributed deconfliction include:

- **Node_status_request()**: Message to obtain the current node status from its immediate 1-hop peers [NSR]
- **Node_status_response()**: Response message to reply with current WD status. If the WD is in the assigned state, an SCM is also sent along with node status [NSRp]
- **ACK()**: Message to acknowledge peers after reception of NSRp, endorsing that the node will be entering running state next [ACK]
- **Assigned()**: Message to inform peers on the completion of running SD algorithm [A]
- **Reset()**: Message to inform peer WDs to perform re-assignment if there are any local changes within its domain [RESET]
Timer types

We also introduce the following list of timers to prevent any contention between the nodes and also to avoid infinite waiting in the particular state.

- **NSR Timer**: Timer for NSRp and NSR from another peer to prevent any conflict and infinite waiting in a pre-assignment state.

- **ACK Timer**: Timer for ACK, required to prevent infinite waiting in the pause state if the peer WD fails to perform spectrum assignment.

- **Restart Timer**: Timer for the reception of the assigned message to prevent infinite waiting in the pause state after receiving an ACK from the node performing spectrum assignment.

Protocol details

We describe in detail the state transition of LND. The assumption here is: after a node performs an assignment, it never changes provided there is no change in spectrum demand and no `Reset()` message has been received.

1) **Pre-assignment**: The WD enters pre-assignment if it needs to perform spectrum assignment. At first, it sends NSR to its peers and starts the NSR timer. Next, it waits to receive NSRp (for the respective NSR) from all its peers. If any of the peers is in a running state, WD enters pause immediately. Moreover, if within the NSR timer window, another NSR from a peer WD is received, based on the conflict resolution procedures described later, WD enters pause to avoid further conflict in performing spectrum assignment. Finally, if all the NSRps have been received within the timer window and the WD is not in a pause state, it enters the running state to perform spectrum assignment (see Algorithm 3).

2) **Pause**: The WD enters a pause state when one of the following events occurs (see Algorithm 4):

   - WD in an unassigned state and received an NSR from the competing peer
Algorithm 3: Pre-assignment state transition

1. **Enter** PRE-ASSIGNMENT state
   - If WD needs to perform spectrum assignment
     - Send NSR and initialize NSRp timer
     - **while** *Timer expired or all NSRp received* **do**
       - **if** *NSR received from a peer node* **then**
         - Based on the conflict resolution procedure determine priority
           - **if** *Low priority* **then**
             - **Enter** PAUSE state, delete timer
       - Evaluate all the received NSRps
       - **if** *any NSRp == RUNNING* **then**
         - **Enter** PAUSE state, delete timer
       - **else**
         - Send ACK to the peer WDs
         - **Enter** RUNNING state

- WD in pre-assignment state and within the NSR timer window, it received an NSR from another high priority node (based on preferred conflict resolution procedure)

- WD in the pre-assignment state but its peer is currently in a running state

In the pause state, WD initializes the ACK timer and waits for acknowledgment from the peer (in running) performing the spectrum assignment. If the ACK message is received, it initializes the restart timer and waits for an assigned message. If the assigned message is received or any of the timers (ACK/Restart) expires, WD enters pre-assignment only if required to assign spectrum, or it stays unassigned.

Algorithm 4: Pause state transition

1. **Enter** PAUSE state
   - If WD in UNASSIGNED and received an NSR from a peer
   - If WD in PRE-ASSIGNMENT, and received an NSR from a high priority peer
   - If WD in PRE-ASSIGNMENT, and received NSRp == RUNNING from a peer
     - Initialize ACK timer
     - **while** *ACK timer expired or ACK received* **do**
       - Wait
       - **if** *ACK timer expired* **then**
         - **Enter** PRE-ASSIGNMENT or UNASSIGNED state
       - **else if** *ACK received* **then**
         - Initialize Restart timer
         - **while** *Restart timer expired or Assigned received* **do**
           - Wait
         - **Enter** PRE-ASSIGNMENT or UNASSIGNED state
3) Running: WD enters the running state, if it was in pre-assignment before and all the NSRps have been received thus allowing it to proceed further with the assignment. The WD executes the SD algorithm and finally issues an assigned message to its peer WDs when an ideal assignment is obtained (see Algorithm 5).

Algorithm 5: Running state transition

1. **Enter** RUNNING state
   - If WD in PRE-ASSIGNMENT and received NSRp != RUNNING

2. **Run** Spectrum deconfliction (Algorithm 1)
   - **if** Complete **then**
     - **Send** assigned message to the peer WDs
     - **Enter** ASSIGNED state

4) Example illustration: Considering Fig. 2.7, lets assume WD 1 is in a pre-assignment state performing spectrum assignment. It requests the node status from its peer WD 2 and WD 3. On receiving this request message, the peers currently unassigned will reply with NSRp having the current node status (but not SCMs) to WD 1. Next, WD 1 will acknowledge peers of it entering the running state and performing spectrum assignment. Since in this particular example, none of the peers of WD 1 are in a running state and no SCMs have been received, WD 1 will setup the link with default transmission parameters. Also, various timers as described before are initialized to prevent infinite waiting in the current state. Furthermore, if any of the peers are in an
assigned state, the WD will use its SCM to perform deconfliction using Algorithm.[1]
Finally, when complete, WD 1 issues an assigned message to its peers. Note that with LND, if there are any non-interfering peers willing to perform deconfliction, they can do so independently with any existing peer performing the same.

5) Conflict resolution: We introduce this procedure to avoid any disagreement with the peer performing spectrum assignment at the same time. The methods to determine high priority node/WD include:

- ID based: Select WD with lower ID or higher ID
- Max-peer: Select WD having the maximum number of peers
- Min-peer: Select WD having the least number of peers

In our work, we use Min-ID to resolve conflict among the peers. Evaluation with respect to Max-peer and Min-peer is left for future work.

2.7 Performance evaluation

To evaluate the performance of our proposed SD algorithm and spectrum access methods, we use a Python-based simulator that assigns Tx/Rx pairs to a determined location on a given operational area and then uses Algorithm.[1] to deconflict spectrum use. For sequential spectrum access, the simulation uses a discrete event-based framework with Tx/Rx pairs joining the system one after the other, whereas, for distributed spectrum access, we use MESA [60], an agent based modeling framework to represent each WD as an agent performing spectrum assignment on its own. To perform CTs, we leverage Octave code from the Spectrum Consumption Model Builder and Analysis Tool (SCM-BAT) [61], which interfaces with our python simulator using oct2py. The simulation parameters are summarized in Table 2.3.

The SCMs used in the simulator are similar to the SCMs from our experimental work described in section 2.5, except that, now we are limiting the transmit power such that the coverage radius for each Tx is around 100 meters. For simulation involving complex topology having a large number of RF devices, the position of the transmitters
Table 2.3: Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational Area</td>
<td>0.5 square mile</td>
</tr>
<tr>
<td>Min Tx-Tx separation</td>
<td>10 m</td>
</tr>
<tr>
<td>Tx-Rx separation</td>
<td>Uniform(10m, 100m)</td>
</tr>
<tr>
<td>Frequency shift (Δ f)</td>
<td>1 MHz</td>
</tr>
<tr>
<td>Power margin threshold</td>
<td>3 dB</td>
</tr>
<tr>
<td>Number of trials</td>
<td>100</td>
</tr>
<tr>
<td>Python version</td>
<td>3.8</td>
</tr>
<tr>
<td>Noise floor</td>
<td>-114 dBm</td>
</tr>
<tr>
<td>Modulation</td>
<td>BPSK</td>
</tr>
<tr>
<td>Bitrate</td>
<td>0.5M</td>
</tr>
<tr>
<td>Machine configuration</td>
<td>Intel i7-4790 (3.60 GHz)</td>
</tr>
</tbody>
</table>

is chosen uniformly at random in a 0.5 square mile area, subject to the constraint of a minimum distance of 10 meters between any two transmitters. Moreover, the separation between a transmitter and the associated receiver is chosen uniformly at random in the interval between 10 and 100 meters.

Agent based modeling framework for distributed deconfliction:

Our agent based simulation framework for distributed spectrum access mainly consists of two classes, namely, **Model** and **Agent** as shown in Fig. 2.8. We use MESA [60], a Python3 based modeling framework that allows us to quickly create agents and models using built-in core components and custom modules. The model class initializes model level parameters and variables such as the number of agents (or WDs), deploys/activates the agents within a deployment area, and also advances the simulation using the `Step()` module. In contrast, the agent class initializes agent level parameters and variables such as node status, frequency usage, Tx reference power adjustment,
etc. Using the \texttt{Step()} module present in the agent class, the WD stages any necessary changes required to resolve conflict with other peers performing spectrum assignment and finally uses \texttt{Advance()} module to apply the staged changes. There’s also a data collector module that collects model and agent level data, and computes global level system performance metrics such as throughput, compatibility error, channel usage, etc. at every time step as the simulation progresses. We next present our results for sequential and distributed spectrum access methods considering discrete event based and ABM simulation frameworks respectively.

### 2.7.1 Sequential spectrum access

In this subsection, we discuss the simulation results for the spectrum deconfliction performance of sequential spectrum access (LCS, Algorithm 2) against the \textbf{baseline algorithm} \cite{19} which is a variant of sequential spectrum access but only adjusts center frequencies to deconflict spectrum use, in terms of its spectrum resource usage and computation time, for a scenario involving a large number of RF devices (see Table 2.3 for detailed simulation parameters).

![Figure 2.9: Histogram showing the number of channels used by Algorithm 2 to deconflict networks of different sizes.](image)
Table 2.4: Performance counters of sequential spectrum access averaged over 100 trials for different network sizes

<table>
<thead>
<tr>
<th>Links</th>
<th>10</th>
<th>20</th>
<th>50</th>
<th>80</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. # of power adjustments</td>
<td>1.09</td>
<td>6.93</td>
<td>29.46</td>
<td>52.74</td>
<td>69.91</td>
</tr>
<tr>
<td>Avg. # of non-reachable events</td>
<td>0.04</td>
<td>0.29</td>
<td>1.9</td>
<td>3.65</td>
<td>5.86</td>
</tr>
</tbody>
</table>

Spectrum usage

We evaluate the spectrum usage of Algorithm 2 by considering the number of frequency channels used. As shown in Fig. 2.9 for 100 trials, the maximum number of channels required to configure 10 links is 5 as compared to 17 channels required to configure 100 links. The mode number of channels required for 10, 50 and 100 links are 3, 8 and 13, respectively. We also observe that, to support a large number of links (100 in our case), we require only a few channels (17 channels max) and this is mainly due to the Tx power adjustment. Table 2.4 shows some of the performance measures we capture which highlight the importance of the power adjustments. We limit the Tx power of new transmitters based on the power margin threshold and the power margin obtained from the CT. If a CT indicates that the Tx power of a transmitter will lead to compatibility with all previous receivers if reduced by an amount that does not exceed the value indicated by the power margin threshold, then the adjustment is applied. Otherwise, the transmitter moves to a different frequency/channel. The average number of Tx power adjustments for 10 and 100 links over 100 trials is found to be 1.09 and 69.91, respectively, highlighting the importance of Tx power adjustments on reducing spectrum usage. For scenarios with a large number of links, the Tx power adjustment step can (occasionally) lead to a transmitter not being able to reach its intended receiver. This forces the use of a different frequency channel for the Tx/Rx pair and the need to perform CTs again. The average non-reachable link events for 100 links over 100 trials is found to be 5.86, requiring higher computation time to deconflict the spectrum as described next.

We also compare the spectrum usage of sequential and the baseline algorithm considering 100 links. In Fig. 2.10 when looking at the mode/max values, higher number
Table 2.5: Presence of aggregate interference averaged over 100 trials for different network sizes

<table>
<thead>
<tr>
<th>Links</th>
<th>10</th>
<th>20</th>
<th>50</th>
<th>80</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0.13</td>
<td>0.64</td>
<td>6.63</td>
<td>20.94</td>
<td>32.99</td>
</tr>
<tr>
<td>Sequential</td>
<td>0.05</td>
<td>0.81</td>
<td>7.18</td>
<td>19.36</td>
<td>29.89</td>
</tr>
</tbody>
</table>

of channels are required for the baseline case as compared to our proposed scheme, since the baseline algorithm just operates on assigning frequencies without adapting Tx power levels. The other important metric in the evaluation is the presence of aggregate interference in the system. We defined a counter for aggregate interference events that is incremented only when a new receiver is compatible with all existing transmitters on a one-to-one basis, but not when taking into account the interference caused by all existing transmitters. Table 2.5 provides the average number of times (cases) that aggregate interference effects led to an incompatibility decision in 100 runs/trials. In each of those cases, the device would need to move to a different frequency to be protected from aggregate interference effects. The results obtained indicate that as the number of devices grows, the effects of aggregate interference and adjacent channel power leakage are well captured with the use of SCMs. A higher number of aggregate interference events is observed for the baseline algorithm as compared to sequential spectrum access, due to the presence of a large number of channels being used at full power. Moreover, the aggregate interference highly depends on the Tx/Rx location coordinates and the channels being used. Further, sequential without power margin threshold (no PMT) when compared against the baseline algorithm, achieves a better spectral efficiency (see Fig. 2.10) due to the more flexible power assignment and fewer number of aggregate interference cases.

