# Application-Aware Cross-Layer Framework for Wireless Multihop Networks

Dem Fachbereich 18
Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung der Würde eines
Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von
Dipl.-Ing. Mousie Fasil
geboren am 19.02.1985 in Bad Soden am Taunus

| | |
|---|---|
| Erstgutachterin: | Prof. Dr.-Ing. Anja Klein |
| Zweitgutachter: | Prof. Dr.-Ing. Matthias Hollick |
| Tag der Einreichung: | 12.03.2021 |
| Tag der mündlichen Prüfung: | 29.10.2021 |

D 17
Darmstadt 2021

Fasil, Mousie : Application-Aware Cross-Layer Framework for Wireless Multihop Networks
Darmstadt, Technische Universität Darmstadt,
Jahr der Veröffentlichung der Dissertation auf TUprints: 2022
URN: urn:nbn:de:tuda-tuprints-202942
Tag der mündlichen Prüfung: 29.10.2021

Veröffentlicht unter CC BY-NC-SA 4.0 International
https://creativecommons.org/licenses/

# Kurzfassung

Der aktuelle wie auch zukünftiger Bedarf an mobiler und sozialer Kommunikation erfordert ein Umdenken bei der Entwicklung drahtloser Kommunikation. Die gängigen Ansätze, welche sich auf die Optimierung der Radioschnittstellen fokussieren, werden nicht mit den ständig steigenden Nutzeranforderungen skalieren. Das Internet der Zukunft erfordert ein drahtloses Kommunikationsnetzwerk, das sich nahtlos an wechselnde Umgebungs- und Serviceanforderungen anpassen kann. Insbesondere die Güteanforderungen von Diensten, welche durch die Nachfrage der Benutzer und die wachsende Vielfalt der Benutzerendgeräte getrieben werden, stellen eine große Herausforderung in Bezug auf die Verteilung von Inhalten dar. In dieser Arbeit wird die Verbesserung der drahtlosen Kommunikation unter Berücksichtigung von vier Hauptaspekten erforscht: Der erste Aspekt ist der Aufbau einer Multi-Layer-Lösung anstelle einer herkömmlichen Single-Layer-Lösung, um einen höheren Durchsatz zu erreichen. Hier werden die physikalische Schicht, die Medienzugriffsschicht und die Netzwerkschicht gemeinsam untersucht, um die Möglichkeiten aller drei Schichten zu nutzen. So wird ein einheitliches graph-basiertes Modell formuliert, um die verfügbaren Mechanismen auf den unteren drei Schichten gemeinsam zu betrachten. Der zweite Aspekt ist der Entwurf eines drahtlosen Multihop-Netzwerks, das mit ansteigender Anzahl von mobilen Geräten skalieren kann. Einerseits steigt die Anzahl der mobilen Geräte immer weiter an und damit auch die Dichte der mobilen Geräte in einem gegebenen Netzwerk. Auf der anderen Seite werden die Anforderungen und Fähigkeiten der mobilen Geräte immer vielfältiger und damit wächst die Heterogenität in einem drahtlosen Netzwerk. Dies führt zu der Schlussfolgerung, dass drahtlose Multihop-Netzwerke zukunftssicherer sind als drahtlose zellulare Netzwerke, die aus mehreren Basisstationen bestehen. Daher konzentriert sich diese Arbeit auf drahtlose Multihop-Szenarien, bei denen mehrere drahtlose Geräte das Netzwerk bilden und die Kommunikation zwischen ihnen über mehrere Zwischenstationen erfolgt. Der dritte Aspekt ist die Berücksichtigung der unterschiedlichen Anforderungen und Fähigkeiten der Applikationen. Die Vielzahl von Applikationen, die in drahtlosen Netzwerken eingesetzt werden, bringen unterschiedliche Anforderungen, z.B. an die Bandbreite, und Fähigkeiten, z.B. die Anpassung der Videoqualität, mit sich. Die Berücksichtigung dieser Anforderungen und Fähigkeiten zusätzlich zu einer Multi-Layer-Lösung kann die Leistungsfähigkeit der Netzwerke weiter steigern. In dieser Arbeit werden die Anforderungen und Fähigkeiten von adaptivem Videostreaming in ein transitionsfähiges, schichtenübergreifendes Optimierungsrahmenwerk integriert. Genauer gesagt, werden skalierbare Videocodierung und dynamisches adaptives Streaming über HTTP in das oben genannte Rahmenwerk integriert. Das neuartige transitionsfähige, schichtenübergreifende Optimierungsrahmenwerk passt Netzwerkunterstützungsstrukturen auf der Netzwerkschicht an, führt eine Ressourcenallokationen auf der Medienzugriffsschicht durch, schaltet zwischen Kommunikationstypen auf der physikalischen Schicht um und berücksichtigt die Fähigkeiten und Anforderungen von Anwendungen, z.B. von adaptivem Videostreaming, auf der Applikationsschicht. Der vierte Aspekt ist die Nutzung der Aggregation verteilter Inhalte. Hier werden Inhalte über das gesamte Netzwerk zwischengespeichert und können dann aggregiert werden, um von Benutzern im Netzwerk konsumiert zu werden. Aktuelle Forschungsarbeiten zeigen, dass vielversprechende Gewinne erzielt werden können, wenn Inhalte auf mobilen Geräten zwischengespeichert werden, allerdings meist für konventionelle drahtlose Netzwerke. Daher wird in dieser Arbeit die Auswirkung mobiler Zwischenspeicher untersucht, in denen beliebte Inhalte zwischengespeichert und über mehrere Geräte im Netzwerk aggregiert werden. Im Detail wird das transitionsfähige, schichtenübergreifende Optimierungsrahmenwerk für das Aggregieren verteilter Inhalte vorgestellt, das sowohl bereits auf mobilen Geräten zwischengespeicherte Inhalte als auch das Umschalten zwischen Mechanismen auf der physikalischen Schicht und der Netzwerkschicht nutzt, um die Inhalte unter wechselnden Netzwerkbedingungen optimal an alle Ziele zu liefern.

# Abstract

Current and future mobile and social communications require a rethinking in the development of wireless communication. Optimizing the radio transmission method is not going to scale with the ever increasing user demands. The future internet requires a wireless communication network which can adapt seamlessly to changing environments and service requirements. Especially, service requirements driven by user demand and expanding user device diversity raise a key challenge with respect to content distribution. In this work, research is conducted to improve wireless communication by considering four main aspects: The first aspect is to build a multi layer solution, instead of a conventional single layer solution to achieve higher throughput gains. Here, the physical layer, the medium access layer and the network layer are studied together to utilize capabilities across all these three layers. Thus, a unified graph model is formulated to adapt available mechanisms on the lower three layers in a joint manner. The second aspect is to envision a wireless multihop network which can scale with the increasing number of mobile devices. On the one hand, the number of mobile devices is ever increasing and so is the density of mobile devices in any given network. On the other hand, the requirements and capabilities of mobile devices are becoming more diverse and hence the heterogeneity in a wireless network is growing. This leads to the conclusion that a wireless multihop network is more future proof compared to a wireless network composed only of several base stations. Therefore, the research is focused on wireless multihop scenarios where multiple wireless devices form the network and communication between them occurs over multiple hops. The third aspect is to incorporate different requirements of applications and capabilities of applications. The plethora of applications used in wireless networks come with different sets of requirements, e.g. bandwidth, and capabilities, e.g. adaption of the video quality. Taking into account these requirements and capabilities in addition to a multi layer solution can further increase the performance. In this work, the requirements and capabilities of adaptive video streaming are integrated into an application-aware cross-layer framework. More precisely, scalable video coding and dynamic adaptive streaming over HTTP are integrated into the aforementioned framework. The novel application-aware cross-layer framework adapts network support structures at the network layer, performs resource allocation at the medium access layer, switches between communication types at the physical layer and takes into account the capabilities and requirements of applications, e.g. adaptive video-streaming, at the application layer. The fourth aspect is to utilize aggregation of distributed content, where content is cached over the whole network and can than be aggregated to be consumed by users in the network. Recent research shows promising gains achievable when content is cached at mobile devices, but mostly for single hop wireless networks. Hence, the impact of mobile content caching where popular content is cached and aggregated over multiple devices in a network is investigated in this work. In more detail, a content delivery framework which jointly exploits content already cached at mobile devices as well as switching between mechanisms at the physical layer and the network layer in order to optimally deliver the content to all destinations under changing network conditions is proposed.

# Contents

**1 Introduction** **1**
- 1.1 Content Distribution and Aggregation in Wireless Multihop Networks . 1
  - 1.1.1 Wireless Multihop Networks . . . . . . . . . . . . . . . . . . . . . 2
  - 1.1.2 Adaptive Video Streaming . . . . . . . . . . . . . . . . . . . . . . 2
  - 1.1.3 Caching in Wireless Networks . . . . . . . . . . . . . . . . . . . . 3
  - 1.1.4 Adaptation and Transitions . . . . . . . . . . . . . . . . . . . . . 4
- 1.2 State of the Art . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
  - 1.2.1 Cross-Layer Approaches . . . . . . . . . . . . . . . . . . . . . . . 4
  - 1.2.2 Adaptive Video Streaming . . . . . . . . . . . . . . . . . . . . . . 6
  - 1.2.3 Caching in Wireless Networks . . . . . . . . . . . . . . . . . . . . 7
  - 1.2.4 Transitions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
- 1.3 Open Issues . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10
- 1.4 Contributions and Thesis Overview . . . . . . . . . . . . . . . . . . . 11

**2 System Model** **13**
- 2.1 System Overview . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
- 2.2 Network Layer . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
  - 2.2.1 Graph Representation . . . . . . . . . . . . . . . . . . . . . . . . 13
  - 2.2.2 Network Support Structures . . . . . . . . . . . . . . . . . . . . . 14
- 2.3 Medium Access Layer . . . . . . . . . . . . . . . . . . . . . . . . . . . 15
- 2.4 Physical Layer . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
  - 2.4.1 Transmission Mechanisms . . . . . . . . . . . . . . . . . . . . . . 17
- 2.5 Application Layer . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20
  - 2.5.1 Adaptive Video Streaming . . . . . . . . . . . . . . . . . . . . . . 20
  - 2.5.2 Caching . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21

**3 Transition-Enabled Cross-Layer Framework** **25**
- 3.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
- 3.2 Unified Graph-Model . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
  - 3.2.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
  - 3.2.2 Network Support Structures . . . . . . . . . . . . . . . . . . . . . 26
  - 3.2.3 Transmission Mechanisms . . . . . . . . . . . . . . . . . . . . . . 26
    - 3.2.3.1 Node Virtualization . . . . . . . . . . . . . . . . . . . . . 27
    - 3.2.3.2 Quantization for Node Virtualization . . . . . . . . . . . 29
  - 3.2.4 Scheduling . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31
    - 3.2.4.1 Edge Coloring . . . . . . . . . . . . . . . . . . . . . . . . 31
    - 3.2.4.2 Vertex Coloring . . . . . . . . . . . . . . . . . . . . . . . 32

3.2.4.3 Extended Graph Coloring . . . . . . . . . . . . . . . . 33
3.3 Single-Rate Multi-Source Optimization Problem . . . . . . . . . . . . . 35
3.4 Transitions in the Cross-Layer Framework . . . . . . . . . . . . . . . . 37
3.5 Performance Analysis . . . . . . . . . . . . . . . . . . . . . . . . . 39
3.5.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . 39
3.5.2 Transitions in Fixed Wireless Multihop Networks . . . . . . . . 39