**Computation time**

We evaluate the time required to deconflict the spectrum use, i.e., the time to determine compatible parameters for every Tx/Rx pair in the operational area. The computation time does not include: (i) pauses between new link pair instances (recall that pairs join the system sequentially); (ii) time to turn on the RF devices, which depends on the
radios being used; (iii) SCM transmission times, which usually take a few milliseconds. Fig. 2.11 shows the mean computation time for networks with different numbers of Tx/Rx pairs. For each simulation setup, 100 trials/runs are executed. We observe that to configure a single new link (Tx/Rx pair) in a scenario composed of 20 pre-existing links sequential with a 3dB PMT takes (on average) 0.37s as compared to the 3.25s required in a scenario with 100 pre-existing link-pairs. The increase in computation time with the increase in the total number of Tx/Rx pairs is expected and it is mainly due to the larger number of CTs that need to be performed. To setup a single new link in a scenario that already has 100 operational link-pairs, the baseline algorithm takes (on average) 3.51s. The increase in computation time with the increase in the number of links for the baseline algorithm is due to the large number of CTs needed to perform deconfliction over a larger number of channels to minimize aggregate interference. This is in contrast to the sequential no PMT case where the number of channels over which to perform CTs is lower. Overall, the computation times can be significantly reduced by using more powerful CPUs and with additional algorithm optimizations.
Sequential spectrum access enhancement

To further evaluate and enhance the performance of the spectrum deconfliction achieved by sequential spectrum access, we introduce a distance parameter (D), where we only perform CTs with the devices that are within a distance D from each other, rather than with all existing devices. When setting D=100m or D=300m, the computation time is significantly reduced as compared to the exhaustive case. As an example, having 100 links, the average computation time for D=100m and D=300m is 0.151s and 0.881s, respectively as compared to 3.25s required for the non-distance constrained version of the sequential algorithm (see Table 2.6). For D=500m, the spectrum usage and the impact of aggregate interference is similar to that of the exhaustive solution (Algorithm 2, see Fig. 2.12), implying that for up to 500m, the aggregate interference from existing devices that is impacting the new device and is well captured with the help of SCMs. In some cases, the use of the distance parameter led to compatibility errors and we compute this error with respect to the exhaustive case where the evaluation of compatibility is done against all existing devices. The results are shown in Table 2.6. We see that the compatibility error decreases with the increase in D, since more devices will be considered in CT computations allowing to accurately capture the impact of
Table 2.6: Global compatibility error (%) and average computation time per link (in seconds) due to the use of the distance (D) parameter in sequential spectrum access (Algorithm 2), averaged over 30 random trials.

<table>
<thead>
<tr>
<th></th>
<th>20 Links</th>
<th>50 Links</th>
<th>80 Links</th>
<th>100 Links</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Error (%)</td>
<td>Time (sec)</td>
<td>Error (%)</td>
<td>Time (sec)</td>
</tr>
<tr>
<td>D = 100m</td>
<td>13.5</td>
<td>0.022</td>
<td>22.33</td>
<td>0.065</td>
</tr>
<tr>
<td>D = 200m</td>
<td>2.0</td>
<td>0.053</td>
<td>2.0</td>
<td>0.175</td>
</tr>
<tr>
<td>D = 300m</td>
<td>1.66</td>
<td>0.103</td>
<td>0.133</td>
<td>0.321</td>
</tr>
<tr>
<td>D = 500m</td>
<td>0</td>
<td>0.205</td>
<td>0</td>
<td>0.662</td>
</tr>
</tbody>
</table>

aggregate interference at each receiver. Also, a high compatibility error is observed for D=100m. Intuitively, the error drops significantly beyond that distance mainly because each transmitter’s coverage radius is roughly 100m. The results indicate that based on the sensitivity of the new device/receiver that is trying to operate in the area, and assuming a constant maximum level for the transmit power for all transmitters, an appropriate D parameter should be selected to reduce the number of compatibility computations and minimize the probability of error from such reduction.

Figure 2.12: Histogram showing the number of channels used - D constrained vs. non-constrained sequential algorithm (100 links & 100 trials).

**Channel separation**

Finally, we evaluate the spectrum usage of the sequential access method by considering a wider channel separation where every new channel has its center frequency 2 MHz away from the last channel used. As shown in Fig. 2.13 for 100 trials, we observe that,
using 2 MHz separation results in a lower number of used channels, but at the cost of wider total spectrum occupancy as compared to channels having 1 MHz separation (see Fig. 2.9). Also, lower computation times (2.39s, 100 links) are observed mainly due to channels being well apart resulting in lower adjacent channel (aggregate) interference and fewer power adjustment events with non-reachable link events, thus allowing the devices to find optimal parameters faster with a lower number of compatibility tests.

Figure 2.13: Histogram showing the number of channels used by the sequential algorithm to deconflict using 2 MHz channel separation.

2.7.2 Comparison of Sequential and Distributed spectrum access

In this subsection, we discuss the comparative performance analysis of sequential and distributed spectrum access considering fixed and complex topology scenarios.

Fixed topology scenario

Firstly, we evaluate the spectrum usage of the SD algorithm against sequential and distributed spectrum access methods by considering a simple topology composed of 5 WDs each having a single $Tx-Rx$ link-pair as shown in Fig. 2.14. The proposed SD algorithm requires 3 channels and the usage is found to be the same for both sequential and distributed spectrum access methods. Table 2.7 provides various performance metrics such as global compatibility error (CE), system throughput, and step count.
(lower bound), evaluated against these access methods. The Shannon throughput per link is computed using the equations 2.1, 2.2, and 2.3 and then aggregated across all the links as reported in Table 2.7. We observe the compatibility error to be 0 \% for sequential spectrum access, but at the cost of high step count since all the devices are considered for deconfliction. The step count improves with distributed approach, as only peer devices are considered and some of the devices will be performing deconfliction in parallel, but with a very high penalty on global compatibility error. This error can be further improved by reducing the Tx power by a very small value of 0.1dB in addition to PM for the devices performing power adjustments. The reason behind

<table>
<thead>
<tr>
<th>Spectrum access methods</th>
<th>CE</th>
<th>Throughput (Mbps)</th>
<th>Step count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distributed (p=100m)</td>
<td>80%</td>
<td>33.87</td>
<td>2</td>
</tr>
<tr>
<td>Distributed (p=200m)</td>
<td>40%</td>
<td>36.98</td>
<td>4</td>
</tr>
<tr>
<td>Distributed (p=100m, 0.1 dB adj)</td>
<td>40%</td>
<td>33.85</td>
<td>2</td>
</tr>
<tr>
<td>Distributed (p=200m, 0.1 dB adj)</td>
<td>0%</td>
<td>36.98</td>
<td>4</td>
</tr>
<tr>
<td>Sequential (all devices)</td>
<td>0%</td>
<td>36.88</td>
<td>5</td>
</tr>
<tr>
<td>Baseline (all devices)</td>
<td>0%</td>
<td>40.09</td>
<td>5</td>
</tr>
</tbody>
</table>

Figure 2.14: Fixed topology setup.

Table 2.7: Performance evaluation of SD algorithm and other spectrum access methods considering fixed topology scenario.
doing this is to prevent the spectrum mask of a transmitter not just touch the underlay mask of other receivers from applied power adjustments, but at least having a $\delta$ dB (in addition) difference between them to account for any potential interference which gets missed out by only considering devices in a peer-to-peer basis, unlike sequential access method where a centralized variable is used to monitor all the interferences. The distributed spectrum access having p=200m (peer) and 0.1 dB power adjustment outperforms other distributed schemes, as the compatibility error is found to be 0% and throughput equaling sequential spectrum access algorithm. Based on the sensitivity of the receiver on how much interference it can tolerate, an appropriate interference graph ($p$ value) should be selected to achieve high spectral efficiency.

$$SINR_i = \frac{Power_{signal_i}}{NF + \sum_{n=0, n\neq i}^{N-1} Interf_{ni}}$$ (2.1)

$$NF = -174 \text{ dBm} + 10 \times \log_{10}(B)$$ (2.2)

$$C_i = B \times \log_2(1 + SINR_i)$$ (2.3)

Where:

$C_i$ = Shannon channel capacity of link $i$ (Mbps)

$B$ = Channel bandwidth (MHz)

$SINR_i$ = Signal to noise plus interference ratio of link $i$

$Power_{signal_i}$ = Signal power of link $i$

$Interf_{ni}$ = Observed interference at the Rx of link $i$ from Tx $n$

$NF$ = Noise figure

$N$ = Total number of links

Complex topology scenario

We further compare sequential and distributed spectrum access by considering a more complex topology and using the simulation parameters as listed in Table 2.3. The evaluation is based on convergence (for distributed), step count/computation time,
((a)) Convergence of LND

((b)) WD state transition

Figure 2.15: Distributed spectrum access (100 links & 1 trial) - (a) Convergence for different peer values (b) Histogram depicting WDs in each state over time (p=200m).

algorithm efficiency, and channel usage. Based on the results provided in section 2.7.2 for simple topology, a 0.1 dB power adjustment is applied for the analysis of distributed spectrum access method.

1) LND convergence: At first, we evaluate the convergence of LND distributed protocol. From Fig. 2.15(a) for a network composed of 100 pre-existing links and a single trial, we observe that convergence highly depends on the selection of p value. The WDs having large number of peers (p is high) require higher number of steps to perform assignment, as more devices are considered for deconfliction. The step count computed in our evaluation represents a lower bound for an ideal scenario based on the following set of assumptions: (i) good fiber connection between the WDs having low latency and zero packet loss; (ii) all the WDs show up simultaneously with default SCM parameters, and each having the intent to use spectrum at every time step. As an example, considering a system of WDs with p=100m, requires 5 time steps as compared to 13 steps required to perform deconfliction with p=200m. Furthermore, Fig. 2.15(b) depicts the operation of LND at each time step considering 100 links/WDs and p=200m. Since all the WDs have the intent to perform assignment at the same time, we observe a large number of them unable to perform deconfliction due to the conflict between the
Table 2.8: Comparison of LND step count with the chromatic number of standard graph coloring algorithms from Python NetworkX package, for a network of different link configurations (p=200m, 100 trials).

<table>
<thead>
<tr>
<th>Links</th>
<th>10</th>
<th>20</th>
<th>50</th>
<th>80</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>2.92</td>
<td>4.52</td>
<td>8.26</td>
<td>11.74</td>
<td>13.71</td>
</tr>
<tr>
<td>DSATUR</td>
<td>2.96</td>
<td>4.35</td>
<td>7.46</td>
<td>10.36</td>
<td>12.09</td>
</tr>
<tr>
<td>BFS</td>
<td>2.76</td>
<td>4.37</td>
<td>7.79</td>
<td>10.93</td>
<td>12.81</td>
</tr>
<tr>
<td>DFS</td>
<td>2.87</td>
<td>4.31</td>
<td>8.18</td>
<td>11.37</td>
<td>13.51</td>
</tr>
<tr>
<td>LND</td>
<td>2.85</td>
<td>4.42</td>
<td>8.37</td>
<td>11.49</td>
<td>13.76</td>
</tr>
</tbody>
</table>

peer nodes and hence entering *pause* state. The situation improves with time, and the WDs enter *running* state, and once assigned, WD changes its state to *assigned*. On average, around 8 WDs perform assignment simultaneously at every time step using our proposed LND method as shown in Fig. 2.15(b) demonstrating the parallelism imposed by our protocol. Moreover, step count is observed to be similar to the chromatic number of a graph when evaluated against the standard graph coloring algorithms as shown in Table 2.8. Graph coloring algorithms from Python NetworkX [62] package are used to color the graph and compared to sequential spectrum access which requires O(N) number of steps, our proposed distributed protocol only requires O(Chromatic-number) of steps to perform deconfliction.

2) Step count and computation time: We evaluate step count and time required to deconflict spectrum use, i.e., the time to determine compatible parameters for sequential and distributed spectrum access in an operational area. The assumptions to obtain computation time as given in section 2.7.1 is considered here. Fig. 2.16(a) and Fig. 2.16(b) provide the step count and computation time per link averaged over 100 trials/runs for networks having different numbers of Tx/Rx pairs respectively. We observe that the distributed spectrum access requires a lower number of steps as compared to sequential since multiple WDs will be performing assignments simultaneously among their non-interfering peers. The average step count for p=100m and p=500m is observed to be 6.84 and 40.77 steps, respectively. We also observe that to configure a single link in a scenario composed of 20 pre-existing links, sequential requires (on average) 0.37s as compared to 3.25s required for a network of 100 pre-existing link-pairs. Distributed, on the other hand, requires lower computation time; a gain corresponding
Figure 2.16: Average step count and computation time per link for sequential and distributed spectrum access to deconflict networks of different size (100 trials).
new receiver, an appropriate interference graph (p/peer value) should be determined to reduce the number of CT computations and minimize probability of error.

b) System throughput: We further compute Shannon throughput per link considering 1 MHz channel bandwidth, as reported in Table 2.9. The throughput obtained is directly proportional to the channel usage, and we observe a high throughput for baseline since it operates only in the frequency domain with high Tx power. Considering sequential spectrum access without PMT, the throughput is 318.71 Mbps for a network of 100 links. Without PMT, the Tx power of the new transmitter is reduced to an ideal relatively low setting and it represents the maximum potential spectrum gain that can be achieved with power adjustments. Subsequently, the throughput obtained is less as compared to other schemes when the 3dB threshold is considered. Moreover, distributed spectrum access achieves throughput performance comparable to sequential and baseline access methods.

c) Channel usage: Finally, we evaluate the spectrum usage of sequential and distributed spectrum access based on the number of channels used. For distributed spectrum access, the channel usage depends on the selection of p (see Fig. 2.17(a) and Fig. 2.17(b)). With p=100m and 100 trials, the mode number of channels required for
Table 2.9: Global compatibility error (\%), system throughput, mean channel usage, and average computation time (in seconds) per link for various spectrum access methods, averaged over 30 random trials.

<table>
<thead>
<tr>
<th>Methods</th>
<th>20 Links</th>
<th>50 Links</th>
<th>100 Links</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CE (%)</td>
<td>TPT (Mbps)</td>
<td>CH (Avg.)</td>
</tr>
<tr>
<td>Distributed (p=100m)</td>
<td>12.16</td>
<td>91.93</td>
<td>3.9</td>
</tr>
<tr>
<td>Distributed (p=200m)</td>
<td>0</td>
<td>96.23</td>
<td>4.43</td>
</tr>
<tr>
<td>Distributed (p=500m)</td>
<td>0</td>
<td>92.96</td>
<td>4.46</td>
</tr>
<tr>
<td>Sequential (no PMT)</td>
<td>0</td>
<td>88.97</td>
<td>4.4</td>
</tr>
<tr>
<td>Sequential</td>
<td>0</td>
<td>92.76</td>
<td>4.26</td>
</tr>
<tr>
<td>Baseline (only freq.)</td>
<td>0</td>
<td>94.77</td>
<td>4.23</td>
</tr>
</tbody>
</table>
10, 50, and 100 links are 2, 7 and 12, respectively. However, the global compatibility error (as shown in Table 2.9) is high for p=100m, implying the need to select a larger peer group to reduce interference and error. We also observe the channel usage of distributed spectrum access with p \geq 200m is similar to sequential (see Fig. 2.9), which means, upto 200m the aggregate interference from nearby peer devices is impacting the channel selection and is well captured with the help of SCMs. Additional results on the channel usage for distributed access are avoided due to its similarity with the sequential spectrum access method.

2.8 DSA experimentation on ORBIT/COSMOS testbed

Our experimentation framework for dynamic spectrum access interactions was built on top of the COSMOS wireless testbed [47, 54]. As described earlier, the framework aims to enable wireless networks to exchange messages in order to coordinate and synchronize their spectrum access, it also facilitates the execution of the computations necessary to determine available spectrum resources and avoid interference events with other networks present in the same spectrum.

In the experiment, there are three wireless networks that have transmitters and receivers near each other’s local area. The goal is for the networks to dynamically configure their wireless transmission characteristics, in this case, their central frequency of operation so that there is no harmful interference between the devices of different networks. Each network contains only a single transmitter and a single receiver. Transmitters and receivers are implemented using NI USRP X310 and B210 devices, while on the control plane, WD controllers operate on separate nodes with CIL message exchanges conducted via out of band links between the WD controller nodes. Additionally, all transmitters use the same spectrum mask and attempt transmission using a 1 MHz wide channel. The radio node acting as a spectrum sensor/monitor and used for visualization is sampling captured signals at 10 Msps. The decision-making logic (whether to transmit and which frequency and power to use) of the network is implemented as part of the WD node, and GNU Radio scripts are utilized to implement the physical layer functionality, including packetization and USRP over-the-air transmission.
Table 2.10: Experimentation Settings

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central Freq</td>
<td>2.0 GHz</td>
</tr>
<tr>
<td>No. of Channels</td>
<td>3</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>1 MHz</td>
</tr>
<tr>
<td>Modulation</td>
<td>BPSK</td>
</tr>
<tr>
<td>Bitrate</td>
<td>0.5M</td>
</tr>
<tr>
<td>Gnuradio</td>
<td>v3.7</td>
</tr>
<tr>
<td>USRPs</td>
<td>x310s and b210s</td>
</tr>
</tbody>
</table>

The experiment showcases a simple scenario. Fig. 2.18 shows the scheme of the experiment – the networks with their WDs, node IDs of the USRPs utilized, and the established interaction links. In the beginning, there are no networks present in the spectrum. At time $t_1$, Network 1 joins, and since the entire spectrum is available, it is free to select any channel. A 1 MHz channel centered around 2 GHz is selected. After that, at time $t_2$ Network 2 joins the domain. Its default frequency setting is 2 GHz, the same central frequency used by Network 1. The interaction language server (Regional Aggregator) determines that the networks are in the same wireless collision domain, and notifies Network 1 that a peer is now present. Networks 1 and 2 establish a peering relationship, and immediately exchange their SCMs. After performing the compatibility checks, Network 2 determines that it can not use the intended frequency, and finds an alternative optimal center frequency – 1999 MHz. Similarly, at time $t_3$, Network 3 joins, with the assistance of the Regional Topology Manager establishes
connections with Networks 1 and 2, and after exchanging a set of SCM messages and performing its spectrum compatibility calculations determines that the optimal center frequency to use is 2001 MHz. Fig. 2.19 shows a summary of the SCM information that the controller node at Wireless Domain 3 (WD3) had to process to carry out the compatibility tests necessary to locate a central frequency where its Transmitter (Tx3) could operate without causing interference to receivers already present in the environment and where its receiver (Rx3) could also operate without being interfered by existing transmitters. Fig. 2.20 shows the fosphor visualization of the frequency selection of three networks that join the system at times $t_1$, $t_2$, $t_3$. 
2.9 Related work

Various dynamic spectrum access methods have been proposed and evaluated in the past. The work described in [63] presents an opportunistic protocol for spectrum access coordination between independent networks operating with different wireless protocols. This opportunistic protocol included a central Cognitive Radio (CR) terminal which assigned spectrum aiming to establish fairness based on data flows. In [64], a simple message exchange protocol (in many ways similar to CIL) is presented. This message exchange protocol, named Common Spectrum Coordination Channel, operated in a separate narrow frequency band to allow networks to exchange simple messages to announce their spectrum usage. The evaluation of its performance was based on ns-2 simulations. The work in [65] showed a semantic-based algorithm that used FFT analysis and energy detection with semantic reasoning to determine available frequency bands for transmission. A distinctive feature of this paper is that it included a real-world implementation of its algorithm using OpenAirInterface (OAI). Another semantic approach is shown in [66]. Spectrum coordination in 5G is considered in [67], where the authors introduced a virtual currency-based non-cooperative negotiation protocol for spectrum access.

In [68], the authors proposed SMAP, a policy-driven distributed spectrum management architecture which uses an aggregated radio map to exchange spectrum information among peers for spectrum assignment. The authors of [68] didn’t provide details on the syntax and schematics of the radio map and their evaluation was restricted to a small topology in which nodes were only allowed to adjust their frequencies. Authors in [69],[70] presented a Radio Environment Map (REM) based spectrum access architecture to protect primary incumbents and share the available spectrum. The REMs are constructed from sensor measurements and are used to determine channel availability and also to estimate interference levels at each location of interest. The architecture is based on centralized storage databases and the methods to determine compatibility among incumbents and secondary users are not standardized. These methods depend on the tools used to perform spectrum use/occupancy analysis and their results are
often debatable by each of the parties involved [43].

With a significant amount of research efforts aimed at enabling the coexistence of heterogeneous networks in the same spectrum bands, it is important to note that the bulk of the work has been theoretical, occasionally backed by simulations (often in MATLAB or a network simulation framework such as ns-3), with few scientific papers that evaluate their findings using practical implementations. Our work aims in part to fill that gap. We evaluate the performance, scalability, and feasibility of our proposed SCM-based DSA for coordinating spectrum use in dynamic and dense communication environments using a custom simulation framework. We further validate our solution using a simple spectrum access framework on the COSMOS testbed [47, 54]. We implement a coordination protocol based on CIL, a prime example of a protocol that was already successfully utilized for spectrum access coordination during DARPA SC2, which in our case is used to exchange SCM messages to enable spectrum resource use coordination.

2.10 Summary

In this chapter, we presented a novel mechanism for performing spectrum coordination using SCMs which offer a standardized means to capture the spectral, temporal and spatial characteristics of spectrum use of RF devices and systems. We utilize a modified version of the interaction language CIL developed by DARPA to enable the exchange of SCM messages between wireless networks. We propose a novel SCM based Spectrum Deconfliction (SD) algorithm that takes into consideration aggregate interference and makes use of frequency and power adjustments to deconflict spectrum use. Using the proposed SD algorithm, we develop sequential and distributed spectrum access methods and evaluate using a simulation tool where SCM based spectrum use deconfliction for large-scale scenarios was modeled. The simulation results and the experimental validation demonstrate the feasibility of our SCM based spectrum use deconfliction technique in performing fine grained spectrum assignments at scale.
Chapter 3
Network Assisted Distributed Mobility Management

3.1 Introduction

Seamless handover (or handoff) between various radio access networks has been a widely discussed topic in recent years. It is increasingly relevant in the current 5G era as the operators are rapidly upgrading their network infrastructure to support latency-critical applications and enhance user experience. The increased network density and higher user mobility is expected to cause frequent handovers between various access networks, and therefore will make UE (User Equipment) attachment and packet forwarding complex due to the hierarchical gateway based architecture of existing mobile core networks based on the 3GPP specifications.

Future internet architectures (FIAs) \cite{48, 49} based on named-object techniques have the potential to enable seamless mobility by dynamically mapping device/object names to their respective network addresses (NAs). The capabilities of these architectures to handle network service interruptions using multihoming and in-network dynamic packet rebinding are still unexplored. In this chapter, we propose two handoff schemes as an initial attempt at solving challenges arising from multiplicity of networks in a single geographical domain, specifically targeted towards seamless handover in heterogeneous networks as proposed by beyond 5G architecture. The major contributions are as follows:

- We demonstrate a tunnel-less set-up to achieve heterogeneous handover using a clean slate flat network architecture having name and address separation.

- We propose two new handover schemes, \textit{Hard Handoff with Rebinding} and \textit{Soft Handoff with Multihoming} in named-object based network architecture.
We evaluate the handoff schemes with proof-of-concept implementation in the ORBIT testbed as well as simulation by considering real user mobility traces and observe the impact of packet loss and handoff latency on performance metrics such as round trip time (RTT) and throughput.

### 3.2 Name Based Approach For Mobility Support

Supporting seamless mobility services in telecom networks is achieved via mobility management protocols including authentication, roaming and handover \[71,72\]. A UE switches from one AP/eNB (Access Point/eNodeB) to another if the neighboring cell has better channel quality as reported through signal quality measurements (RSSI, RSRP, RSRQ, SINR, SNR, CIR) and UE specific parameters (location, velocity, and direction of movement) \[73\]. With conventional roaming in WLAN, the UE continuously monitors the signal strength (RSSI – Received Signal Strength Indicator) at the currently associated AP. If the RSSI drops below a certain threshold, the UE scans for available APs on all the channels. The association takes several seconds due to the large number of channels to be scanned as shown in the example in Fig. 3.1 obtained using emulation on the ORBIT \[45\] testbed and \textit{wpa_supplicant} \[74\] roaming package. The handoff disconnection period is 2.92s. Once the best AP is found, the UE disconnects from the previous AP, and attaches with the newly discovered AP post-authentication \[75\] by acquiring a new IP address. The old IP address can be reused during handoff if the APs are managed by a centralized entity like the SDN controller \[76\], which introduces additional control plane overhead and decision time. The break-before-make approach therefore, typically interrupts an on-going session. A similar approach is followed in LTE during X2 handover (IP address of the UE is maintained), but a tunnel is set up between the source eNB and the target eNB to forward packets before using the new bearer connecting S-Gw (Service-Gateway) to target eNB for the downlink (DL) traffic.

The problems arising due to constant monitoring as well as allocation of IP addresses due to UE mobility can be addressed in clean-slate network designs by utilizing the concepts of globally unique identifiers (GUID) which are dynamically mapped to network
addresses (NAs). This concept of named-based networking has been extensively studied in \cite{77, 78} wherein the name service layer acts like a narrow waist of protocol stack. Each end host or content object is identified with the help of a GUID which remains unchanged during the entire lifetime of a UE session. GUIDs are anchored to the network with the help of NA. The name to address mappings are resolved dynamically by the GNRS (global name resolution service) which is a globally distributed but logically centralized mapping database, periodically updated and queried by the NEs (network entities). This separation of name and address space handles mobility related events, and thus is best suited for handover, specifically when dealing with multiple NEs and RATs in a heterogeneous network. In such a named-object based architecture, packets are directly addressed using GUIDs and SIDs (service IDs). The SID allows the NEs to take an action on a packet, e.g., storage or scheduling. The routers are storage capable and implement store or forward logic on a per hop basis to achieve reliability \cite{79}.

This work builds on an implementation of name-based FIA (MF – MobilityFirst \cite{49}) to overcome limitations of the TCP/IP architecture concerning mobility (handover), and particularly focuses on re-binding and multihoming as the key enablers for seamless mobility services.
3.3 Seamless Handover Techniques

In this section, we propose two handover schemes supporting seamless handoff using the MF architecture – hard handoff with rebinding and soft handoff with multihoming. The AP/eNBs (Base Stations - BS) run a fully decentralized version of the MF protocol thereby eliminating the need of setting gateways and tunnels. The flat network routing is supported by GNRS which provides current GUID→NA bindings. It is based on the DMap design which maintains K copies of the bindings in the network, ensuring single overlay hop-path with good storage distribution while providing lower lookup latencies as low as 86ms for $K = 5$.

This flat architecture facilitates easy plug-and-play deployment of heterogeneous network avoiding the need of additional authentication mechanism while supporting seamless handovers. The system implementation details of key MF components involving GNRS, MF routers and Hoststack can be found in [29]. We use the LTE (OpenAirInterface no-S1 with L2TP tunnel) [81] and Wi-Fi (802.11g) stack for experimental evaluation using MF.