**4 Adaptive Video Content Distribution using DASH with AVC 47**
4.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 47
4.2 Adaptive Bitrate Streaming . . . . . . . . . . . . . . . . . . . . . . 47
4.3 Graph Extension: Video Representations as Virtual Sources . . . . . . 48
4.4 Multi-Source Optimization Problem with Independent Sources . . . . . 49
4.5 Rate Mapping Problem . . . . . . . . . . . . . . . . . . . . . . . . . 52
4.6 Performance Analysis . . . . . . . . . . . . . . . . . . . . . . . . . 54

**5 Adaptive Video Content Distribution using RTP with SVC 61**
5.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 61
5.2 Layered Coding Streaming . . . . . . . . . . . . . . . . . . . . . . . 61
5.3 Graph Extension: Video Layers as Virtual Sources . . . . . . . . . . . 62
5.4 Multi-Source Optimization Problem with Dependent Sources . . . . . . 62
5.5 Rate Mapping Problem . . . . . . . . . . . . . . . . . . . . . . . . . 65
5.6 Performance Analysis . . . . . . . . . . . . . . . . . . . . . . . . . 66

**6 Aggregation of Distributed Cached Content 71**
6.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 71
6.2 Caching in Wireless Networks . . . . . . . . . . . . . . . . . . . . . 71
6.3 Graph Extension: Super-Source Virtualization for Distributed Content 72
6.4 Multi-Source Multi-Destination Optimization Problem . . . . . . . . . 74
6.5 Performance Analysis . . . . . . . . . . . . . . . . . . . . . . . . . 77

**7 Conclusions 81**
7.1 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 81
7.2 Outlook . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 83

**List of Acronyms 85**

**List of Symbols 87**

**Bibliography 89**

**Own Publications 95**

**Supervised Student Theses** **97**

**Lebenslauf** **101**

# Chapter 1

# Introduction

## 1.1 Content Distribution and Aggregation in Wireless Multihop Networks

Today, multimedia content is increasingly consumed through wireless networks, which is a challenge considering that multimedia content consumes a large fraction of the scarce resources in wireless networks. Recent advances in wireless technologies tackle this challenge of limited resources. First, 5G improves the spectral efficiency [OBB+14] through advancement of physical layer and medium access layer techniques. Second, device-to-device (D2D) communication [LZLS12], which enhances infrastructure-based wireless networks, further improves resource utilization with respect to the reuse of resources. However, the demand for multimedia content is increasing rapidly, and it is estimated that multimedia content will contribute to three-fourths of the overall mobile traffic [15], hence is going to consume a large part of the available resources in wireless networks.

Furthermore, the landscape of content distribution and aggregation services, from telephony to Artificial Reality (AR) application, has become more and more diverse, where current networks have to fulfill a wide range of service requirements. One of the biggest drivers in the field of content distribution and aggregation is video-streaming, which requires a network which is capable of efficient resource allocation and is adaptable to diverse services and environments. Especially, the lower three layers are a good starting point to enable a more capable network, since resource allocation is a key task of these layers. At the same time, content distribution has become more flexible through adaptive video streaming approaches, e.g. Dynamic Adaptive Streaming over HTTP (DASH) and Scalable Video Coding (SVC). Therefore, one promising way to improve content distribution and aggregation in wireless networks is by combining the advances and capabilities from different layers. In more detail, a solution is required, which interlocks the requirements and properties of applications on the one hand, while utilizing the available capabilities of a wireless network. At the same time, an approach is required which is able to adapt to different environments by changing the mechanisms available at the lower three layers. Hence, in this thesis the combination of network layer, medium access layer, physical layer and application layer is researched, which can provide gains through providing optimal combinations of mechanisms on different layers with respect to the current condition of the environment.

### 1.1.1 Wireless Multihop Networks

Wireless multihop networks have become more widespread through the rise of Internet of Things and the fourth industrial revolution. The core idea of a wireless multihop network is that a multitude of devices, called nodes, are interconnected with each other over the wireless medium, called links. Nodes form a wireless multihop network with their connections, where the term multihop emphasizes that nodes can reach other nodes either over a direct wireless link or via multiple hops when a direct link is not present. One main challenge in wireless multihop networks is to achieve high rate communication avoiding collision and interference, while ensuring the connectivity of all nodes inside the wireless multihop network.

### 1.1.2 Adaptive Video Streaming

The transmission of multimedia content over the Internet faces a plethora of challenges, like bandwidth fluctuation, buffering and diverse screen resolutions with respect to user end devices. Adaptive Video Streaming addresses these challenges by adapting to the network conditions and by considering the properties of end user devices. In broad terms, adaptive video streaming can be divided into two categories. The first category provides multiple versions of a video and switches between the different versions of a video, in order to adapt the provided video stream to the current network conditions. The second category introduces a scalable video where a video is composed of multiple video layers. This allows to add or remove video layers of a video to a stream, thus adapting the video stream to the current network conditions. The first category of adaptive video streaming is represented by Dynamic Adaptive Streaming over HTTP (DASH), while the second category is represented by Scalable Video Coding (SVC). The fundamental idea behind DASH can be explained by a simple example illustrated in Fig. 1.1. In the example, a server transmits a non-scalable video through a network, represented by the pipe, to a user. Furthermore, the server stores multiple versions of the video, i.e. in different qualities with different bandwidth requirements. Assume the network bandwidth changes, e.g. due to increased traffic volume in the network, and further the user is able to measure the available bandwidth. Then a user can either request a different version of the video from the server. Alternatively, a user can signal the available bandwidth to the server, thus allowing the server to switch the version of the video. As a result, the video stream from the server to a client adapts the selected version with respect to the available bandwidth. Therefore, DASH introduces adaptive video streaming through multiple representations of a video and by switching between these multiple representations when the network environment changes. However, there

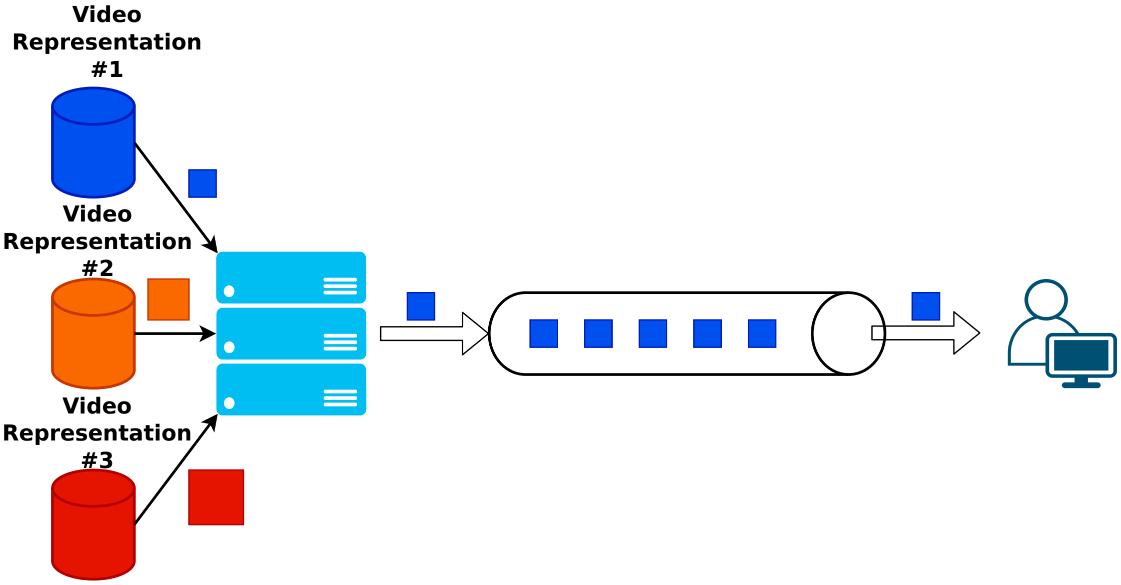


Figure 1.1: A simple client-server example, transmitting non-scalable video from the server to the client.

are two challenges introduced by utilizing DASH for adaptive video streaming. The first one is storage, every representation of a video has to be stored on the server, depending on the number of representations and videos, storage can become a huge cost factor. Second, network resource consumption becomes a challenge, because DASH introduces additional video streams, for each video representation it transmits. Hence, every video stream utilizing DASH could consume a multitude of network resources. As already mentioned, SVC introduces the concept of a layered video, where a video is encoded into several video layers. The advantage of utilizing a SVC encoded video is that not multiple representation have to be stored, e.g. at a server, compared to DASH. Instead, a video stream with a SVC encoded video can be adapted by adding or removing a video layer depending on the available bandwidth. Therefore, adaptive video streaming utilizing SVC can also handle changes in the network. However, there are two challenges introduced by utilizing SVC for adaptive video streaming. First, SVC introduces overhead, for every video layer generated, additional overhead is required, which increases the overall size of the SVC encoded video. Second, also SVC can lead to an increased network resource consumption, because adaptive video streaming utilizing SVC introduces a data stream for every layer it transmits.

### 1.1.3 Caching in Wireless Networks

The application of content caching in wireless networks, especially on end user devices, is a promising approach to improve the utilization of network resources in wireless

networks. Here, multimedia content is stored closer to the user, e.g., at a base station or even directly at mobile devices [WCT+14].
Especially, content caching at mobile devices has shown promising results, which can increases significantly the achievable throughput in wireless networks [JCM16]. Caching in wireless networks can improve network resource utilization. However, in order to achieve high network resource utilization with caching in wireless networks, several aspects of caching have to be taken into account, i.e. what to cache, where to cache and how to design caching policies [MN14], [JTLC14], [JHJC15].

### 1.1.4 Adaptation and Transitions

Wireless communication systems are designed with the application area and the wireless environment in mind, since designing a general purpose system that fits all applications and requirements is hard to realize. Although, wireless communication system can perform adaption, e.g. adjusting the transmission power, to react to changes in the environment, adaption cannot address changes in the requirements of an application. In recent years, the idea of mechanism transitions is studied, to address the limits of wireless communication systems capabilities to adapt to changes in application and network requirements. Transitions is a new paradigm, which means that in the presence of equivalent mechanisms a switch from one mechanisms to the other and vice versa is performed. A transition is performed when the network conditions or the application requirements change. In this thesis, switching between an optimal combination of mechanisms on different layers is referred as a cross-layer transition and the prime focus with respect to transitions.

## 1.2 State of the Art

### 1.2.1 Cross-Layer Approaches

Obtaining the optimal multicast rate in wireless multihop networks (WMNs) is a challenging task. The goal is to obtain the maximum achievable rate between one source and an associated group of destinations, which are interested in the same information (e.g. audio or video). In a single source multicast scenario, where only one source is routing to a group of destinations, the problem can be expressed as a linear optimization problem [SE07]. The solution for a multi-source multicast scenario, where multiple sources route to their respective group of destinations, can be obtained by determining a multicast tree for each source, which is NP-hard [YLYL06].

In addition, the wireless communication introduces interference and collisions, where a collision occurs if a node tries to transmit and receive at the same time. Therefore, scheduling needs to be performed in addition to routing. Another aspect in WMNs is the utilization of the broadcast gain of the wireless medium, especially in combination with routing. The aim is to make use of the omnidirectional communication such that the network benefits by either higher achievable throughput or by minimized energy consumption.