The following features are enhanced to achieve seamless handover in a heterogeneous network, (a) **In-network rebinding** : the intermediate routers are capable of resolving packets to a new UE address using name resolution server. (b) **Multihoming** : enables UE devices to receive packets on multiple interfaces if available without notifying end hosts as the intermediate routers are capable of handling packet replication and scheduling between these multiple interfaces.

**Integrating with IP network:** The handoff schemes discussed above can be integrated with the current IP based network using an overlay, by having a naming layer over IP or UDP for incremental deployment [82]. Overlays are useful for clean slate deployment as the approach allows use of the existing network infrastructure.

### 3.3.1 Hard Handoff with Rebinding

In the hard handoff (break-before-make) scenario, mobility related events are handled by the UE and it always attaches to the best BS based on channel quality measurements.
In such traditional schemes, the undelivered packets due to poor connectivity are either lost or buffered at the edge router. Thus, these packets must be forwarded to the new BS through IP tunnels, or the end host can request retransmission using a transport layer solution (TCP). These mechanisms induce additional network overhead particularly when the number of devices attached to the network is large.

The rebinding feature (given sufficiently large router buffer size) makes it possible to design a zero-packet loss system by querying the updated NA for a given GUID, thus allowing a router to dynamically reroute packets. The UE is not aware of packet rerouting except for listening to the new BS channel. Fig. 3.2 gives an overview of handover procedure using rebinding support. Considering a Wi-Fi AP as an example, the supporting timing diagram involving handover is shown in Fig. 3.3. Initially the UE is attached to the AP1 using the default 802.11 attach procedure. AP1 inserts a new binding ($GUID_{UEx} \rightarrow NA_{AP1}$) into the GNRS which is periodically queried and updated by itself as well as other NEs. On successful attachment, a data flow path is created between the UE and AP1. The data flow continues using AP1 until the RSSI is less than a specified threshold (-75 dBm) when the handover procedure is initiated. As soon as the UE finds a better AP (in this case, AP2), the UE disconnects from the current AP (AP1) and gets attached to the new target AP (AP2) which will update the GNRS with the new network address, ($GUID_{UEx} \rightarrow NA_{AP2}$). A new data flow path is created.
During this handoff process, the in-network packets are stored at the access router. The absence of periodic link probe packets due to disconnection caused by UE mobility between host (UE) and router (AP1) will force the access router to query the GNRS for the current NA of the UE. The stored packets are then forwarded to the new NA and all the in-flight packets reroute as the intermediate routers update their routing table based on the link state advertisements (LSA). This scheme has no packet loss during handover and does not involve the use of tunnels or bearers. Finally, it can support heterogeneous handover between multiple networks (e.g., WiFi and LTE) without any additional configuration.

**Push vs Pull GNRS Service:** The above specified rebinding scheme uses a PULL based approach to query the GNRS for the updated NA if there are no periodic link probes between router and a host. This introduces additional latency, which in turn depends on the frequency of link probe packets. In contrast, a PUSH based scheme can also be introduced where the GNRS will send out updates to the previous NA about the current attachment of GUID, and this can be achieved by comparing current vs previous NA logs.
3.3.2 Soft Handoff with Multihoming

During soft handoff (make-before-break) as shown in Fig. 3.4, we use the multihoming feature to provide seamless mobility support without packet loss. As an example, a timing diagram for Wi-Fi is shown in Fig. 3.5 where the UE is initially attached to AP1 and a new entry is inserted into the GNRS with \((GUID_{UEx} \rightarrow NA_{AP1})\) binding as discussed earlier. The Wi-Fi interface at the UE continuously monitors the RSSI every 100ms and reports averaged RSSI values to the L3 layer every 1s. This averaged RSSI computation enables appropriate handover decisions. If the reported RSSI is less than a certain threshold (-80 dBm in our implementation), the second interface, either LTE or WiFi if available, is activated and gets attached to the network. Once the attachment process is complete a new entry with \((GUID_{UEx} \rightarrow NA_{AP2})\) is added to the GNRS. Thus, the UE is multihomed with two available data paths as shown in Fig. 3.5 and is capable of receiving packets on both the interfaces, simultaneously. We have incorporated Algorithm 6 in our host software design which allows the UE to access any of the available networks.

The multihoming feature is achieved using a variable length header specifying multiple network addresses for packets destined to a UE from the server, thereby allowing intermediate routers to select a suitable bifurcation point \([29]\) for traffic splitting. Therefore, packets can traverse different paths towards LTE/WiFi interfaces based on metrics such as ETT (expected transmission time) and achievable data rate. Uplink
Figure 3.5: Timing diagram for soft handoff

Algorithm 6: Soft handoff for 2 Interfaces

1. **Interface 1**: Inf 1,  **Interface 2**: Inf 2
2. **Parameters**: RSSI, Throughput
3. Periodically monitor parameters on available interfaces
4. **if** parameter < Threshold **then**
5. Begin scanning on Inf 2 for new AP/BS
6. **if** AP/BS exists **then**
7. Associate Inf 2 with the best AP/BS
8. Disassociate Inf 1 (**if** poor channel quality)
9. **else**
10. Continue probing on Inf 2;

traffic uses the best available interfaces. Use of multiple interfaces along with the in-network rerouting support, eliminates the scenarios with ping-pong effects and thus provides a robust connection to at least one of the available interfaces.

### 3.4 Performance Evaluation

We evaluated the following scenarios for seamless handover using *MobilityFirst* name based network architecture: (a) hard handoff with rebinding for a homogeneous WiFi network, (b) soft handoff with multihoming for homogeneous as well as heterogeneous interoperable WiFi and LTE networks, (c) impact of router memory on handover schemes, (d) comparison with IP based handover schemes and (e) Large scale dense HetNet simulation. The evaluation topology for (a)-(d) consists of two BSes and a UE
with mobility induced by changing the attenuation between the pair of BS. The BS can be either Wi-Fi APs or a Wi-Fi AP with an LTE eNB.

### 3.4.1 ORBIT/COSMOS testbed experimentation

In this subsection, we discuss our handover evaluation results considering ORBIT/COSMOS \[45, 83\] testbed.

**Rebinding for Hard Handoff**

Fig. 3.6 shows RTT and throughput plots during hard handoff considering a homogeneous Wi-Fi network obtained using mfping and mperf (modified versions of ping and iperf for MF respectively). As observed in the RTT plot, the handover happens at time $t = 20$s (packet sequence number 20 as the ping inter-packet interval is set to 1s). A disconnection period of 3s is observed because the L2 handoff itself takes around 2.92s as shown in Fig. 3.1 leading to 3 packets being stored at the access routers which need rebinding. After obtaining the updated network address of UE from GNRS, the stored packets are rebound with the new address, and are delivered to UE after a period of 3.21s for pull based GNRS service as against 3s for push based service. The 200ms additional delay for pull based GNRS is mainly due to routing and timers as all the routers are not synchronized at the start of the experiment. For throughput evaluation
mfperf is run over an interval of 80s and the handover happens at time, \( t = 20 \) s. A throughput value of 0 is observed for 2s right after handover and the peak obtained after the handover is mainly due to the packet rebinding. The average throughput before and after handover is observed to be 17.28 Mbps and 14.23 Mbps respectively.

**Multihoming for Soft Handoff**

*Homogeneous WiFi Network* - Considering multihoming based handover with 2 Wi-Fi interfaces, Fig. 3.7 shows the RTT during handover. The decrease in channel quality with the initially attached AP triggers handover with the next best available AP obtained from the scan using multiple interfaces present at the UE. In our controlled experimental setup, this handover happens at packet sequence number 20 where RSSI of `wlan0` is less than RSSI of `wlan1`, and we observe a drop in the RTT after handover as the packets are rerouted to the new interface. The peaks observed after 20s are mainly due to the multihoming as the packets are delivered via the old interface with lower channel quality leading to a rise in the RTT. The intermediate router is capable of deciding which path to select based on the data rate and hence we observe fewer peaks with high RTT values.

*Heterogeneous WiFi and LTE Network* - In this experiment we evaluated RTT for handover between heterogeneous Wi-Fi and LTE network. Initially, the UE is attached

![Figure 3.7: Homogeneous soft handoff with multihoming in WiFi](image)
to only the Wi-Fi network and when the RSSI of Wi-Fi is less than the set threshold (-80dBm), the LTE interface switches from idle to active mode. The UE then attaches to the LTE network’s eNB. We ran two different scenarios during this experiment. In the first scenario, upon switching to the LTE network, the UE is disconnected from the Wi-Fi AP. In the second scenario, upon attaching to the LTE network, the Wi-Fi AP is remained in the attached state to the UE, enabling it to receive and send from both the interfaces. Fig. 3.8 compares the RTT performance for both the scenarios. It is observed that after handover (at packet sequence number 20), the RTT of the second scenario is significantly lower than the first case of single interface use. Therefore, harnessing multiple interfaces during handover can provide seamless connectivity without packet loss.

**Impact of Router Memory on Handover**

The packet loss performance during the handover is evaluated with respect to router queue size and inter-packet transmission delay (IPTD) as shown in Fig. 3.9. For an IPTD of 1s, there is no packet loss irrespective of the available router memory. As the packet transmission rate is increased by lowering the IPTD to 0.01s, 45 packets are lost. The loss is because there is a handover delay which needs to accumulate packets for store-and-forward feature used in this work which fails if there is an insufficient router queue size. Moreover, the network congestion introduced due to higher arrival
rate than the packet processing capability of the router causes packet loss. Therefore, it is essential that for a lossless handover, routers should have sufficiently large memory considering the number of users attached to the network.

**Comparison with Existing Handover Schemes**

We also evaluated packet drops in the IP based handover by using ping with different inter-arrival rate as shown in Table 3.1. For a lower ping interval of 0.1s, 41 packets are lost as compared to 4 packets for 1s inter arrival which aligns with our hard handoff experiment. In addition to the packet loss, applications also have to reset their TCP connection every time a handoff is performed, caused either due to timeout or allocation of new IP address. Further, it is noted that using storage aware routing with sufficient router memory, there are no packet losses using our proposed handover techniques.

<table>
<thead>
<tr>
<th>Packet Inter-Arrival (s)</th>
<th>0.1</th>
<th>0.3</th>
<th>0.5</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packet drop</td>
<td>41</td>
<td>13</td>
<td>7</td>
<td>4</td>
</tr>
</tbody>
</table>

### 3.4.2 City Scale Simulation

To evaluate the performance of multihoming based handover in a realistic deployment scenario comprising of macro and small cells, we developed a discrete event based simulator consisting of dense heterogeneous network (HetNet) deployment with real user
mobility traces obtained from CRAWDAD [84]. The user mobility trace consists of X-Y coordinates sampled every 30s obtained through a measurement campaign (duration of about 2hrs), we inserted location updates of 1s to the original dataset for more accurate handover performance evaluation. The system simulation parameters are listed in Table 3.2. The macro cell has a coverage of 500m as against the small cell having 50m coverage, providing maximum throughput of 75Mbps and 18Mbps with 20MHz and 5MHz bandwidth respectively. The macro cell is comprised of 3 sectors with small cells randomly deployed within macro radius with inter-cell distance of 75m.

Network Deployment: A flat core architecture as shown in Fig. 3.10 with no hierarchy to reach the Internet for macro and small cells is considered. The routers are

Table 3.2: System Simulation Parameters

<table>
<thead>
<tr>
<th>Simulation Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deployment scenario</td>
<td>Single Macro cell with uniformly deployed small cells inside it</td>
</tr>
<tr>
<td>Macro cell radius</td>
<td>500m</td>
</tr>
<tr>
<td>Small cell radius</td>
<td>50m</td>
</tr>
<tr>
<td>Path Loss Model</td>
<td>LogDistancePropagationLossModel</td>
</tr>
<tr>
<td>Inter-cell Distance (small)</td>
<td>75m</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>Macro: 5MHz, Small: 20 MHz</td>
</tr>
<tr>
<td>Tx Power</td>
<td>Macro: 46 dBm, Small: 30 dBm</td>
</tr>
<tr>
<td>Carrier Frequency</td>
<td>2 GHz</td>
</tr>
<tr>
<td>Modulation</td>
<td>64 QAM, No MIMO</td>
</tr>
<tr>
<td>Number of Small Cells</td>
<td>60 per Macro</td>
</tr>
<tr>
<td>Backhaul link BW</td>
<td>10Gpbs</td>
</tr>
</tbody>
</table>
capable of providing both radio as well as backhaul support, and have sufficient memory for packet storage and rebinding. An average latency of 65ms [85] for the handover between macro cell and small cell, and 300 ms [86] for inter-small cell is considered. Owing to the MF architecture, the routers have in-network packet storage and per-interface scheduling capabilities to inherently support multihoming. The named-based multihoming capability is simulated, and the performance with respect to throughput and RTT over user mobility pattern is evaluated against various schemes.

**Access Schemes**: (user association)

- **Macro only**: UE always uses macro cell during entire mobility path.

- **Best available interface**: UE selects the best interface based on throughput between macro and small cells.

- **Multihoming with rebinding**: UE uses both the interfaces if available; packets lost during the handover are rebound to a new NA.

**Scheduling Schemes**: (between UE and intermediate router)

- **Macro only**: only macro cells are used.

- **Round robin for both UL and DL**: the intermediate router and UE device uses the available interfaces alternatively for downlink and uplink transmission respectively.

- **Round robin for DL (UL) & best interface for UL (DL)**: UE uses best available interface based on throughput for UL transmission and for DL, the intermediate router schedules packet alternatively on available interfaces (vice versa).

- **Best available interface always**: UE/router selects the best available interface based on achievable throughput for both DL and UL paths.

The key network parameters such as throughput, RTT and impact of load at the intermediate router on handover performance for a dense heterogeneous network are evaluated. Cells are assumed to occupy non overlapping channel and the cell deployment layout is shown in Fig. 3.11 along with user mobility trace.
The maximum achievable throughput is computed against various UE access schemes. Fig. 3.12 shows the plot of throughput against various UE access schemes with respect to time sampled every 60s for 10 random trials (7200s x 10). The multihoming scheme provides higher throughput (e.g., at 10-20 time ticks) when the user is at the optimal location to receive the multiplexing gain of both macro as well as small cells. At other locations, the gain varies but still outperforms the best interface scheme. The throughput is mainly limited by the small cell availability due to random deployment within macro cell. The large macro cell always provide better connectivity and seamless mobility to UE and significantly help in throughput performance in the absence of small cells.
Throughput Evaluation

Throughput Gain and Packet Loss: The use of best interface and multihoming with re-binding provide an average throughput of 20.9Mbps and 23.5Mbps respectively (Fig. 3.13 right). Multihoming provides a gain of 12.5% with respect to best interface during the entire simulation run. For a single experimental run (7200s), considering stationary UE (10-20 time ticks), multihoming has a gain of 16% against best interface scheme. Hence, use of multiple interfaces for stationary and mobile UE cases improves user perceived QoS, but the current IP architecture lacks this support as it’s achieved mainly through MPTCP [33] without in-network support needed for seamless handoff. For a full buffer simulation, we also observed around 12.4k packets (size 1000 Bytes) being buffered at the respective base stations during entire UE trajectory which includes multiple handoff between small and macro cells. With the help of storage capable routers and GNRS, these packets are rebound to the updated UE location avoiding end to end retransmission as in the case of TCP. Use of multiple interfaces with in-network support provides seamless mobility with increased user throughput.

Proportional vs Equal split of flows: Flows at intermediate router can be split proportionally based on achievable throughput across individual interfaces. The bifurcation router will request for the channel quality from each of the access routers and based on the obtainable data rate an optimization problem as given in Eq. 3.1 can be formulated where the objective is to maximize allocation on individual interfaces ($f_i$) subject to
flow rate ($F_r$) and the bottlenecked throughput ($R_j$). $N$ is the total number of interfaces available on a UE device, $w_i$ represents weight for interface $i$ and $j$ represents links present along path (interface) $P_i$.

$$\max \sum_{i=1}^{N} w_i \cdot f_i \quad \text{s.t.} \quad \sum_{i=1}^{N} f_i \leq F_r$$

$$0 \leq f_i \leq \min(R_j) \quad \forall i \in N, \forall j \in P_i$$

(3.1)

The above optimization problem solves for the optimum rate allocation along individual interfaces. As can be seen from Fig. 3.14, the proportional scheme always outperforms equal share for a flow (sending) rate of 30 Mbps. An average gain of 44% is observed with $w_i = 1$ considering 10 experimental runs sampled every 60s. The gain is especially higher when user is very close to a small cell and obtains higher throughput than with a macro cell (90-95 time ticks). Equal-share does not prioritize between the interfaces and allocates same rate between them, under utilizing the available bandwidth. Fig. 3.15 plots the multihomed user throughput for various flowrates. A higher throughput gain is observed in the case of lower rate application in proportional scheme as compared to equal share scheme. The gain is not that significant for higher rate applications and the proportional fair scheme converges to the equal share scheme. This is mainly because for higher flow rate, we have access being bottlenecked limiting higher user throughput (the core backhaul links have 10Gbps capacity).
RTT Evaluation

To evaluate per packet round trip time from a UE to a remote server, we simulated the scheduling feature at UE and also at the intermediate router to account for multiple interfaces. Fig. 3.16 shows the plot of average RTT per packet (size 1000 bytes) along the entire UE trajectory sampled every 60s as before. Macro cell provides an average end to end RTT of 34.4ms, the majority of which is contributed by UL channel grant access (17ms) [87]. Use of the best interface for both UL and DL improves the performance of packet reception and also allows for seamless handoff between various RATs.

Fig. 3.13 (left) shows the average RTT for different scheduling schemes. We observe that round robin for DL and best interface for UL provides comparable performance
with respect to best interface for both UL/DL which has an additional overhead of periodically querying the access router for the best interfaces.

**Load at Intermediate Router**

The bifurcation router helps in scheduling of packets between multiple available interfaces. Any load present at the router affects the overall performance as the router has to perform scheduling decisions for multiple users. We evaluated the impact of load on intermediate router with respect to RTT considering various scheduling schemes as described earlier. We considered \( \sim 200 \) users under macro/small cell representing the highly loaded case (load 1) and no load is represented as load 0 serving a single user. We observed that the round robin scheduling performs poorly under heavily loaded scenario as the total end-to-end latency can be as high as 1.6s (shown in Fig. 3.17), which impacts live real-time application. Best DL/RR UL and RR DL/best UL performs fairly the same under all load conditions. Best DL/best UL scheme has a better performance as compared with the above schemes having RTT of around 700ms for heavily loaded scenario (roughly half of RR scheme). Therefore, based on the load at intermediate router appropriate scheduling schemes should be selected as for the lightly loaded (load 0 to load 0.4) case all the schemes perform similarly.

![Figure 3.17: Impact of load at intermediate router on RTT](image)

Figure 3.17: Impact of load at intermediate router on RTT
3.5 Related Work

The increased network density and higher user mobility is expected to cause frequent handovers between various access networks, and therefore will make UE (user equipment) attachment and packet forwarding complex due to the hierarchical gateway based architecture of existing mobile core networks based on the 3GPP specifications. The prime contributor of handoff latency in current implementations is L2 switching, motivating the authors of [75, 88] to propose mechanisms using single and dual 802.11 radio cards on UEs. These schemes attempt to reduce the channel probing latency with multiple radio cards but involve changes to the kernel and limited support for heterogeneous networks. Various other schemes have also been proposed to optimize WiFi L2 handoff [89–92]. However, they face similar issues in deployment, channel selection/grouping and Min/MaxChannelTimes threshold settings. Efforts have also been made for seamless heterogeneous handover and scheduling between WiFi and LTE. Architectures pertaining to core assisted (S2a/S2b interfaces) and RAN assisted (LWA/LWIP) LTE-WiFi integration have been widely discussed in 3GPP standards meetings [93].

### Table 3.3: Comparison of different Mobility Management Schemes

<table>
<thead>
<tr>
<th></th>
<th>Mobile IP</th>
<th>Proxy MIP</th>
<th>SDN</th>
<th>Name Based</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Architecture</strong></td>
<td>Partially</td>
<td>Partially</td>
<td>Centralized</td>
<td>Fully</td>
</tr>
<tr>
<td><strong>Management</strong></td>
<td>Distributed</td>
<td>Distributed</td>
<td>Network/Controller</td>
<td>Distributed</td>
</tr>
<tr>
<td><strong>Addressing</strong></td>
<td>Host</td>
<td>Network</td>
<td>Fixed IPs/IDs</td>
<td>Network</td>
</tr>
<tr>
<td><strong>Mobility Anchor</strong></td>
<td>HA and CoA [1]</td>
<td>HA and CoA</td>
<td>Single/Multiple</td>
<td>Fixed IDs</td>
</tr>
<tr>
<td><strong>Data Transport</strong></td>
<td>Multiple</td>
<td>Multiple</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td><strong>Route</strong></td>
<td>IP Tunnel</td>
<td>IP Tunnel</td>
<td>No Tunnels</td>
<td>No Tunnels</td>
</tr>
<tr>
<td><strong>Optimization</strong></td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Signalling</strong></td>
<td>High</td>
<td>High</td>
<td>Med</td>
<td>Low</td>
</tr>
</tbody>
</table>

The rise in network heterogeneity requires additional solutions such as distributed

---

1Home Address and Care of Address
mobility management (DMM) and centralized software defined network (SDN). In [30],
the author’s emphasised on elimination of single mobility anchor and categorized IP
level DMM schemes into 3 categories – Proxy Mobile IPv6 (PMIPv6), SDN and routing
considering BGP. In PMIPv6 based DMM [30,94], the centralized core architecture of
PMIPv6 having LMA (local mobility anchor) and MAGs (mobility access gateways)
are eliminated using DMM-GW to provide direct connectivity to the Internet. This
approach introduces multiple distributed gateways which in turn increases switching
latency and tunneling overhead during UE mobility. Several SDN based DMM schemes
have also been proposed with and without IP tunnels [31,32]. For a large scale network
deployment, these solutions face scalability and controller management issues. Further,
services such as multicast, multihoming for the current TCP/IP based architecture
work as overlays incurring additional overhead and limited network visibility [95].

The architectures described above face challenges in: (a) configuring gateways, (b)
setting-up tunnels between various RATs either on per UE basis or operator specific
policies, and (c) lack of network support for efficient handover decisions. Table 3.3
provides a qualitative comparison between different mobility management schemes cur-
rently under consideration with the named-object based solution considered here. The
architecture proposed in this work aligns well with routing based DMM for seamless
handover considering a flat network having name and address separation, and allowing
network components to assist in seamless handoff through rebinding and multihoming.

3.6 Summary

This chapter evaluated two handover schemes based on an implementation of a named-
object based flat mobile core network on the ORBIT radio grid testbed. The results
show that rebinding and multihoming capabilities in the architecture can be used to
provide seamless mobility. Handover performance was evaluated with respect to RTT
(round trip time), throughput and packet loss. We demonstrate that using multiple
interfaces during handover can realize seamless connectivity with low RTT as compared
with a single interface where the RTT is bounded by L2 handoff latency. Furthermore,
dynamic network rebinding for an in-transit packet during handover lowers the average
RTT of the system as it avoids end-to-end retransmissions. The proposed approach thus solves the mobility problem in dense HetNets without resetting of end-to-end sessions and tunnels during the handover. Also, unlike IP based handovers, our schemes can achieve seamless connectivity with zero packet loss given sufficient router memory. Soft handoff with multihoming schemes were also evaluated in a dense HetNet scenario with macro and small cells along with real mobility traces. The results show that the above schemes provide an average throughput gain of 12.5% for mobile users and 16% for static users with zero packet loss when compared with a best interface policy. The proportional flow splitting between multiple interfaces achieves a gain of around 44% relative to equal share for lower bitrate applications.
Chapter 4

Cross-layer Assisted Pull based Transport for mmWave Networks

4.1 Introduction

In this chapter, we address the design of an end-to-end (E2E) transport protocol capable of harnessing the fast radio link layers associated with 5G mmWave network. We overcome the problem of TCP probing and wastage of bandwidth over intermittent mmWave channel by proposing mmCPTP, a novel cross-layer E2E transport protocol which uses concepts from information centric networks (ICN) to fully utilize the bandwidth of the mmWave access link. We use a pull-based mechanism on the fluctuating radio link to opportunistically pull the packets based on the resource availability at the base station (BS) or access point (AP). The goal is to ensure that data is always available to send to the mobile client when the radio PHY speed is high. To further reduce control latency associated with the transport protocol, we introduce a transport proxy (cache) closer to the user. The PHY aware cross-layer pull mechanism is only used between the BS and proxy, and if the content is not present at the proxy, traditional protocols which use AIMD mechanisms such as TCP can still be used to retrieve the content from the file-server to the proxy. Our solution eliminates frequent probing as in TCP and thus makes it possible for data transfer speeds to match available 5G link bandwidth as resources become available.

The major contributions of this work are as follows:

- We study the behavior of TCP over mmWave channel under a lab environment, identify its problems and list the key requirements of a transport protocol to support such links.
We propose **mmCPTP**, a novel cross-layer pull-based transport solution to work with mmWave channel, whilst achieving high throughput efficiency and resiliency from outages.

The proposed scheme is implemented in 5G NR (New Radio) stack and is validated using NS3 simulation as well as on ORBIT/COSMOS and COSMIC testbeds against various TCP versions over intermittent mmWave channel.

## 4.2 TCP Performance over mmWave

Short wavelength mmWave bands are subjected to attenuation and blockages resulting in higher and more variable pathloss than in sub-6 Ghz bands. Frequent switching between LoS and nLoS links in the mmWave causes the buffers at the intermediate router to overflow due to the high difference in the data rate triggering TCP fast retransmit mechanisms and timeouts required for adjusting to the new available bandwidth. Upon re-transmission timeout (RTO, usually set as 1s), the low slow start threshold causes TCP to take longer time to achieve optimum capacity due to high round-trip-time (RTT). On each successful handover, TCP probing takes a considerable amount of time and sometimes it is not even possible to reach the peak mmWave capacity. Here, we present some simple evaluation results, in order to get an idea of potential throughput gains that could be achieved with a redesigned protocol. We start with an off-the-shelf mmWave transceiver (InterDigital EdgeLink - 60 GHz, 802.11ad) on the COSMOS sandbox, and then move to an emulated mobile user scenario implemented on the ORBIT radio grid testbed.