One approach to address routing and resource allocation in wireless multihop networks is addressed in [Loc15] and [Hoh20], where the authors propose corridor-based routing. In corridor-based routing multiple neighbor nodes cooperate to transmit data from one source to one destination, which allows to tune between efficient or robust transmission. Another approach to address the routing problem on the network layer is to use network coding [ACLY00], which allows the formulation of a linear optimization problem as in [LLJL05]. Network coding was introduced by Ahlswede et al. in [ACLY00]. In their pioneering paper, they showed that assuming information as flows is not achieving optimality in a multicast scenario. Instead of replicating and forwarding the information, the information should be combined at intermediate nodes. There are several works on wired [RSW05], wireless [SE07], [LLJL05], [NBK12], and hybrid networks [YLYL06] with regards to network coding. Nevertheless, network coding does not always provide a gain over plain routing as seen in [LLJL05]. Hence, Li et al. [LLJL05] present an approach which utilizes plain routing and network coding, but do not apply it for WMNs.

In [SE07], a heuristic to perform conflict-free scheduling at the medium access layer is proposed. The heuristic separates the network so that nodes do not interfere with each other and do not transmit and receive at the same time. Although the approach is used in a wireless network, they assume equal link capacities. In [NBK12], the above approach from [SE07] is extended with virtualization. In the context of [YLYL06] and [NBK12], virtualization utilizes the broadcast gain of the wireless medium by adding virtual nodes to the network. Virtual nodes are connected to multiple physical nodes over virtual links. The virtualization represents a physical layer broadcast at the network layer. Although the authors in [NBK12] address all three layers, they also assume fixed link capacities, which does not hold in a WMN.

Therefore, a framework is needed which can switch between the routing mechanisms at the network layer and which can also switch between the communication types unicast, multicast and broadcast communications at the physical layer. This is not provided by current approaches in the literature, especially not for WMNs with unequal link capacities. Furthermore, evaluating the behavior of current approaches and a framework which can perform seamless switching between mechanisms in the presence of unequal link capacities needs further investigation.

### 1.2.2 Adaptive Video Streaming

The performance of a video-streaming service in a wireless multihop network (WMN) highly depends on the degree of user heterogeneity. User heterogeneity includes the diversity of the mobile device, concerning, e.g., screen resolution and processing power as well as device connectivity and channel conditions [ZAP+07]. Hence, in the presence of one or multiple weak users the performance of a video-streaming service will be low in terms of service quality and resource utilization [ZGZ+09].
At the application layer (APP), adaptive video streaming technologies have been proposed in order to address device heterogeneity as well as changing network conditions [YA14]. Two adaptive streaming approaches. The first one is DASH, which provides multiple versions of a video encoded with AVC. The second approach is to utilize a simple protocol e.g. RTP but encode the video itself with SVC, hence, the video itself is adaptive [SMW07], [SSH+11].
SVC breaks the limitation of video-streaming by introducing video layers, which allows to serve each user with an individual video quality [SMW07]. However, SVC cannot address the variations in a WMN like changing network topology, available resources and channel conditions. Since, SVC cannot adapt the lower layers, a cross-layer approach is needed.
Several works [ZYZ+06], [LE13] and [CQZ+12] present a cross-layer approach, where they combine SVC and network coding [ACLY00]. The papers show that considering APP and network layer together increases the performance compared to approaches without a layered video distribution scheme. Nevertheless, APP and NET cannot adapt to changes in available resources and channel conditions. Thus, they cannot adapt to variations at the data link layer and at the physical layer.
In other related works [VCO+07], [KSL07] and [DTXL11], APP is combined with data link layer. At the data link layer, a resource allocation problem is formulated which maps SVC video layer rates such that every user can receive at least the basic video layer and the remaining resources are allocated to the enhancement video layers. The presented results are limited to the one hop case and different communication types, e.g. broadcast and unicast, where not taken into account.

Today, the standard gaining the current momentum of both academia and industry is DASH. In DASH, multiple representations of a video are stored, each corresponding to a certain quality [NWGG14]. From a network-centric view, both adaptive video streaming approaches introduce multiple data streams, but SVC-based video-streaming and DASH utilize the network resources differently.

SVC-based video-streaming introduces a data stream for every video layer, where a video layer with a higher order is distributed when all layers with a lower order can be received. Thus, multiple dependent data streams flow through the network to the users, when SVC-based video-streaming is utilized. In DASH, each user aims to obtain the video in the best quality with respect to his device capabilities. This may result in multiple independent data streams, where users are receiving different video representations of the same content. The multiple independent data streams introduced with DASH lead to new challenges with regard to the utilization of network resources.

Since adaptive video streaming cannot properly address the variations in a WMN like changing network density, available resources and interference related to the lower layers, a cross-layer approach combining adaptive video streaming and lower layer transitions is needed. There are several related works which combine adaptive video streaming with a cross-layer approach.

In [ZYZ+06],[LE13] and [CQZ+12], SVC at the APP and network coding [ACLY00] at the network layer are utilized to maximize the performance in terms of rate or video quality. Nevertheless, APP and NET cannot adapt to variations at the medium access layer and at the physical layer. Thus, they cannot adapt to changes of the available resources and channel conditions. In [KSL07] and [DTXL11], a resource allocation problem at the MAC is formulated which maps SVC video layer rates to PHY rates such that every user can receive at least the base layer, while the remaining resources are allocated to the enhancement layers. Nevertheless, the presented results are limited to the one hop case and different communication types, e.g., broadcast and unicast, are not taken into account.

In [ZGL+14], the authors are combining DASH at the APP with cross-layer information at the MAC and the PHY. They presented results for cellular networks, but are not considering the multihop case and are not taking into account the switching between different communication types, e.g., broadcast and unicast.

### 1.2.3 Caching in Wireless Networks

Recent research discusses the benefits and implementation of caching in wireless networks. Nevertheless , the amount of work on how to deliver cached content is limited. Especially the question how to deliver and aggregate cached content in wireless networks.

Some work on content delivery of cached content is available for wired networks, which has been investigated under the paradigm of information-centric networking [ADI+12; FYH+14]. In [LTGK13], an analytical content delivery cross-layer framework for wired networks is proposed, which serves multiple users by using network coding and caching techniques. A combination of caching policies and content delivery in wired networks has been investigated in [BGW10], where distributed caching algorithms for minimizing the bandwidth costs in a hierarchical cache network with a tree structure are proposed.

A few steps have been made to take into account lower layer aspects into the delivery of cached content in wireless networks. In [JTLC14], message coding and multicasting for a single-hop scenario with one source and multiple destinations is considered. Additionally, the authors show that by caching a fraction of the content directly at the destinations, a common message can be forwarded to all users by utilizing multicasting, to improve the achieved rate. In [PIT14], joint caching and routing algorithms for small cell networks are proposed, which minimize the traffic at the macro base station by jointly selecting the cache content of small base stations and assigning each user request to a small base station that has the requested content in their cache. The small base station then delivers the content by utilizing a single-hop unicast transmission. In [GPT13], content replication and routing in multihop networks are considered jointly and scaling laws for the required link capacity are presented. In [VSR03], the authors combine caching and multipath routing to improve the reliability in wireless multihop networks. In [XNI+15], a cross-layer approach for a cache-enabled wireless relaying network is presented, where the goal is to minimize the content delivery time.

However, the works mentioned above are not considering changing network conditions at the lower three layers. These changes can be due to variations in the network topology at the network layer, in the availability of resources at the medium access layer and the ever changing quality of the wireless channel at the physical layer. An approach is required, where the traffic demand can be handled in a much more flexible manner by delivering content to destinations from close-by nodes which have already cached the content.

Thus, investigating content delivery where content is already cached at mobile devices with a focus on wireless multihop networks is crucial to further improve performance. In more detail, examining a content delivery strategy which utilizes content cached at mobile devices, while taking into account the underlying network conditions like the network topology, the available resources and the channel quality can lead to significant gains.

### 1.2.4 Transitions

A shift in paradigms towards mechanism transition in wireless communications, due to the high dynamics in wireless network environments and diversity of applications is needed [AWB+19]. A growing amount of research is conducted to show and utilize transitions in dynamic wireless environments.

In [FRLB15a] and [FRLB15b] a rule based system is proposed which executes transitions triggered by events and by evaluating the current conditions. Transition frameworks based on game-theory are proposed in [MMA+16] and [MK19], where the individual nodes are selfish, but are incentivized to form a network and participate in forwarding multimedia content.

Research investigating user generated content and transitions is conducted in [Wil16]. In [Sto18] the advantages of adaptive video streaming and transitions is investigated from an application-centric view. In [RRZ+16] and [RRL+16] a framework is proposed which performs transitions in a publication/subscribe system for mobile applications, where the transitions are performed between filter mechanisms. A transition-enabled monitoring framework is proposed in [RLR+18], the framework provides monitoring in wireless mobile networks by utilization transitions from a centralized, decentralized or hybrid monitoring approach when the environment changes.

In [RRKS17] a transition-enabled framework is discussed which reduces the stress on challenged networks by introducing offloading approaches which perform transitions through role assignments. In [SSHE15] the authors demonstrate a combination of scalable video coding and transitions on the physical layer on real hardware.

## 1.3 Open Issues

The current literature provides approaches where the lower three layers are modeled jointly. At the PHY, usually either unicast or broadcast are considered. Hence, the following questions arise:

**Q1:** How can the PHY transmission mechanisms unicast, broadcast and in addition multicast be considered jointly with the different mechanisms at the MAC and the NET?

**Q2:** How to ensure collision-free scheduling at the MAC when the three transmission mechanisms unicast, broadcast and multicast are utilized simultaneously at the PHY?

**Q3:** How to model and utilize transitions at the lower three layers?

Video streaming is responsible for 55% of the mobile traffic [15]. At the application layer, adaptive video streaming technologies have been proposed in order to address changing network conditions. In the last decade, two methodologies for adaptive video streaming systems were proposed: one is utilizing RTP with SVC and the other is DASH with AVC. Both DASH with AVC as well as RTP with SVC have unique properties and hence the following questions arise:

**Q4:** How can the properties and requirements of DASH with AVC be modeled and expressed jointly with the properties and requirements of the lower three layers?

**Q5:** How can the properties and requirements of RTP with SVC be modeled and expressed jointly with the properties and requirements of the lower three layers?

Recent work in caching for wireless mobile networks have gained momentum, since caching can partially alleviate the ever growing need for bandwidth. Here content or parts of a content are available at multiple devices in the network by caching content closer to or directly at the user device. The literature answers the questions how to cache, where to cache and what caching policy to apply. Nevertheless, questions regarding the selection and transportation of content cached at different nodes is not answered for wireless multihop networks.

**Q6:** How to model content or parts of a content cached at one or multiple nodes?

**Q7:** How to select nodes to deliver the cached content to the users?

## 1.4 Contributions and Thesis Overview

In the following, the structure and the content of the thesis is given. In more detail, the open questions formulated above are answered in the subsequent chapters, as well as applicable solutions are presented.

Chapter 2 gives an overview of the wireless multihop system and the layers APP, NET, MAC and PHY.