The first set of lab measurements was carried out with the EdgeLink mmWave transceivers as shown in Fig. 4.1. The Tx/Rx devices were placed 25ft apart and the link was subjected to physical blockage at a distance 15ft from the Tx for over 5s interval. From Table 4.1, we see that with a slight shift in Tx orientation, the throughput drops as much as 10x when compared to the LoS scenario having a max throughput of 811Mbps. Also, the mmWave link is highly susceptible to blockages as the throughput drops to zero during the blockage interval (see Fig. 4.2).
Experimentation Setup

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tx-Rx distance</td>
<td>25 ft</td>
</tr>
<tr>
<td>Blockage interval</td>
<td>5 s</td>
</tr>
<tr>
<td>Ping test (RTT)</td>
<td>0.295 ms</td>
</tr>
</tbody>
</table>

Throughput (Mbps, 100s interval)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LoS</td>
<td>811.00</td>
</tr>
<tr>
<td>±10° Tx shift</td>
<td>451.00</td>
</tr>
<tr>
<td>±30° Tx shift</td>
<td>8.52</td>
</tr>
<tr>
<td>During blockage</td>
<td>Zero</td>
</tr>
</tbody>
</table>

Table 4.1: InterDigital EdgeLink TCP performance on COSMOS

---

Figure 4.1: InterDigital mmWave Tx/Rx experimentation setup

Figure 4.2: TCP Cubic under human blockage

We also evaluated the recovery time from the outages under different RTT settings using a 3 node experimental setup on ORBIT testbed (shown in Fig. 4.3). We measured the time required by the rate adaptation algorithm to switch from the lowest setting of 20Mbps to the peak value of 1Gbps and as shown in Fig. 4.4 the TCP recovery time from 20Mbps to 1Gbps can be as high as 34s for RTT 100ms (mimicking distant server location). As observed, higher the RTT, the more time it takes to reach optimal capacity. This is mainly due to the presence of a long feedback loop in TCP potentially resulting in the starvation of data in the critical radio link.

Figure 4.3: ORBIT/COSMOS topology setup
As the next step, we considered the performance of TCP (Cubic) for a mobile user scenario via emulation of the available handoff model with traffic shaping of an Ethernet link in the testbed setup. The emulation switches between a maximum of 1Gbps and a minimum of 20Mbps (RTT set to 50ms), based on the mobile device mobility traces obtained from [15]. The standard Linux tool `tc` was used to limit the bandwidth. As observed from Fig. 4.5 after each handoff event, TCP reacts slowly taking several seconds to reach possible peak mmWave speeds. This is clearly observed at time 15s, 50s, and 330s in the figure. Running the experiment over the entire 450s mobility trace, the achievable data transfer with TCP is 16 GB, while the theoretical maximum is 29.14 GB. This means that the use of TCP for transport incurs a 1.82x penalty over best-case transport protocol considering all PHY layer channel limitations, motivating our work on a new transport layer design that approaches the upper limit of achievable performance. From the above experiments, we can infer that there is a serious performance penalty associated with the use of TCP over a high bandwidth.
mmWave link. Our proposed transport scheme uses a novel pull mechanism to receive packets with the main objective of efficient utilization of high bandwidth-delay product (BDP) links when available.

4.3 Design Objectives

In this section, we present below the set of design objectives for a transport protocol comprising a heterogeneous network, having mmWave and sub 6 GHz links, with the following set of assumptions- user stationary or mobile, long-lived traffic flow, and sufficient resource availability at the BS and core.

Support mmWave intermittency and capacity: Considering TCP, the probing phase takes a considerable amount of time and bandwidth to reach optimal capacity. This phase often underutilizes the network since there are frequent fluctuations between various available paths and probing for all these short term paths consumes valuable time and resources. The desired protocol should reach link layer capacity as early as possible.

Support seamless mobility: Frequent handoffs occurring between sub 6 GHz and mmWave paths causes TCP to reset the connection with each handover (due to the change in the IP address assignment if managed by different entities). This, in-turn causes timeout and retransmission of packets from the end host due to the packets being dropped as they cannot be forwarded. The protocol should provide seamless mobility among various available networks and moreover, it should also be capable of supporting in-network rebinding/re-routing of data packets based on the current user location.

Congestion control and packet loss: The intermittent wireless link having high bit-error-rate (BER) and packet loss, degrades the performance of transport protocol (in TCP) as it conflates loss due to link failure as congestion invoking rate limiting mechanisms to minimize congestion. Also, providing E2E Gbps throughput over high RTT is challenging in mmWave due to its propagation characteristics and intermittency. A better mechanism for congestion control and packet loss is needed to achieve high data transfer speeds over mmWave access channels.
4.4 mmCPTP - Transport Protocol

In this section, we describe briefly the design concepts and design details behind our proposed mmCPTP protocol.

4.4.1 Design Concepts

Based on the above objectives, the key design principles behind mmCPTP are described in this section. The high level architecture of mmCPTP is shown in Fig. 4.6. The link between BS and proxy uses the concept of receivers pulling the data based on an interest profile or globally unique identifier depending on the specific ICN technique used.

Avoid probing: To reduce the effect of fluctuations in mmWave band, we introduce a novel pull mechanism to retrieve packets from the file-server present closer to the user. There is no probing involved as the packets are pulled based on the feedback from the lower layer (RLC/MAC) buffer information. This particularly helps in mmWave scenarios since there are frequent outages and probing/reacting to all these temporary paths causes the channel to remain underutilized right after signal outages. Our solution sits on top of the last hop bottleneck link thus allowing us to use pull mechanism.

Split connection: If the content (file-server) has significant RTT to the user, to provide low latency and better throughput efficiency, we break the link into wired and wireless parts with the help of a proxy. Such an approach can perform better in a highly dynamic scenario where the intermediate proxy node acts like a temporary
cache/anchor to store packets received from the distant file-server and subsequently transfer temporarily stored data to the mobile user. In doing so, we fully utilize the potentially high BDP backhaul network by preventing the sender from limiting the rate due to frequent interruptions, thus, avoiding E2E retransmissions and minimizing latency.

**Separate packet loss and congestion control:** To overcome the issues faced by TCP in handling packet loss events and also to prevent any adverse impact because of this on the higher layers, we use a separate mechanism (only between proxy and user) for packet loss recovery and congestion control. The loss over the wireless access link is taken care by the lower layers of the stack such as link/MAC so that the transport layer only needs to be concerned with flow and congestion control.

### 4.4.2 Design Details

In this subsection, we present further design details of our proposed mmCPTP transport protocol.

**Flow and congestion control:** 1) *PULL based congestion and flow control:* The pull based data transfer is subjected to pull buffer occupancy on a per-user/UE basis at the transport layer of BS or AP. The data packets are pulled from the file-server/proxy by the BS depending on specific pull mechanisms as described next and the received packets are further sent to the lower RLC/MAC layer based on their buffer status report. The reason we use pull is to initiate traffic on the bottleneck link where we know how much to send every time or alternatively how much to pull every time. The described pull process eliminates probing and allows for fast fluctuations between LoS and nLoS paths, thus keeping the radio channel better utilized than TCP (please refer to section 4.5 and 4.6 for more details).

2) *PULL mechanisms:* Multiple pull techniques for data retrieval can be used at the BS (see Fig. 4.7), for example; a) *Periodic pull:* packets are periodically pulled based on current pull buffer occupancy. The period $T$ can operate at a much larger timescale and ideally, the buffer size should be the product of max-last-hop BW * periodicity (in bytes or packets); b) *Threshold based:* pull buffer occupancy is continuously monitored.
Figure 4.7: Pull mechanisms at the base station, a) Periodic, b) Threshold based

and if it is less than a certain predetermined threshold, packets are pulled to fill the buffer. Here, the size can be very low but at the cost of an increased number of pulls.

3) AIMD based flow control: TCP which extensively relies on AIMD, is used between the file-server and proxy provided the content is not present at the proxy, for a guaranteed fair share of bottleneck link capacity. The main rationale behind this is, TCP performs fairly well in the core and the throughput reduction issue only exists in the wireless medium [14,101].

Error recovery: Reliable delivery of data packets is always ensured between the components of our transport protocol. TCP takes care of reliably delivering packets between the file-server and proxy, whereas between the BS and proxy, SACKs are used to selectively acknowledge proxy in the event of any packet loss. Furthermore, NACK (Negative Acknowledgement) is used to indicate handover along with the id of the last packet sequence successfully delivered from the source BS to UE, and ACKs (Acknowledgement) between UE and BS are used to indicate any loss in the wireless medium. In addition to this, lower layer error recovery techniques such as HARQ, also help in reliable packet delivery.

Cross-layer plug-in: Our scheme is designed and implemented as a cross-layer BS plug-in with no major changes at the sender and mobile client stacks. This new functionality is mostly at the BS as it needs to have access to the L2 information. Since our protocol involves a cross-layer design (involving L2 and L4), it can work with different L3 options (IP + 3GPP or NDN [48] or MF [49]) allowing for incremental as well
as backward compatible deployment. We also provide an abstraction layer for unified access over WiFi, LTE, NR, or any future L2 protocol.

### 4.5 Protocol Overview

To illustrate the operation of mmCPTP over IP network, we consider a UE (X), BS (B), proxy (P), and file-server (S) having respective IP addresses and port numbers as shown in Fig. 4.8. At first, the client UE resolves Y.URL (address of content Y) to a nearby proxy using the resolution server. Next, the UE requests for a content Y from the proxy. If the content is present locally at the proxy, it is retrieved using pull, or else an `HTTP_REQUEST()` is made to the file-server by the proxy requesting for content Y. A TCP session is setup between the file-server and proxy to receive data packets. The proxy immediately notifies UE for any failure in `HTTP_REQUEST()`, and if required the UE can retry requesting the same content again. As the packets start arriving, it is cached locally at the proxy and a packet is sent to the UE (via BS) initiating data transfer. The BS on receiving this packet allocates a per-UE pull buffer (if it is the first time and no buffer has been initialized for that UE before) and requests data packets from the proxy using pull. On receiving the pull, a counter is initialized at the proxy and it starts sending requested data packets to the UE. The pull packet, issued either periodically (every T ms) or threshold based, depends on the packets being successfully delivered to the UE during the previous pull interval. The BS receives these data packets and selectively acknowledges (SACK) the proxy if there is any packet loss. Otherwise, based on the MAC layer scheduler, the BS will deliver these packets to the UE. The UE acknowledges (ACK) the received data packets and the BS will discard these ACKs, perform re-transmissions if required, and clear the buffer based on the received ACKs. As the proxy receives the request for N packets, it keeps sending N data packets (N is considered as window size). The proxy will decrement the pull counter based on the number of data packets being sent and will increment with every new pull request. Also, with every new pull request the proxy can clear the corresponding number of bytes from its cache.

In the event of handovers between the BSs, proxy helps in performing the switch
and ensures the reliable transfer of data. The source BS issues NACK to the proxy, indicating unsuccessful delivery of packets (and also handover) along with their IDs. Based on this feedback message the proxy will reroute the packets from the previously undelivered sequence to a new user location (target BS). Since the proxy is located at an infrastructure end, it expects less failure [50]. In any case, the states are always stored and if required the proxy will re-initiate a new session with the file-server and subsequently transfers data to the UE.

4.5.1 ICN based future internet architecture

Future Internet Architectures (FIAs) [48, 49] based on named-object techniques have the potential to enable seamless mobility by enabling storage aware efficient routing [79] involving delay tolerant networking (DTN) mechanisms. But the capabilities of these architectures over mmWave are still new and unexplored. Our transport protocol for beyond 5G networks can also be implemented on top of these FIAs and as an example we describe our solution considering MobilityFirst (MF) [49] which has a clean separation of name and network address (NA) space, resolved dynamically with the help of a logically centralized but a physically distributed global name resolution service (GNRS). The global unique identifier (GUID) is an identifier assigned not just to UE devices but also
to groups, content, or context and remains fixed throughout the lifetime of a device.

As illustrated in Fig. 4.9 to access content with GUID \( Y \), UE \( X \) associates to a BS and issues \texttt{GET()} with SRC GUID \( X \) and DST GUID \( Y \). The server on receiving this \texttt{GET()} message, starts sending data to proxy (intermediate MF router having transport proxy support [100]) using an AIMD with SRC GUID: \( Y \), DST GUID: \( X \), SRC NA: server location and DST NA: proxy location. As the file (in chunks) starts arriving from the server, the proxy tries to deliver these chunks to the UE. Initially, the proxy issues a CSYN packet, informing BS (NA1) about the content for UE \( X \). On receiving the CSYN, as seen before the BS will allocate a per-UE pull buffer and requests those many data packets from the proxy. On receiving the pull, a counter is initialized at the proxy and it starts sending data packets along with CSYN (chunk-id 1 in the figure). BS will receive these data packets and will only issue CSYN ACK (with bitmap) immediately to the proxy if there is any packet loss. Otherwise, the BS will try delivering these packets to the UE based on the MAC layer scheduler and once the chunk has been successfully delivered to the UE (when BS receives CSYN ACK from UE), it will issue CSYN ACK to the proxy indicating the chunk has been successfully delivered and can clear those many bytes from its cache. The pull counter is updated in a similar fashion as described before for IP based scheme.
4.6 Implementation Details

This section discusses the mmCPTP implementation details considering service layer abstraction, integration with 5G NR stack, and design details involving Click modular software.

**Pull based abstraction layer.** The purpose of this layer is to provide a unified pull based transport access across different technologies ranging from WiFi, LTE, NR, or any future L2 protocol. It consists of a pull buffer of predetermined size and various interfaces to access the sockets required to send/receive packets to/from the lower radio layers and file-server/proxy respectively. This abstraction layer is only present at the BS since it does the pulling based on pull buffer occupancy and lower layer buffer information. As shown in Fig. 4.10 (considering the NR stack), the `Send_Pull()` is used to receive data packets to fill the pull buffer. The `Report_Buffer_Status()` is used to get information on the lower layer buffer irrespective of the radio access technology being used and is called for every MAC scheduling interval. In the case of LTE/NR stack, the RLC buffer is being reported whereas, in the case of WiFi, it is the status of the MAC queue that is being reported periodically to the abstract layer. Based on this buffer report, packets are sent out using `Push_Data_Packets()` from the pull buffer to lower RLC or MAC queues. Later, from these queues, the packets are dequeued to the UEs based on MAC scheduling decisions.