Chapter 3 addresses the questions **Q1**, **Q2** and **Q3**. Here, the incorporation of the lower three layers NET, MAC and PHY into a novel graph model is presented. In this unified graph model, the concept of node virtualization is presented and extended to represent unicast, broadcast and multicast. Furthermore, a novel collision-free scheduling approach is introduced to jointly schedule transmissions taking into account the PHY transmission mechanisms unicast, broadcast and multicast.

Moreover, a sum rate optimization problem is formulated. The combination of the sum rate optimization problem and the unified graph leads to the transition-enabled cross-layer framework, which performs transitions in the network. The transition-enabled cross-layer framework is application-agnostic. Hence, the framework includes the properties and requirements of the lower three layers, but does not take into account the APP.

An extension of the transition-enabled cross-layer framework for APP awareness for different applications is presented in Chapter 4 to 6.

In Chapter 4, the properties and requirements of the adaptive video-streaming application DASH with AVC are modeled by extending the unified graph model. Hence, an answer to **Q4** is given in this chapter. Furthermore, a multi-source optimization problem with independent sources is formulated. A rate mapping problem is formulated to map the data rate requirements of the different video representations in DASH with AVC at the APP to rate requirements at the PHY. The rate mapping problem is a mixed integer non-linear problem and hence an efficient heuristic solution is proposed.

In Chapter 5, the adaptive video-streaming application RTP with SVC is discussed. In this chapter, **Q5** is answered by extending the unified graph to represent the different video layers introduced by SVC. Moreover, a multi-source optimization problem with dependent sources is formulated. A rate mapping problem is formulated to map the data rate requirements of the SVC video layers at the APP to rate requirements at the PHY. Also, this rate mapping problem is a mixed integer non-linear problem and hence appropriate heuristic approach is presented.

In Chapter 6, the application-aware and transition-enabled framework is extended by the concept of super-source virtualization to represent content or parts of a content cached at multiple nodes in the wireless multihop network. Moreover, a multi-source multi-destination optimization problem is formulated, which selects one or multiple nodes to forward cached content through the network to the destinations. Thus, **Q6** and **Q7** are answered in Chapter 6.

Throughout Chapters 3 to 6, the performance of the different cross-layer frameworks is evaluated by means of simulation and compared with conventional schemes which cannot perform transitions.

Finally, Chapter 7 concludes the thesis and gives some final remarks regarding future work.

# Chapter 2
# System Model

## 2.1 System Overview

The conventional research and development approach in wireless communications will not provide enough gains with respect to bandwidth to cope with rapid increase in mobile internet traffic. Thus, instead of investigating the different layers individually a cross-layer approach is required. Especially, a cross-layer approach which takes into account lower layer aspects at the NET, MAC and PHY together with higher layer aspects of the APP. Therefore, this chapter first gives a system overview and presents the layers NET, MAC, PHY and APP. First in Section 2.2, the NET is discussed, where a graph representation and network support structures are introduced. Next in Section 2.3, collisions and time-based scheduling are discussed at the MAC. Third in Section 2.4, the channel model and transmission mechanisms are presented. Finally in Section 2.5, at the APP a short introduction to adaptive video streaming and caching is given.

## 2.2 Network Layer

At the network layer, the end-to-end routes between source and destination nodes are determined. The routes are determined based on the available routing information, which usually consists of a list of neighboring nodes and a metric sorting the respective neighbors. The routing metric can be anything from hop count to throughput. An example of a wireless multihop network is depicted in Fig. 2.1. The wireless multihop network is illustrated as a graph, which consists of vertices representing nodes in the network and edges representing links between nodes. Furthermore, two network support structures are illustrated in Fig. 2.1, where $S_1$ utilizes a tree structure to forward data to the nodes $D_1$ and $D_2$, while $S_2$ is utilizing a butterfly structure to forward data to $D_3$ and $D_4$.

### 2.2.1 Graph Representation

As mentioned earlier, wireless multihop networks are represented as graphs, hence a short introduction of terminology is given next. The modeling of wireless multihop

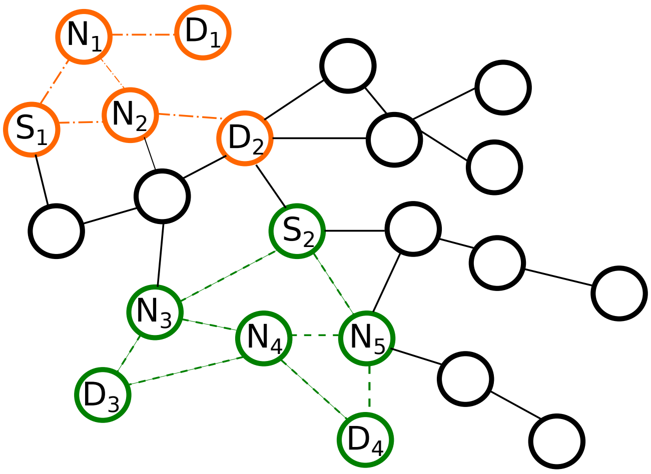


Figure 2.1: Example of a wireless multihop network.

networks is based on directed graphs $\mathcal{G} = (\mathcal{N}, \mathcal{L})$. The graph $\mathcal{G}$ contains a set $\mathcal{N}$ of nodes in the network and a set $\mathcal{L}$ of links representing connections between nodes. The set of nodes contains three subsets, the subset $S \subset \mathcal{N}$ of source nodes, the subset $D \subset \mathcal{N}$ of destination nodes and the subset $N \subset \mathcal{N}$ of relay nodes, where each node can be in one, two or all of the subsets. The link between two nodes $N_i$ and $N_j$ is denoted as a directed edge $l$, where $N_i$ is the transmitting node and $N_j$ is the receiving node.

### 2.2.2 Network Support Structures

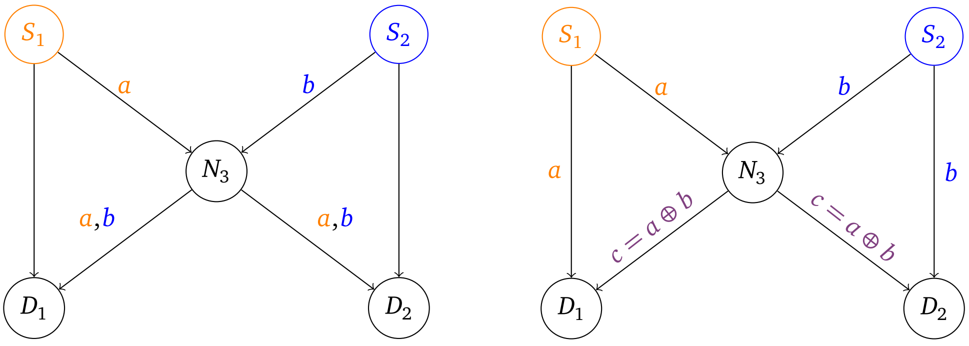
In a wireless multihop network, it is beneficial to utilize different network support structures at the NET, in order to adapt to changes in the network topology. A network support structure identifies a set of nodes favorable to deliver messages between source and destination nodes. Throughout this thesis, butterfly and tree are considered as network support structures.

In a tree structure, the aim is to use the smallest number of nodes to forward messages, which enables selected nodes to better utilize the shared wireless medium. The objective of a butterfly structure is to exploit as many independent paths as possible, in order to apply network coding [ACLY00] between different messages flowing through the wireless multihop network.

The advantage of network coding is illustrated in Fig. 2.2, where source nodes $S_1$ and $S_2$ want to transmit their respective messages to destinations $D_1$ and $D_2$. In Fig. 2.2 (a), a tree is utilized where $N_3$ simply receives and forwards the incoming messages $a$ and

$b$. Thus, in total four transmissions are required to forward the two messages to both destinations when a tree structure is used. In comparison, only three transmissions are required when network coding is utilized, as depicted in Fig. 2.2 (b). This is done in the following steps, first $S_1$ broadcasts message $a$ to $N_3$ and $D_1$. Next, $S_2$ broadcasts message $b$ to $N_3$ and $D_2$. In the third and final transmission, relay $N_3$ first combines the two incoming messages $a$ and $b$ from $S_1$ and $S_2$ into one outgoing message $c = a \oplus b$, and then broadcasts message $c$ to $D_1$ and $D_2$. Since $D_1$ and $D_2$ received message $c$ and either message $a$ or $b$, they can retrieve message $b$ or $a$ respectively, by performing an XOR operation. Thus, the butterfly structure leads to an increase in the overall system throughput, by utilizing multiple paths and relays.

The example shows the trade-off between the two network support structures, but the performance of the network does not only depend on the network topology. Furthermore, the scheduling at the MAC and the available PHY mechanisms play a major role. Therefore, to obtain an overall overview of the system the MAC and PHY are discussed next.

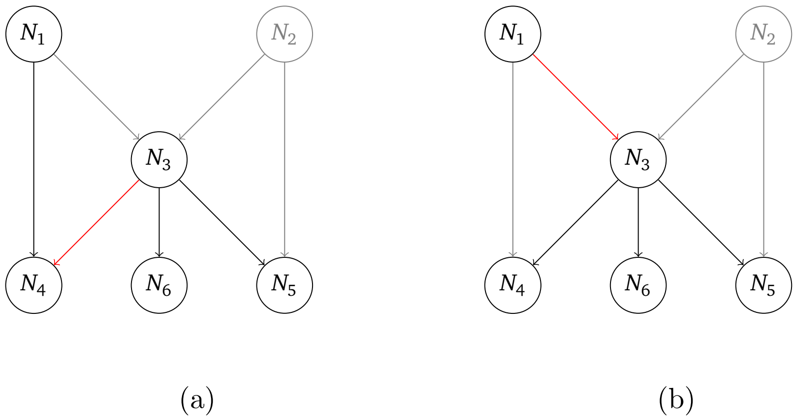


(a) (b)

Figure 2.2: Utilization of network support structures (a) tree and (b) butterfly.