**Integration with 5G NR stack.** We modified the 5G NR stack to support mmCPTP as shown in Fig. 4.10. For our implementation, we leveraged most parts of the code and support from the NS3 LTE stack, since NR will be using the same RRC, PDCP, and RLC stack as in LTE but different PHY/MAC layers. Our transport service sits on top of the PDCP layer and there are various socket interfaces for sending and receiving packets as follows. Sockets `S1-u NetDevice` and `LTE NetDevice` are used to send/receive packets to/from the remote server and lower radio layer respectively. Between UE and packet gateway (PGW) there exists a default GTP/UDP/IP tunneling procedure where information on tunnel endpoint identifier (TEID), radio network temporary identifier (RNTI), and radio bearer identifier (RBID) are required to setup
the tunnel with appropriate QoS identifier. A one-to-one mapping between the UE IP address and these identifiers is present which is then required by our abstract layer to allocate appropriate pull buffer size and access RLC buffer status. An RLC service access provider (SAP) is used by the transport layer to periodically access the RLC buffer status, which is required to push in data packets from the pull buffer to the RLC for further delivery to the user. Also, note that mmCPTP is independent of any MAC layer scheduling decisions.

**Click based protocol modules.** We implement mmCPTP on ORBIT using Click modular software [102]. Click provides an easy to configure modular schema individually for each of the components as shown in the element graph (Fig. 4.11). The *FromDevice* and *ToDevice* elements are configured with their respective Ethernet addresses. At the BS, a trace driven *Scheduler* is implemented which schedules bytes in terms of packets, based on trace reporting interval. The pull buffer periodically pulls the data based on buffer occupancy (*PullLogic*) from the file-server and uses the *Scheduler* element to push data to the UE. The *Reliability* element ensures reliable transfer of data in a hop-by-hop fashion. Finally, the *Encap* and *Decap* elements are respectively configured to encapsulate and decapsulate the packets with IP headers to enable routing.
4.7 Performance Evaluation

We evaluate the performance of the proposed mmCPTP transport scheme in 5G NR and also trace driven emulation over Gbps Ethernet channel using NS3 simulation (v3.33) and ORBIT testbed. The mmWave channel traces were obtained from NYU mmWave simulator [103]. The deployment includes a mobile UE surrounded by buildings modeled as commercial type having concrete walls and height uniformly distributed between 30m and 40m, deployed within 500m [X] * 100m [Y] area having a certain minimum distance between them to avoid overlapping. We consider four mobility models: 1) UE moving horizontally (along X axis, towards origin) with a speed of 15m/s, 2) UE moving vertically (along Y axis) with a speed of 15m/s, 3) Random walk mobility model, where every 10s the UE changes its speed and direction based on a normal distribution with mean 2m/s and variance 0.04m/s and 4) an emulated channel model, where outages are emulated by moving UE far away (nLoS) from the BS for a time uniformly distributed between 0.1s to 2s and vice versa. For models 1 and 3, the BS is deployed at (25m, 120m, 10m), whereas for models 2 and 4, the BS is deployed at (25m, 50m, 10m). The UE originates at (550m, -20m, 1.6m) for mobility models 1 and 2 whereas, for a random walk, the UE chooses a random coordinate within the deployment area. The channel traces for all these mobility models are shown in Fig. 4.12. In the case of NR based evaluations, the underlying channel used is LTE having BuildingsChannelCondition model (see Fig. 4.12(a) and Fig. 4.12(b)) whereas, for Ethernet based emulation, mmWave channel (see Fig. 4.12(c) and Fig. 4.12(d)) of DL bandwidth 1 GHz is considered. More details on the simulation parameters and
4.7.1 Evaluation with 5G NR stack

System setup

In NR/LTE, the evaluations were carried out using topology consisting of a file-server, evolved packet core (EPC), BS, and a UE (RTT~25ms). The mmCPTP module at the BS periodically pulls data packets from the file-server.
Table 4.2: System Simulation Parameters

<table>
<thead>
<tr>
<th>Simulation Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deployment scenario</td>
<td>Single mmWave BS &amp; UE in downtown area</td>
</tr>
<tr>
<td>Path loss model</td>
<td>3GPP Uma Propagation &amp; Hybrid Buildings (Fig. 4.12(b))</td>
</tr>
<tr>
<td>Channel</td>
<td>Buildings Channel</td>
</tr>
<tr>
<td>Channel Condition Model</td>
<td>(3GPP)</td>
</tr>
<tr>
<td>Deployment area</td>
<td>500m x 100m</td>
</tr>
<tr>
<td>Number of buildings</td>
<td>15</td>
</tr>
<tr>
<td>Building dimension</td>
<td>Variable (height: 30 - 40m)</td>
</tr>
<tr>
<td>Carrier frequency</td>
<td>2 GHz (NR/LTE) &amp; 28 GHz (Emulation)</td>
</tr>
<tr>
<td>DL Bandwidth</td>
<td>RBs: 25, 50, 100 (NR/LTE) &amp; 1 GHz (Emulation)</td>
</tr>
<tr>
<td>Tx power</td>
<td>BS: 30 dBm, UE: 30 dBm</td>
</tr>
<tr>
<td>BS height</td>
<td>10m</td>
</tr>
<tr>
<td>UE height</td>
<td>1.6m</td>
</tr>
<tr>
<td>UE mobility model</td>
<td>Constant velocity (15m/s) &amp; outdoor 2-d random walk</td>
</tr>
<tr>
<td>Congestion control</td>
<td>TcpNewReno and Cubic</td>
</tr>
<tr>
<td>Sampling interval</td>
<td>100ms</td>
</tr>
</tbody>
</table>

Single UE

We evaluate the performance of mmCPTP for a single user considering channel traces as shown in Fig. 4.12(a) and Fig. 4.12(b). The simulation is run for 40s and 10s respectively for each of these traces and from Fig. 4.13(a), we see that, for high speed mobile UE (moving along X axis), mmCPTP achieves a max throughput gain 2.5x times as compared to TCP New Reno and 1.3x times as compared to TCP Cubic congestion control mechanism. The gain increases with the increase in resource block size or bandwidth. With the emulated UE outage model, the gain observed is fairly high as compared to the mobile UE model since we are randomly injecting outages, and our protocol is meant to optimize these temporary outages. From Fig. 4.13(b), we see that mmCPTP achieves a gain close to 4x and 2x as compared to New Reno and Cubic respectively.

Furthermore, we also evaluate the throughput of mobile UE over 30 unique trials considering an emulated channel model as given in Fig. 4.13(b). From Table 4.3, for
100 resource blocks, mmCPTP achieves a max throughput gain of 1.93x and 3.88x as compared to TCP Cubic and New Reno respectively. The mean throughput and standard deviation over a large number of trials for mmCPTP transport scheme is observed to be 37.21 Mbps and 6.59 Mbps respectively.

Table 4.3: Statistical throughput analysis of 5G NR UE (100 RB, channel: Fig. 4.12(b))

<table>
<thead>
<tr>
<th>Transport Schemes</th>
<th>Mean</th>
<th>Std.</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP New Reno</td>
<td>9.59 Mbps</td>
<td>6.56 Mbps</td>
</tr>
<tr>
<td>TCP Cubic</td>
<td>19.20 Mbps</td>
<td>5.46 Mbps</td>
</tr>
<tr>
<td>mmCPTP</td>
<td>37.21 Mbps</td>
<td>6.59 Mbps</td>
</tr>
</tbody>
</table>

Multiple UEs and fairness

The impact of having multiple users is studied using the same simulation setup as before. UE 1 is stationary (loc: 0, -20, 1.6) and remains in the LoS region all the time, whereas UE 2 is having temporary outages as shown in Fig. 4.12(b). The eNB MAC scheduler allocates appropriate transport block size (TBS) on a per subframe basis to these users. We evaluate the throughput performance of mmCPTP against New Reno and Cubic for a 10s simulation interval. From Fig. 4.14 we see that mmCPTP performs well among other considered protocols for the mobile user (UE 2), but the performance deteriorates for a stationary user (UE 1). This behavior is expected and it is because of
the Proportional Fair scheduler. The radio resource blocks are allocated proportionally to these UEs and since our protocol is suitably designed for UEs having frequent outages or blockages, it achieves a max throughput gain of 2.8x and 1.5x times as compared to New Reno and Cubic schemes respectively for UE 2 (see Fig. 4.14(b)). We also compute the Jain’s Fairness Index (JFI) \cite{104} to validate the fairness of our protocol over other TCP variants. The JFI is given in equation 4.1, where $x_i$ characterizes throughput proportional to the maximum achievable MAC throughput per user and $n$ represents the total number of users.

$$J(x_1,..x_n) = \frac{(\sum_{i=1}^{n} x_i)^2}{n \cdot \sum_{i=1}^{n} x_i}, x_i = \frac{APP.tpt. of \ UE_i}{MAC.tpt. of \ UE_i}$$ \hspace{1cm} (4.1)

As seen in Table 4.4, mmCPTP provides a better fairness index as compared to other TCP versions, since it operates closely with the MAC scheduler and lower layer buffer status, thus ensuring better resource utilization.

<table>
<thead>
<tr>
<th>Transport Schemes</th>
<th>Resource Blocks (RBs)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>25</td>
</tr>
<tr>
<td>TCP New Reno</td>
<td>0.9123</td>
</tr>
<tr>
<td>TCP Cubic</td>
<td>0.9998</td>
</tr>
<tr>
<td>mmCPTP</td>
<td>0.9999</td>
</tr>
</tbody>
</table>
4.7.2 End-to-end simulation

System setup

In addition to the evaluation of mmCPTP over NR stack, evaluations were also carried out using trace driven emulation implemented with a traffic shaped ethernet link emulating radio link’s channel traces. The system topology is shown in Fig. 4.15 consisting of a file-server, BS, UE, and a proxy between the end hosts. The SINR channel traces were mapped to their respective bit-rate based on the TBS and are injected into the simulator sampled every 100ms. A TCP session between the file-server and proxy is considered in our simulation. The performance comparison of our scheme with respect to TCP and Indirect-TCP (I-TCP) [50] is evaluated against a large content transfer application.

![Simulation setup](Figure 4.15: Simulation setup for trace driven mmWave emulation)

Content delivery over mmWave channel

We evaluate the performance of mmCPTP using the outdoor random walk mobility setting. The SINR corresponding to its trajectory is shown in Fig. 4.12(c) respectively for 110s of simulation duration. From the traces, we see that during the LoS path, the SINR can be as high as 20dB, but in the event of NLoS i.e. when the links get blocked by buildings (between 10-50s interval) the SINR drops to -40dB leading to the low data rate. Fig. 4.16 presents the file transfer size vs. time for various transport schemes, and we observe that mmCPTP is quickly able to reach optimum capacity as compared with I-TCP and TCP. As observed with mmCPTP, a gain of 31% can be obtained against I-TCP considering Cubic. TCP performs poorly both with Cubic and New Reno schemes as only 970MB and 800MB of data are respectively transferred during the entire user trajectory. In the congestion avoidance phase, due to the presence of high RTT, the recovery time for TCP during frequent interruptions is very high, leading to low utilization of the mmWave channel.
Impact of RTT and loss rate on throughput

To evaluate the performance of mmCPTP under different RTT and loss rates, we perform a similar experiment as before considering channel as shown in Fig. 4.12(d). We repeat this experiment for different network parameters by varying RTT (X set as 5ms, 10ms, 15ms) and introducing loss at the last-hop mmWave link (0 or 0.1% loss). From Fig. 4.17, we observe that the throughput is relatively high and remains fairly stable in all the cases considered for mmCPTP, mainly because there is no feedback loop as in the case of TCP or I-TCP and any packet loss is recovered locally. The throughput of
I-TCP drops significantly in the presence of high RTT and packet loss. A gain of 4.5x is observed with our scheme as compared to I-TCP (Cubic), having X set to 15ms and a loss ratio of 0.1%.

**Trade-offs between multiple Pull techniques**

The pull mechanisms as described before were evaluated considering high-speed mobile UE channel trace (as shown in Fig. 4.12(d)). From Fig. 4.18, we see that the ideal file transfer size is around 340MB for the entire user trajectory. We also observe that, with periodic pull, the buffer size has to be sufficiently large as well as pull periodicity low in order to fully utilize the mmWave channel. But, considering threshold-based pull, the buffer occupancy is periodically monitored every 10ms, and having a low pull buffer size of 5000 packets with a buffer size threshold set to 80%, around 340MB of data can be retrieved from the proxy. Ideally, the pull buffer size should be the product of max last hop BW * periodicity (in bytes or packets). In threshold-based, the buffer size can be very small but at the cost of an increased number of pulls, and having sufficient packets at the pull buffer always ensures the channel is fully utilized.

![Figure 4.18: Evaluation of multiple Pull mechanisms](image)

**4.7.3 Real-time ORBIT emulation**

To further validate the performance of mmCPTP on a real system, we implement it in Click software (as described in section 4.6) and compare its performance against various
TCP versions on ORBIT, using the channel as shown in Fig. 1.2 and topology as given in Fig. 4.3. A standard Linux package tool tc is used to limit the bandwidth and introduce any additional delays between the nodes. From Table 4.5, we see that having a low RTT of 0.3ms (default), all the TCP schemes perform fairly similar, and a total of 11.80GB of data is transferred on average along the entire user trajectory. But the performance degrades with the increase in RTT due to the presence of a longer feedback loop. A performance gain of 17.2x and 5.7x is observed with mmCPTP as compared to New Reno and Cubic, respectively with RTT 50.3ms. The proposed mmCPTP is also very close to the theoretical maximum transfer size of 13.27GB. When compared against other existing TCP versions supporting high BDP links such as TCP HighSpeed [105], Yeah [106] (Yet Another HighSpeed TCP) and also BBR [107] (Bottleneck Bandwidth and Round-trip propagation time; model based), mmCPTP still outperforms these schemes in all the considered scenarios.

Table 4.5: File transfer size in GB over different RTTs using channel as shown in Fig. 1.2 on the ORBIT testbed

<table>
<thead>
<tr>
<th>Transport Schemes</th>
<th>Round Trip Time (RTT)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.3ms</td>
</tr>
<tr>
<td>TCP New Reno</td>
<td>11.80</td>
</tr>
<tr>
<td>TCP Cubic</td>
<td>11.80</td>
</tr>
<tr>
<td>TCP HighSpeed</td>
<td>11.80</td>
</tr>
<tr>
<td>TCP Yeah</td>
<td>11.80</td>
</tr>
<tr>
<td>TCP BBR</td>
<td>11.70</td>
</tr>
<tr>
<td>mmCPTP</td>
<td>12.96 (average)</td>
</tr>
<tr>
<td>Max</td>
<td>13.27 (theoretical limit)</td>
</tr>
</tbody>
</table>

These results demonstrate that the proposed mechanisms such as pull, cross-layer feedback support, separation of packet loss recovery, and congestion control from the transport layer, help highly intermittent mmWave links to reach optimal bottleneck capacity as early as possible and improve goodput.

4.7.4 Real-time emulation on COSMIC testbed

We also validate the performance of TCP and our proposed mmCPTP protocol on the COSMIC [96] tested. The COSMIC platform enables the use of programmable wireless, optical, and edge-cloud network infrastructure for international collaborative
Figure 4.19: COSMIC experimentation setup for trace driven emulation

experiments. Currently, we use ORBIT (USA) and CONNECT (Ireland) testbeds to run our experiments demonstrating a high BDP transatlantic link. The RTT of this link is found to be 112.42 ms and our topology consisted of two ORBIT nodes to emulate UE and BS and one CONNECT node (COSMOS-IC) representing a remote file-server as shown in Fig. 4.19. From Table 4.6, considering TCP Cubic the E2E throughput (100 s interval) is observed to be 270 Mbps, whereas with TCP BBR the throughput is found to be 640 Mbps. A standard Linux package tool \texttt{tc} as described before is used to limit the bandwidth and introduce any loss (at the intermediate router) between the nodes. With just 0.1 % loss, the TCP performs so poorly over this high BDP link further motivating a need for a better transport protocol to overcome loss and fluctuations in the wireless channels. Please note, the results presented in this subsection are dated as the link between GW (ORBIT) and COSMOS-IC is shared by multiple parties.

Table 4.6: Evaluation of TCP throughput under loss on the COSMIC testbed (Dated: March 20, 2023)

<table>
<thead>
<tr>
<th>Transport Schemes</th>
<th>Loss percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0%</td>
</tr>
<tr>
<td>TCP New Reno</td>
<td>206 Mbps</td>
</tr>
<tr>
<td>TCP Cubic</td>
<td>270 Mbps</td>
</tr>
<tr>
<td>TCP BBR</td>
<td>640 Mbps</td>
</tr>
</tbody>
</table>

Moreover, we also evaluate the file transfer size over the mmWave channel using the trace as shown in Fig. 1.2, but max throughput is capped at 500 Mbps for a fair comparison between the protocols. The click implementation of mmCPTP is used and from Table 4.7, we see that, mmCPTP achieves a file transfer size close to the
theoretical limit, and a gain of 5.6x is observed corresponding to TCP Cubic over highly intermittent mmWave channel. Also, the results (TCP Cubic and New Reno) are in coherence with the ORBIT evaluation results provided in section 4.7.3. Future evaluation on the COSMIC includes integrating mmCPTP on the 5G NR SDR System (such as Amarisoft) and validating the E2E performance using outdoor field trials.

Table 4.7: File transfer size over mmWave channel as shown in Fig. 1.2 on the COSMIC testbed (Dated: March 20, 2023)

<table>
<thead>
<tr>
<th>Transport Schemes</th>
<th>File transfer size</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP New Reno</td>
<td>514 MB</td>
</tr>
<tr>
<td>TCP Cubic</td>
<td>1.67 GB</td>
</tr>
<tr>
<td>TCP BBR</td>
<td>5.53 GB</td>
</tr>
<tr>
<td>mmCPTP</td>
<td>9.46 GB</td>
</tr>
<tr>
<td>Theoretical limit</td>
<td>9.47 GB</td>
</tr>
</tbody>
</table>

4.8 Related Work

TCP [108] is the de-facto standard transport protocol for reliable transmission although it is known to perform poorly in certain wireless scenarios [109]. Various MAC and RLC [38, 110] layer schemes have been proposed to mask the last hop losses from the TCP layer, especially for mmWave networks. Yet, this causes increase in E2E latency as the packets are queued temporally during NLoS scenarios. Proxies or middlebox based solutions have also been proposed for TCP, specifically targeting wireless scenarios to improve performance. In [111], a milliProxy is described following TCP semantics. The milliProxy allows controlling maximum segment size (MSS) and congestion window separately between wired and wireless parts thus improving goodput and reduction in latency for high bandwidth delay product (BDP) mmWave links. The authors in [112] proposed a proxy based TCP architecture called mm-PEP, breaking TCP E2E semantics. The module installed at the base-station helps in improving the packet delivery ratio by maintaining the sending rate during LoS/NLoS switches. Various other schemes such as I-TCP [50], M-TCP [113] adopt a split TCP approach by breaking the TCP session between wired and wireless parts of the network for boosting throughput and improving BER. However, all the above discussed schemes have the same underlying
TCP session governed by probing, which often complicates the overall process.

Several other congestion control algorithms have been proposed particularly targeting high BDP links such as HighSpeed TCP [105], TCP Yeah [106], H-TCP [114], etc. Yet, they perform poorly over mmWave channel [14, 115] resulting in high latency and slow recovery in the event of handovers. TCP BBR, recently proposed by Google [107] operates by estimating bottleneck bandwidth and RTT. It enters a state called “probeRTT” having a smaller congestion window (several kB) periodically to measure real-time RTT. However, this is unfriendly for streaming applications and traffic over the Internet since the traffic might choose different routes and get different RTTs. The protocol is still under development [116] and also has issues on coexistence with other loss-based algorithms. QUIC [117] also proposed by Google deals better with packet loss and achieves faster connection establishment. Nevertheless, the problem of probing also exists in QUIC since it uses the same congestion control mechanism as in Reno or Cubic [118]. Hence, due to various implementation challenges, we restrict our study to New Reno and Cubic for the most part of the work. A detailed evaluation of these newer end-to-end transport protocols against mmCPTP, with more complex topologies and multiple users is left for future work.

ICN based architectures such as Named data networking (NDN) [48], use a receiver driven content retrieval model based on named content. Since ICN protocols are receiver driven, interest shaping can proactively control the congestion. The classical AIMD algorithm used by TCP for congestion control is used by NDN receivers to control interest rates. In addition to the consumer based congestion control, Hop-by-hop transport control [100] at routers have been well studied for FIAs such as NDN and MF [49]. Yet, the above described transport solutions still depend on AIMD for probing the bottleneck bandwidth and perform poorly as in TCP when there is high fluctuation in bandwidth and switching between alternate paths.
4.9 Summary

In this work, a novel cross-layer assisted pull based transport protocol is presented to overcome the deficiencies of TCP in mmWave networks. Fluctuating bit-rate and frequent handoffs between various available paths often underutilizes the mmWave links due to the gradual probing and additive increase multiplicative decrease behavior of TCP. The proposed scheme “mmCPTP” is based on an in-network proxy where the BS periodically fetches data based on cross-layer information from the lower layer (RLC/MAC) of the stack. In doing so, we avoid the slow start probing phase required to probe the available bandwidth. The evaluation of the proposed protocol was carried out using NS3 simulation considering high speed mobile UE and random walk user traces. The results show that when compared against conventional protocols, a gain of 4.5x is observed with respect to I-TCP (Cubic) in the presence of high RTT and packet loss. The protocol performs quite well in the presence of multiple users considering the NR stack, achieving a performance gain of 2.8x and 1.5x relative to New Reno and Cubic respectively. Also, when evaluated with a real-time implementation on the ORBIT [45] testbed, a gain of 17.2x and 5.7x is observed with mmCPTP, as compared to New Reno and Cubic respectively in scenarios with high RTT. The simulation results and the experimental validation demonstrate the feasibility of our cross-layer assisted transport protocol which has been shown to maintain better channel utilization in mmWave access scenarios. Future work includes extending the mmCPTP design to multi-homing scenarios with user mobility. Field trials using the outdoor COSMOS [47, 83] testbed are also under consideration.
Chapter 5
Conclusion

Future wireless networks (5G/6G) introduce new challenges at multiple layers of the protocol stack due to a lack of interoperability and heterogeneity. In this thesis, we presented protocol solutions to overcome some of these challenges with the objectives to provide better spectrum access, seamless mobility, and high-speed data transfer matching MAC layer limits. This work focused on designing protocol solutions that are based on 1) standardized spectrum models, 2) cross-layer analytics, 3) mobility protocols using the named-object network architecture, and 4) associated distributed algorithms. The following contributions are made.

Firstly, we address some of the challenges associated with dynamic spectrum access in the physical layer with the introduction of a spectrum management architecture and algorithm design that leverages Spectrum Consumption Models (SCMs) and Collaborative Interaction Language (CIL). A novel SCM-based deconfliction algorithm and spectrum access methods were developed for large scale wireless network environments. The simulation results and experimental demonstration on the ORBIT/COSMOS testbed validate the benefits of using SCMs and their capabilities to perform fine grained spectrum assignments in dynamic and dense communication environments.

Secondly, we described a novel distributed mobility management (DMM) scheme for the “named-object” information centric network (ICN) architecture intended to provide seamless data transfer with improved throughput during handovers. Two specific handover schemes namely, hard handoff with rebinding and soft handoff with multihoming are proposed and evaluated using system simulation and ORBIT testbed experimentation. The results validate the proposed ICN-influenced mobility management protocol to improve the network layer performance of future mobile users.
Finally, to overcome the inefficiencies of TCP in mmWave networks, we presented a novel cross-layer assisted pull based transport protocol supporting mmWave channel characteristics whilst achieving high throughput efficiency. The proposed scheme “mm-CPTP” is based on an in-network proxy where the BS periodically pulls the data based on cross-layer information from the buffers present in the lower layer (RLC/MAC) of the protocol stack. The protocol was evaluated using NS3 system simulation in 5G New Radio (NR) stack and trace driven end-to-end simulation considering random walk and high speed mobile UE channel models. The results show significant gains when compared against traditional TCP and Indirect-TCP.

5.1 Design trade-offs and integration

The protocol and algorithm solutions presented in this thesis are consistent and compatible with the existing network architecture. Considering SCM-based spectrum assignment, we use IEEE standardized spectrum models to quantify spectrum usage and perform compatibility analysis between multiple RF devices. We also use CIL (operating at the IP level), certified by DARPA along with our proposed changes to exchange SCMs between peers. These methods are well-defined and can work on any devices involving mmWave, sub 6 GHz, and also THz bands. Next, the distributed handoff schemes presented in chapter 3 can be readily integrated with the current IP-based network using an overlay, by having a naming layer over IP or UDP. Overlay solutions [119] may be preferable over clean-slate deployment due to their ease of implementation and cost advantages. The transport solution proposed in chapter 4 overcomes the TCP limitations over fluctuating mmWave links. We introduce the notion of a proxy, pull mechanism, and cross-layer support to reach optimal link capacity at the earliest. We also integrate our solution on the 5G NR stack making way for easy and standard adoption. The use of proxies has been well-studied and appreciated in the past [50,120]. Our design logic involves using TCP as much as possible (backhaul) with minimal proposed changes restricted only to the BS stack and the link between BS and proxy. A detailed performance trade-off study on using our solutions as compared to existing network infrastructure would be a useful next step.
5.2 Looking ahead

The protocol solutions presented in this thesis potentially overcome some of the major challenges in the deployment of future mobile wireless networks as listed in [121, 122]. Most importantly, wireless devices should be able to dynamically share precious wireless spectrum, and our proposed work in chapter 2 on SCMs, is an initial attempt to show how the declarative usage of spectrum improves channel selection procedures and ensures aggregate compatibility among other devices. Also, the usage of mmWave bands alleviates the spectrum demands, but a number of technical challenges pertaining to channel characteristics, mobility, and application performance [123] still remain for wide scale deployment and adoption. Chapter 3 and chapter 4 address some of these issues motivating the proposed solution to be integrated into future network stack to boost capacity and efficiency. Further evaluation with real 5G systems, integration with an open source 5G NR stack, and also applying machine learning aided techniques to enhance spectrum efficiency and mmWave performance remain as open challenges.
Acknowledgment of Previous Publications

(P1) **P. Netalkar**, C. E. C. Bastidas *et al*. “Scalable Dynamic Spectrum Access with IEEE 1900.5.2 Spectrum Consumption Models [under submission 2023].


(P6) S. Maheshwari, **P. Netalkar**, and D. Raychaudhuri, “Disco: Distributed control plane architecture for resource sharing in heterogeneous mobile edge cloud scenarios,” in 2020 IEEE 40th International Conference on Distributed Computing Systems [ICDCS 2020]
References


109