## 2.3 Medium Access Layer

In this section, the scheduling at the MAC is discussed. The scheduling of resources at the MAC is an important aspect, since concurrent transmissions through the wireless medium may introduce collisions and hence degrade the utilization of available resources. Throughout this thesis, two types of collisions are considered. A collision



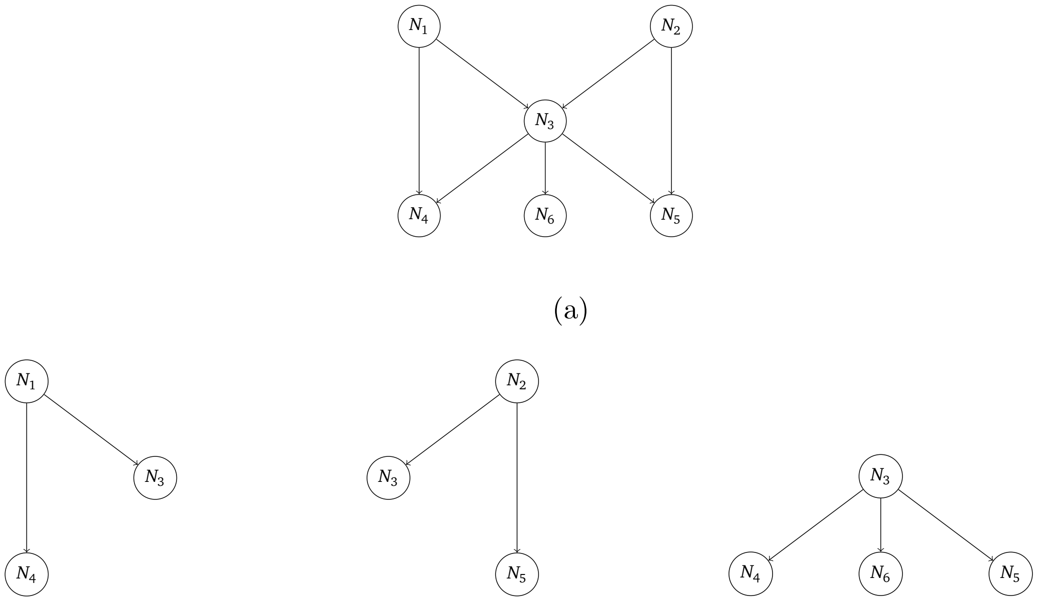
(a) (b)

Figure 2.3: Two types of collisions (a) due to multiple transmissions and (b) due to violation of half-duplex constraint.

occurs either when a node receives multiple transmissions at the same time or when a node is transmitting and receiving at the same time. One reason is that nodes cannot simultaneously process multiple transmissions, since each node is only equipped with a single antenna. Another reason is that all nodes are assumed to be operating in half-duplex mode, where a node cannot be active as a transmitter and as a receiver at the same time. Both types of collisions are illustrated in Fig. 2.3 (a) and 2.3 (b) respectively. In Fig. 2.3 (a), $N_1$ transmits data to $N_4$ while simultaneously $N_3$ transmits data to $N_4$, $N_5$ and $N_6$. This leads to a collision at $N_4$, since $N_4$ receives the transmission from $N_1$ as well as the transmission from $N_3$.
The limitation due to the half-duplex mode is illustrated in Fig. 2.3 (b), where $N3$ is transmitting data to $N_4$, $N_5$ and $N_6$, while simultaneously, $N_1$ is transmitting data to $N_3$. This leads to a collision at $N_3$, since $N_3$ is not capable of receiving and transmitting data at the same time.

Hence, a scheduler is required which avoids collisions. In order to avoid collisions in a wireless multihop network, a scheduler needs to separate nodes into subsets, where no interference or half-duplex constraint violation occur. An example is depicted in Fig. 2.4, where the wireless multihop network shown in Fig. 2.4 (a) is separated into three subgraphs as shown in Fig. 2.4 (b), (c) and (d) respectively.

However, to obtain an optimal collision-free scheduling is an NP-hard problem. Therefore, a discussion about the design, complexity and optimality of collision-free schedulers will be given in Sec. 3.2.4.



(a)

(b) (c) (d)

Figure 2.4: Separation of (a) wireless multihop network into three collision-free subsets (b), (c) and (d).

## 2.4 Physical Layer

### 2.4.1 Transmission Mechanisms

In this section, the PHY transmission mechanisms and the channel model are discussed. Throughout this thesis, the three PHY transmission mechanisms unicast, multicast and broadcast are considered. It is assumed that nodes are equipped with an omnidirectional antenna which is illustrated in Fig. 2.5, where the transmission range of nodes $N_1$ in green, $N_2$ in blue and $N_3$ in red are shown. Moreover, two nodes are connected if the intended receiver lies within the transmission range of the transmitter. This is shown in Fig. 2.6, where a dotted line between two nodes indicates a link and thus expresses that these two nodes can communicate with each other. However, the graph representation with unweighted links in Fig. 2.6 can only tell if a communication between nodes is possible. The amount of information and the direction of the information cannot be expressed by a graph representation with unweighted links. Therefore, a graph representation with weighted and directed links is more suitable. Such a representation is depicted in Fig. 2.7, where each link has a direction and a weight $c_{i,j}$. Here a link originates at the transmitter and ends at the receiver which is indicated by the arrow. Furthermore, the weight on each link expresses the link capacity, meaning the

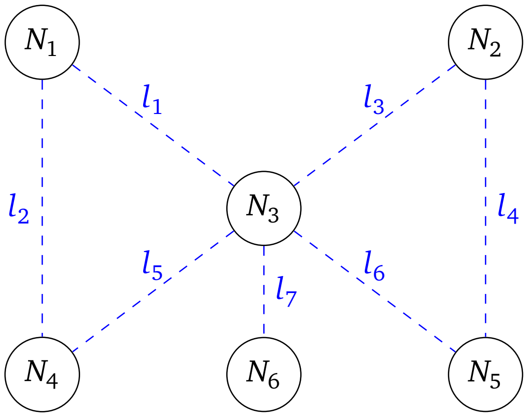


Figure 2.5: An example of omnidirectional transmissions of the three nodes $N_1$, $N_2$ and $N_3$.

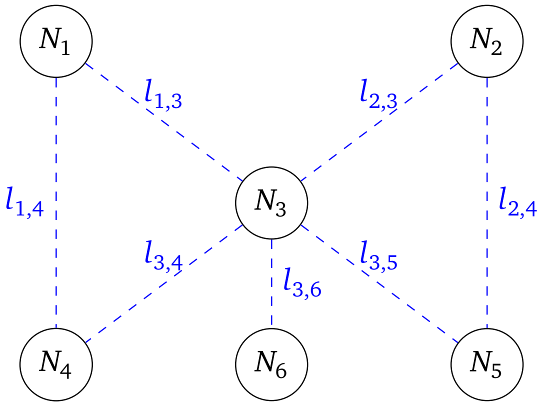


Figure 2.6: Graph with unweighted bidirectional links.

maximum amount of information that can be transmitted through the link. Based on the graph representation with weighted and directed links the PHY transmission mechanisms unicast, multicast and broadcast can be differentiated. A unicast transmission is utilized, when a node transmits a message to one specific neighbor node. This is achieved by adapting the rate to the link capacity of the intended neighbor node. As an example, $N_1$ wants to utilize a unicast transmission to $N_3$ for that the rate has to be set smaller or equal to $c_1$, but at least greater than $c_2$ under the assumption that $c_1 > c_2$. Hence, a unicast transmission can utilize available link capacities the most. However, transmitting to multiple neighbor nodes solely with unicast transmissions requires several time resources. In contrast, a broadcast transmission is utilized, when a node forwards messages to all its neighbor nodes, by adjusting the rate to the

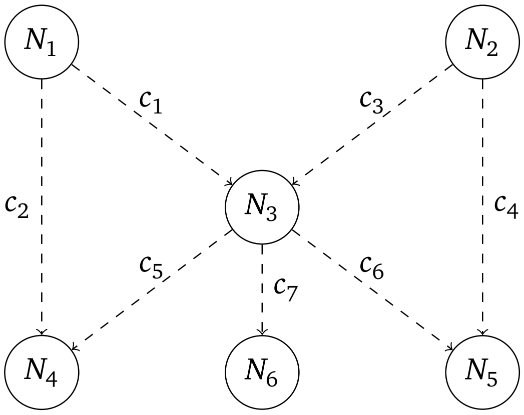


Figure 2.7: Graph with weighted directional links.

neighbor with the weakest link capacity. For an example, if $N_2$ wants to broadcast to all its neighbor nodes namely $N_3$ and $N_5$ it has to set the rate to the minimum of $c_3$ and $c_4$. Here, one advantage of the broadcast transmission is that all neighbor nodes receive a message after one transmission. However, the broadcast transmission always has to the node with weakest link capacity, which reduces the amount of information that can be transmitted. Another possibility is a multicast transmission, here a node selects a group neighbor nodes to which it forwards a message. One example, $N_3$ wants to communicate with $N_4$, $N_5$ and $N_6$, than $N_3$ has to set the rate to the minimum of the link capacities of $c_5$, $c_6$ and $c_7$. Therefore, multicast transmissions can be seen as a trade-off between unicast transmission and broadcast transmission, since a node can select neighbors with high link capacity and transmit the message within one time resource.

Regardless of the PHY transmission mechanisms, at all times the capacity of every link must be known. The link capacity between node $N_i$ and $N_j$ is determined based on the Shannon formula [Rap09]

$$c_l = \log_2(1 + SNR_j), \tag{2.1}$$

where $SNR_j$ is the signal to noise ratio measured at the receiving node $N_j$. The received signal to noise ratio $SNR_j$ is composed of the received power $P_j$ and the noise power $P_j^N$ at the receiving node $N_j$. Furthermore, the received power $P_j$ is composed of the transmit power $P_i$ at the $N_i$, the pathloss $PL_{N_j}$ and the channel gain $|h|^2$. Thus, a system which only makes use of one of the above PHY transmission mechanisms cannot adapt to changing network condition. Therefore, it is assumed that a node can switch between unicast, multicast and broadcast.

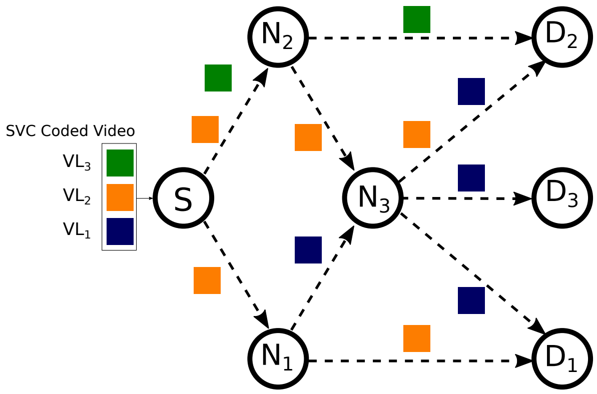


Figure 2.8: Source node $S$ transmits a SVC coded video to destination nodes $D_1$, $D_2$ and $D_3$.

## 2.5 Application Layer

### 2.5.1 Adaptive Video Streaming

In this section, we introduce the requirements and properties of adaptive video streaming at the APP. Video streaming in wireless multihop networks is a challenge due to the different capabilities of end-user devices and changing network conditions. Especially, the change in channel quality in wireless communication has a big impact. In the last decade, two methodologies for adaptive video streaming were proposed: one is adaptive bitrate streaming with non-scalable video coding [SSS19] and the other one is layered coded streaming [SMW07].

DASH with AVC can be categorized as an adaptive bitrate streaming which applies non-scalable video coding, where different representations with different bitrate requirements of the same video are stored. DASH with AVC can cope with diverse end-user device capabilities by choosing and switching between different representations of a video. RTP with SVC is an example for layered coded streaming, where a video is divided into one base layer and several enhancement layers. The enhancement layers are built upon the base layer, thus all layers have to be received in order to recover the video in the highest quality [SMW07]. Hence, if a user cannot meet the bitrate requirements of the base layer as well as all the enhancement layers, then the user receives the base layer and an appropriate subset of enhancement layers.

From a network-centric view, both DASH with AVC and RTP with SVC introduce multiple data streams, but they utilize the network resources differently. RTP with SVC introduces a data stream for every video layer, where a video layer with a higher order is distributed when all layers with a lower order have been received successfully.

Thus, multiple dependent data streams flow through the network to the users when RTP with SVC is utilized. In case of DASH with AVC, each user aims to obtain the video in the best quality with respect to his device capabilities. This may result in multiple independent data streams, where users are receiving different video representations of the same video. The multiple independent data streams introduced by DASH with AVC lead to a competition for the available resources in the network.

An example of adaptive video-streaming utilizing RTP with SVC is shown in Fig. 2.8, where source node $S$ transmits the video to the three destination nodes $D_1$, $D_2$ and $D_3$. The SVC-encoded is composed of three video layers, where $VL_1$ is the base video layer, $VL_2$ is the first enhancement video layer and $VL_3$ is the second enhancement video layer. As shown in Fig. 2.8 source node $S$ transmits the base layer $VL_1$ to the three destination nodes $D_1$, $D_2$ and $D_3$. The first enhancement layer $VL_2$ is received by $D_1$ and $D_2$, while the second enhancement layer $VL_3$ is received by $D_2$, as seen in Fig. 2.8. Hence, $D_3$ receives the video in the lowest quality, $D_1$ receives the video in medium quality and $D_2$ receives the video in high quality.

Also, DASH with AVC introduces multiple flows, however these flows are independent of each other. Hence, the amount of network resources required for DASH with AVC may be higher than for RTP with SVC, since the flows are dependent in RTP with SVC. Fig. 2.9 shows three examples, when DASH with AVC is applied. Here, source node $S$ has three video representations $VR_1$, $VR_2$ and $VR_3$ available to transmit the video to the three destinations $D_1$, $D_2$ and $D_3$. In Fig. 2.9 (a), all destination nodes receive the low quality video representation $VR_1$. Another example is depicted in Fig. 2.9 (b), where $D_1$ and $D_3$ receive the low quality video representation $VR_1$ and $D_2$ receives the medium quality video representation $VR_2$. In the third case shown in Fig. 2.9 (c), only $D_2$ receives video representation $VR_3$, which is the high quality video representation.

Both DASH with AVC and RTP with SVC can adapt to changing user capabilities and changing network conditions. However, both perform adaptation solely at the APP, which limits the level of adaptation with regards to changing network conditions. Therefore, to fully utilize the network resources and to fully adapt to changing network conditions, the lower three layers need to be taken into account.

### 2.5.2 Caching

In this section, we introduce the requirements and properties of caching at the APP, where a content, which might be split into parts, is distributed over a wireless multihop

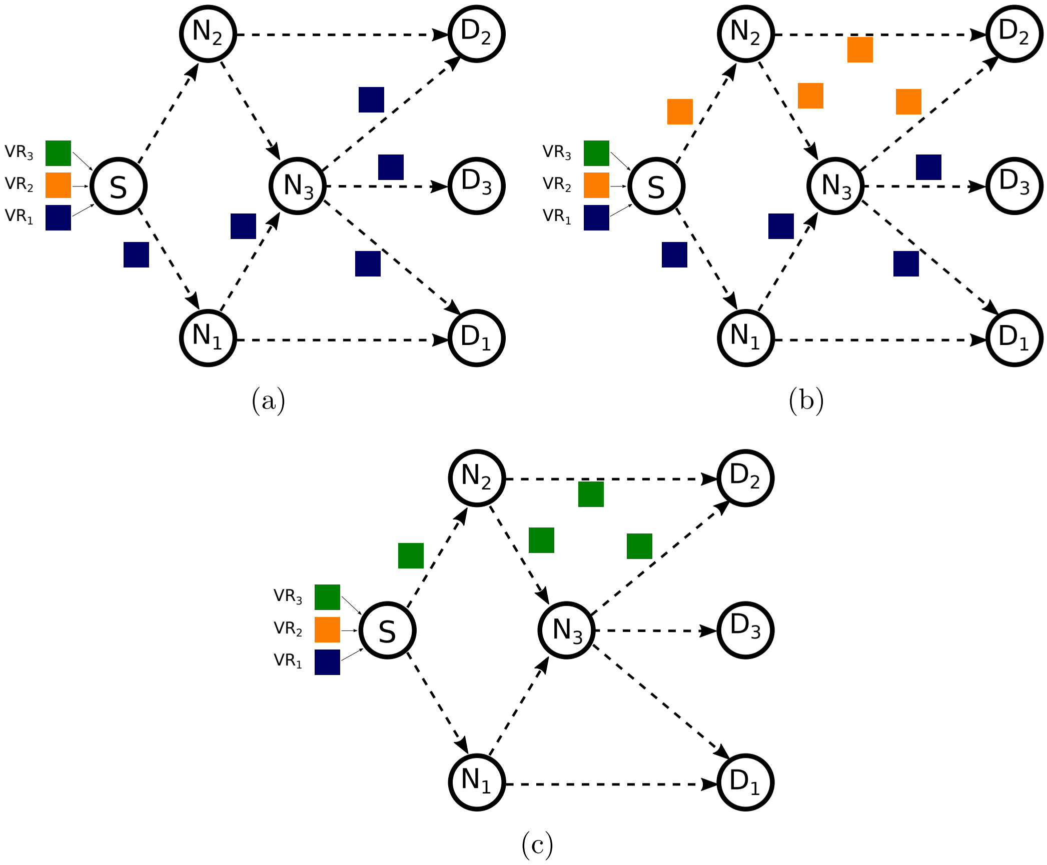


Figure 2.9: Source node $S$ transmits a AVC coded video via DASH to the destination nodes $D_1$, $D_2$ and $D_3$. In (a) $VR_1$ is received by all three destination nodes, (b) $VR_1$ is received by $D_1$ and $D_3$, while $VR_2$ is received by $D_2$ and (c) $VR_3$ is received only by $D_2$.

network. The content is simultaneously requested by multiple destinations within the network. Furthermore, it is assumed that the content is already cached in a distributed manner among nodes in the network, which requested the content previously. Hence, a destination may request missing parts of the content instead of the complete content. The scenario is illustrated exemplarily in Fig. 2.10, where the content $F$ is composed of four disjoint parts, denoted by $F_1, F_2, F_3, F_4$. Five nodes, $N_1, N_2, N_3, N_4, N_5$, have parts of the content cached and three destination nodes $D_1, D_2, D_3$ request the complete content $F = \{F_1, F_2, F_3, F_4\}$. As shown in Fig. 2.10, different nodes have different parts of the content, e.g. $N_2$ cached $F_1$ and $F_2$, while $N_4$ has $F_3$ and $F_4$ in the cache. Destinations may request any part of a content from any node which has the respective part of the content in its cache. However, a destination cannot request a part of the content from more than one node. In addition, each destination requires the complete content $F = \{F_1, F_2, F_3, F_4\}$. Therefore, the goal is to determine for each destination and each part of the content from which node that part of the content should be

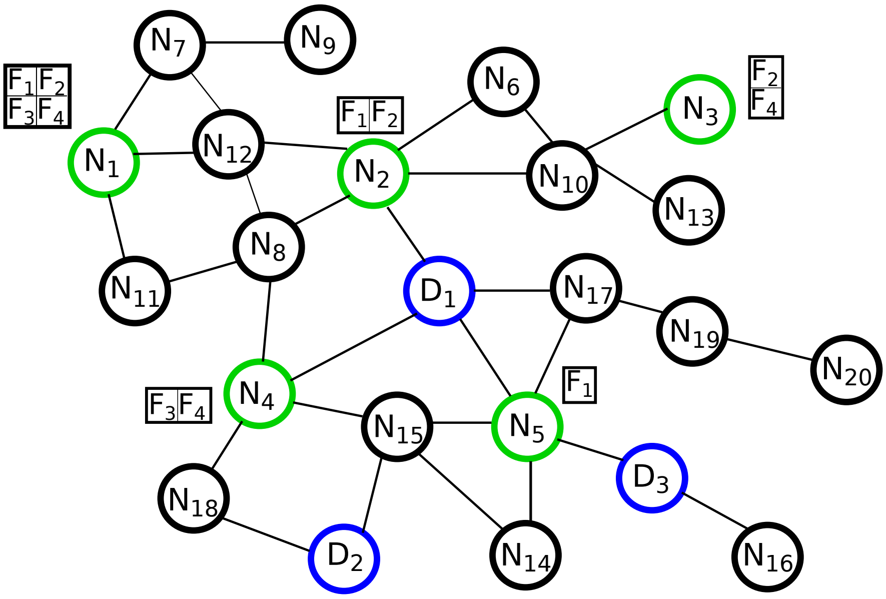


Figure 2.10: Destinations $D_1$, $D_2$ and $D_3$, marked in blue, request content $F$, which is distributed over the wireless multihop network, where nodes with the complete or part of the content in the cache are marked in green.

retrieved and through which paths it should be delivered. It has to be noted that these decisions are coupled and depend on the available paths, resources and channel qualities with respect to all destinations. The number of possible forwarders as well as the number of possible paths to the destinations grow when the number of nodes with cached content grow, which enlarges the complexity of the problem. Thus, a modeling is introduced in Chapter 6, which can cope with a high number of nodes with cached content. Furthermore, this model takes into account the possibility that nodes have cached only parts of the content.

# Chapter 3

# Transition-Enabled Cross-Layer Framework

## 3.1 Introduction

In this chapter, a transition-enabled cross-layer framework is formulated. The framework delivers content by taking into account the changing network conditions and by considering the mechanisms at the lower three layers. This allows the cross-layer framework to perform transitions, which is the switching between equivalent mechanisms on the same layer. Especially, cross-layer transitions at the lower three layers are utilized, to cope with changing network conditions. In more detail, transitions between network support structures at the NET and between PHY transmission mechanisms are considered. Consequently, a unified graph is developed which jointly models network support structures and PHY transmission mechanisms. Based on the unified graph, a collision-free scheduler at the MAC is proposed, which performs scheduling taking into account the available PHY transmission mechanisms. In the following, first the unified graph model integrating the lower three layers into one model is discussed in Sec. 3.2. Next, in Sec. 3.3 a linear optimization problem considering the unified graph model is formulated. In Sec. 3.4, the concept of transitions is introduced. Finally, in Sec. 3.5 the performance of the proposed transition-enabled cross-layer framework is compared with schemes which do not perform transitions.

## 3.2 Unified Graph-Model

### 3.2.1 Introduction

In this section, a unified graph model is introduced which aims to incorporate the capabilities and restrictions of the NET, MAC and PHY. The representation of wireless multihop networks as graphs is advantageous, since a graph reflects aspects of the lower three layers, e.g., an edge represents a wireless link between two nodes. However, a graph containing simple vertices and edges cannot convey all the aspects of the lower three layers, e.g., the broadcast nature of the wireless channel, Therefore, a unified graph model is introduced, which is based on the general graph model, where a graph $G$ is constructed based on a set of vertices $V$ and a set of edges $E$. This general graph model is expanded in the unified graph model by introducing virtual nodes and links to reflect the capabilities of the PHY transmission mechanisms. Additionally, scheduling constraints to incorporate the restriction at the MAC imposed by the wireless channel are considered, as well as, network support structures. The results of this section have been published by the author in [FKK14].

### 3.2.2 Network Support Structures

The utilization of different network support structures depend on constraints applied on the messages flowing through the network. The activation and deactivation of flow constraints is done in the optimization problem, which is discussed in Sec. 3.3. However, also the scheduling decisions at the MAC and the available PHY transmission mechanisms play a role if a tree structure or a butterfly structure should be utilized.

### 3.2.3 Transmission Mechanisms

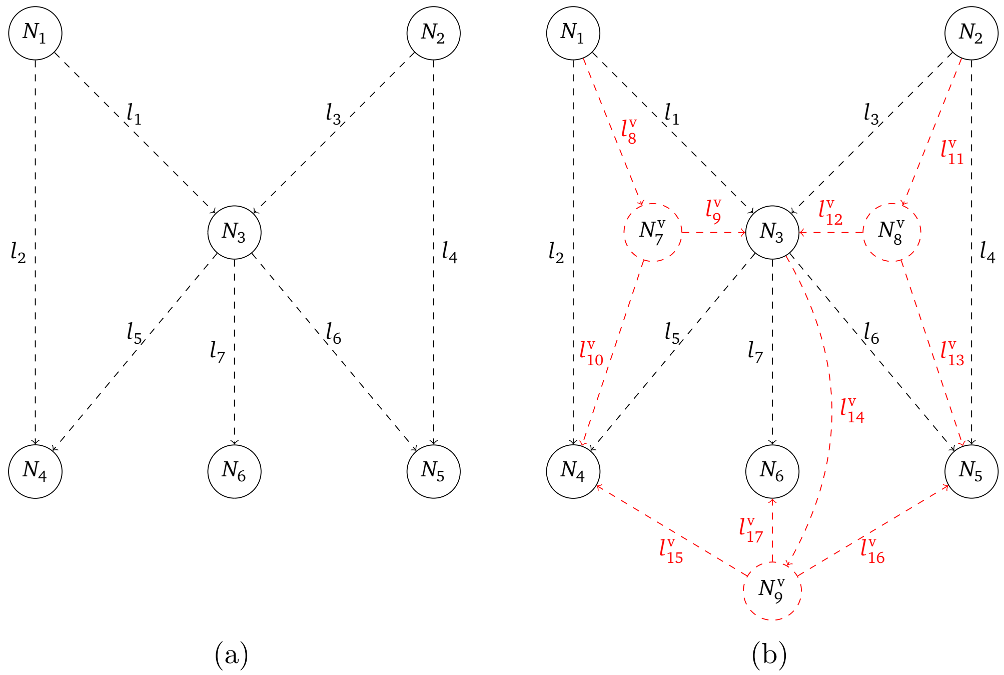


Figure 3.1: (a) An example network consisting of six nodes and eight links, (b) the resulting extended graph after applying node virtualization once.

As mentioned above, the PHY transmission mechanisms unicast, broadcast and multicast are considered. However, the general graph model, depicted in Fig. 3.1 (a), only takes unicast into account. Since all links in $\mathcal{L}$ represent a unicast transmission, where a link $l$ corresponds to the physical connection between node $N_i$ and node $N_j$ and the link capacity is expressed as $c_l$. For example, $N_1$ can perform a unicast transmission to $N_3$ with a rate equal to $c_1$ or can perform a unicast transmission to $N_4$ with a rate equal to $c_2$. However, the capability to utilize a broadcast transmission from $N_1$ to both $N_3$ and $N_4$ with a rate equal to the minimum of $c_1$ and $c_2$, which is not reflected in the graph depicted in Fig. 3.1 (a). Hence, the graph has to be extended, such that unicast transmissions, broadcast transmissions and multicast transmissions are represented in the graph. Therefore, the concept of node virtualization is applied to introduce broadcast transmissions and multicast transmissions. This results in an extended graph including three PHY transmission mechanisms unicast, broadcast and multicast which will be called the unified graph.

#### 3.2.3.1 Node Virtualization

The idea of node virtualization is based on the work in [YLYL06] and [NBK12], where virtual nodes and virtual links are introduced into the graph to represent broadcast communications. However, it should be noted that the aforementioned work was focusing solely on broadcast and graphs with equal capacities. In contrast, this thesis considers unequal link capacities. Furthermore, the virtualization concept is extended to also represent multicast.

Node virtualization is applied to each node with at least two outgoing edges, since only nodes with more than one neighbor node can utilize a broadcast transmission. For each node $N_i$ fulfilling this constraint, a virtual node $N_i^{\mathrm{v}}$ is created, which is then connected to the originating node $N_i$ and to the corresponding neighbor nodes $(N_i, N_j) \in \mathcal{L}$ via virtual links. The capacity of the virtual links is equal to the minimum capacity of the outgoing edges of $N_i$.

As an example, node virtualization is applied to the graph depicted in Fig. 3.1 (a), which results in the extended graph illustrated in Fig. 3.1 (b). In Fig. 3.1 (b), node virtualization is applied to nodes $N_1$, $N_2$ and $N_3$, since all three nodes have at least two outgoing edges. Hence, the virtual nodes $N_7^{\mathrm{v}}$, $N_8^{\mathrm{v}}$ and $N_9^{\mathrm{v}}$ are added and connected to $N_1$, $N_2$, $N_3$ and to the corresponding neighboring nodes of $N_1$, $N_2$ and $N_3$, respectively. The link capacities of the virtual links are equal to the minimum capacity of the unicast links. Therefore, the incoming and outgoing virtual links at $N_7^{\mathrm{v}}$ are equal to

$c^{\mathrm{v}}_8 = c^{\mathrm{v}}_9 = c^{\mathrm{v}}_{10} = \min\{c_1, c_2\}$. In the same manner, the virtual links of $N^{\mathrm{v}}_8$ and $N^{\mathrm{v}}_9$ are set respectively.

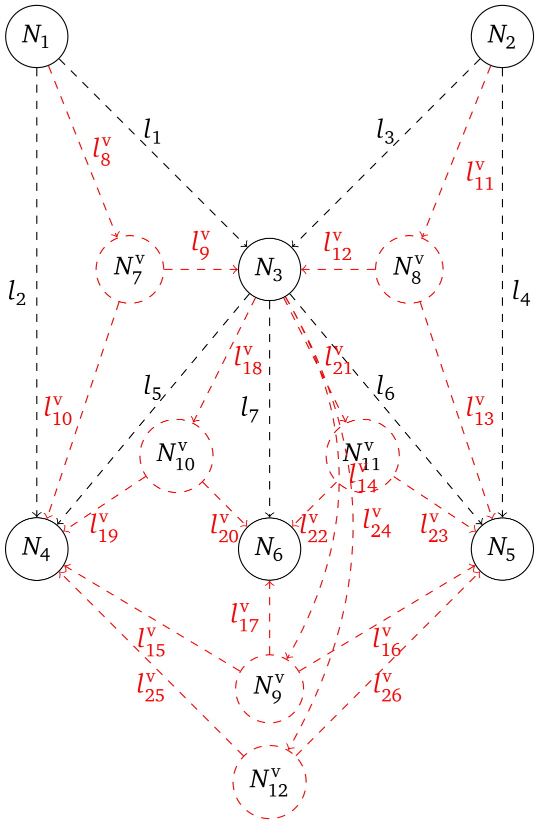


Figure 3.2: A unified graph with all PHY transmission mechanisms.

Based on the same principle, node virtualization can be applied to represent multicast, but instead of connecting a virtual node to all neighbor nodes, it is connected to a subset of neighbor nodes. A unified graph, fully utilizing unicast, broadcast and multicast is depicted in Fig. 3.2. This is achieved by applying node virtualization to all nodes in $\mathcal{G}$. Especially, performing node virtualization multiple times on node $N_3$ leads to the representation of the broadcast transmission and the three possible multicast transmissions.

Thus, the unified graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ includes the set of physical nodes $\mathcal{N}^{\mathrm{phy}} \subset \mathcal{N}$, the set of physical unicast links $\mathcal{L}^{\mathrm{phy}} \subset \mathcal{L}$, the set of virtual nodes $\mathcal{N}^{\mathrm{v}} \subset \mathcal{N}$ representing the multicast and broadcast transmission mechanisms, and the set of virtual links $\mathcal{L}^{\mathrm{v}} \subset \mathcal{L}$ to connect virtual nodes from $\mathcal{N}^{\mathrm{v}}$ with physical nodes of the network. For each node $N \in \mathcal{N}$ in the unified graph, we denote the sets of its outgoing links and its incoming links as $\mathcal{O}(N) \subset \mathcal{L}$ and $\mathcal{I}(N) \subset \mathcal{L}$, respectively. All links in $\mathcal{L}^{\mathrm{phy}}$ represent the PHY transmission mechanism unicast, where a link $l$ corresponds to the physical connection between two nodes and the link capacity is expressed as $c_l$, where all link capacities are normalized by the maximum link capacity.

Through node virtualization, all three PHY transmission mechanisms are present in the unified graph. Hence, a node can chose to either utilize a unicast by using one of the physical links in $\mathcal{L}^{\text{phy}}$ or to utilize a multicast or broadcast by activating the corresponding virtual nodes and the respective virtual links.

However, the unified graph, shown in Fig. 3.2, reveals a downside of node virtualization. By applying node virtualization, the size of the network increases and therefore the size of the problem increases, due to the addition of virtual nodes and links. The increase of the network can be calculated by counting the different possibilities a node can communicate to its neighbors. Based on the number of outgoing links $|\mathcal{O}(N_i)|$ of a node $N_i$ the number of all possible transmissions can be calculated as

$$2^{|\mathcal{O}(N_i)|} - 1. \tag{3.1}$$

However, only the possible multicast and broadcast transmissions impact the increase of the network. Hence, the number of unicast transmission need to be deducted from Eq. 3.1, which gives the number of possible multicast and broadcast transmissions as

$$2^{|\mathcal{O}(N_i)|} - (|\mathcal{O}(N_i)| + 1). \tag{3.2}$$

Based on Eq. 3.2, the total number of virtual nodes added to the network is given by

$$\sum_{\forall N_i \in \mathcal{N}} \left( 2^{|\mathcal{O}(N_i)|} - (|\mathcal{O}(N_i)| + 1) \right), \tag{3.3}$$

while the number of virtual links added to the network can be calculated by

$$\sum_{\forall N_i \in \mathcal{N}} \left( \left( 2^{|\mathcal{O}(N_i)|-1} \cdot |\mathcal{O}(N_i)| \right) - |\mathcal{O}(N_i)| \right) + \left( 2^{|\mathcal{O}(N_i)|} - (|\mathcal{O}(N_i)| + 1) \right). \tag{3.4}$$

Therefore, a quantization method for node virtualization is proposed in the next section, to find a trade-off between the utilization of the broadcast nature and the exponential increase of the problem.

#### 3.2.3.2 Quantization for Node Virtualization

In order to reduce the computational complexity introduced by the node virtualization, a quantization method is proposed. The idea is as follows, before the node virtualization is performed, the unicast link capacities are sorted and then ordered to a quantized capacity value. The advantage, instead of representing all possible multicast transmissions a subset of multicast transmissions is chosen. This reduces the number of virtual nodes and links added to the network and hence reduces the computational complexity of the problem. However, two aspects have to be considered. First, a rough quantization results in a lower number of node virtualizations and hence to a lower computational complexity, but leads to a great loss in capacity due to the restriction by the lowest observed capacity value. Second, a quantization with a narrow window length leads to high number of node virtualizations, thus to a higher complexity. But, the capacity loss will be small in comparison to a rough quantization and hence the solution will be closer to the optimum.

Therefore a quantization method is required, which reduces the number of virtual nodes and links while keeping multicast transmissions with high link capacities.

As a first step, each node generates $Q$ quantization groups which are generated based on the the minimum capacities $\min(c_l)$ and the maximum capacities $\max(c_l)$ of the unicast link capacities. The generated multicast groups have an equal width $W$, which is calculated as follows

$$W = \frac{\max_l(c_l) - \min_l(c_l)}{Q}, \tag{3.5}$$

In the second step, each node determines for each quantization group $Q_i$ the respective lower bound $c^{i,\min}$ and upper bound $c^{i,\max}$ and sorts the unicast link capacities in descending order. In the third step, all unicast links are grouped to the $Q$ quantization groups, where a unicast link can be grouped to more than one quantization group. This is due to the broadcast advantage which means that neighbors with strong links can overhear communications of other neighbor nodes. Hence, the size of the quantization group with the lowest lower bound has the highest number of unicast links. As a final step, for each quantization group $Q_i$ it is checked if the current lower bound $c^{i,\min}$ is the greatest lower bound, if that is not the case $c^{i,\min}$ is set to the minimum capacity within the quantization group $Q_i$.

In summary, quantization for node virtualization can reduce the computational complexity tremendously. Furthermore, it allows to consider all three PHY transmission mechanisms unicast, broacast and multicast, even in a network with high node density. However, the feasibility of considering unicast, broadcast and multicast, infers that the obtained solution may be sub-optimal.

### 3.2.4 Scheduling

The unified graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ includes both NET mechanisms and PHY transmission mechanisms. The next step is to design a collision-free scheduler, which allocates resources to nodes such that no collisions occur between them. At the same time, the number of required resources should be kept as low as possible. In the context of this thesis, a collision is present if one of the following conditions are meet. First, a collision occurs when a node is active as a transmitter and as a receiver, which is called the half-duplex constraint. Second, a collision occurs when a node receives more than one transmission at the same time, which is called the interference constraint. Hence, the idea is to split the unified graph $\mathcal{G}$ into $K$ collision-free subgraphs $\mathcal{G}_k = (\mathcal{N}_k, \mathcal{L}_k)$, where $K$ represents the number of time resources required by the network. In graph theory, the concept of coloring is very helpful to solve different types of scheduling problems. In the literature, there are two approaches edge coloring and vertex coloring, which are introduced next.

#### 3.2.4.1 Edge Coloring

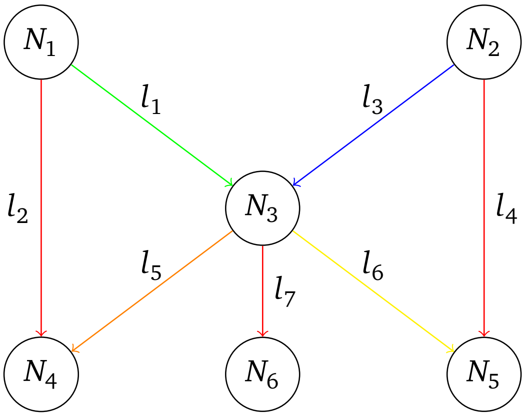


Figure 3.3: Edge coloring of a wireless multihop network with minimum number of colors.

The following definition is taken from [**AldousWilson2007**].

**Definition 1.** *Let $\mathcal{G}$ be a graph without loops. A **k-edge coloring** of $\mathcal{G}$ is an assignment of at most k colors to the edges of $\mathcal{G}$ in such a way that any two edges meeting at a vertex are assigned different colors. If $\mathcal{G}$ has a k-edge coloring, then $\mathcal{G}$ is **k-edge colorable**. The chromatic index of $\mathcal{G}$, denoted by $\chi^{'}(\mathcal{G})$, is the smallest number k for which $\mathcal{G}$ is k-edge colorable.*

An edge coloring is an assignment of $K$ colors to all edges in the graph, where two adjacent edges have different colors. The assignment of colors can be seen as the allocation of resources, where each color represents a resource. The resulting coloring is represented by forming the respective subsets from the set of links $\mathcal{L}$ into $K$ subsets, such that each subset $\mathcal{L}_k$ only contains edges with the same color $k$. An example for an edge coloring is depicted in Fig. 3.3, where the chromatic index $\chi^{'}(\mathcal{G})$ is 5. Hence, the resulting five subsets are as follows: $\mathcal{L}_1 = \{l_2, l_4, l_7\}$, $\mathcal{L}_2 = \{l_1\}$, $\mathcal{L}_3 = \{l_3\}$, $\mathcal{L}_4 = \{l_5\}$ and $\mathcal{L}_5 = \{l_6\}$. In general edge coloring can be used to implement a scheduler. However, applying edge coloring to design a scheduler for wireless networks, leads to collisions. This can be seen in Fig. 3.3, where although it is a proper edge coloring, a collision occurs.

The collision occurs when $N_3$ is activated as a transmitter, it will interfere with the communication between $N_1$ and $N_4$ as well as between $N_2$ and $N_5$.

This example illustrates that edge coloring does not cover the interference constraint, which is required for a collision-free scheduler. A collision-free scheduler would assign the link between $N_3$ and $N_6$ a new color leads to a coloring which is a proper coloring in terms of the half-duplex and interference constraint.

#### 3.2.4.2 Vertex Coloring

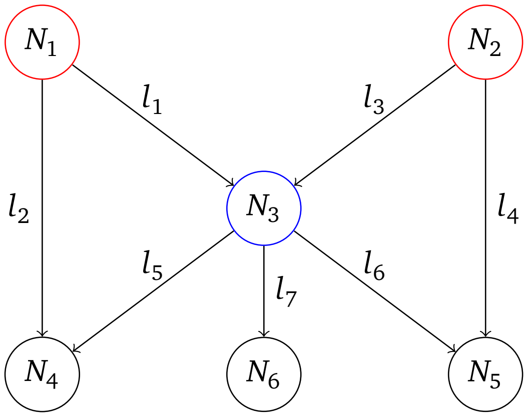


Figure 3.4: Vertex coloring with minimum number of colors.

Another coloring is vertex coloring, which is discussed next. First, the following definition is taken from [**AldousWilson2007**].

**Definition 2.** *Let $\mathcal{G}$ be a simple graph. A* ***k-coloring*** *of $\mathcal{G}$ is an assignment of at most k colors to the vertices of $\mathcal{G}$ in such a way that adjacent vertices are assigned different colors. If $\mathcal{G}$ has a k-coloring, then $\mathcal{G}$ is* ***k-colorable****. The* ***chromatic number*** *of $\mathcal{G}$, denoted by $\chi(\mathcal{G})$, is the smallest number for which $\mathcal{G}$ is k-colorable.*

Vertex coloring assigns colors to the vertices in a graph instead of the edges in a graph. A proper vertex coloring is given when no two adjacent vertices have the same color. The result of a vertex coloring can be expressed by separating the set of nodes $\mathcal{N}$. An example of a vertex coloring is depicted in Fig. 3.4. Here, the resulting subset are as follows, $\mathcal{N}_1 = \{N_1, N_2, N_6\}$, $\mathcal{N}_2 = \{N_3\}$ and $\mathcal{N}_3 = \{N_4, N_5\}$.

Again the problem arises that collisions can still occur, since $N_3$ could receive two transmissions simultaneously, one from $N_1$ and another from $N_2$. This would clearly violate the interference constraint that a node cannot receive more than one transmission at the same time. Thus, also vertex coloring is not applicable to design a collision-free scheduler.

Both edge and vertex coloring only cover the half-duplex constraint and do not cover the interference constraint. Since both edge and vertex coloring only check if edges or nodes are adjacent, respectively. This is to lax for the interference constraint, because it is crucial to identify if a node is active as a transmitter or as a receiver. Therefore, an extension based on edge and vertex coloring is proposed and discussed next.

#### 3.2.4.3 Extended Graph Coloring

In this section, a new collision-free scheduler based on an extended graph coloring is proposed. First, the non-adjacency rule of the general graph coloring approaches covers the half-duplex constraint. Thus, the extended graph coloring focuses on covering the interference constraint, additionally. The extension takes into account, that any transmission over the wireless medium is a broadcast transmission. Hence, in the extended graph coloring instead of coloring either vertices or edges, both vertices and edges are colored together.

In more detail, assigning a color to a node directly colors also the respective edges of the colored node. Thus indicating that also unintentional transmissions are activated. However, since activating unintentional transmissions does not directly mean interference an additional rule is required. This rules is as follows: Two non adjacent nodes

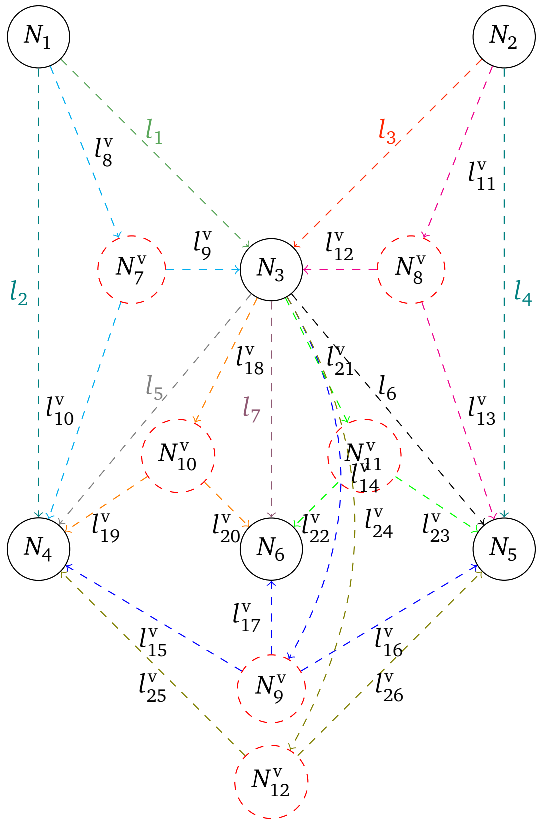


Figure 3.5: Coloring solution with the proposed collision-free scheduler.

$N_i$ and $N_j$ with a common neighbor $N_k$ are not interfering each other as long as $N_k$ is not active as a receiver.

Based on this extended graph coloring, a collision-free scheduler can be implemented. The proposed collision-free scheduler works with conventional graphs and with extended graphs which contains virtual nodes and links, but here the application of the collision-free scheduler on the extended graph is discussed.

As mentioned before, the scheduler takes into account the broadcast nature of the wireless medium. Hence, in case a node is scheduled, its corresponding virtual node and virtual links are colored in the same color. Since node virtualization may introduce multiple virtual nodes and virtual links for a node, only the node virtualization representing the broadcast transmission is considered. This ensures that the broadcast nature is taken into account and thus ensuring that the interference constraint is not violated.

Since all three PHY transmissions are considered, different scheduling combinations are possible. Determining all scheduling combinations is computationally expensive and instead the collision-free scheduler finds combinations where every link is active and every node is active as a transmitter and receiver at least once. Thus, the proposed collision-free scheduler does not guarantee an optimal solution. Nevertheless, the scheduler aims to find the most efficient combinations by operating as follows: A coloring performed with the proposed collision-free scheduler is illustrated in Fig. 3.5, where $K = 12$. The resulting subgraphs are $\mathcal{G}_1 = \{\{N_3, N_9^{\mathrm{v}}\}, \{l_{14}^{\mathrm{v}}, l_{15}^{\mathrm{v}}, l_{16}^{\mathrm{v}}, l_{17}^{\mathrm{v}}\}\}$, $\mathcal{G}_2 = \{\{N_3, N_{10}^{\mathrm{v}}\}, \{l_{18}^{\mathrm{v}}, l_{19}^{\mathrm{v}}, l_{20}^{\mathrm{v}}\}\}$, $\mathcal{G}_3 = \{\{N_3, N_{11}^{\mathrm{v}}\}, \{l_{21}^{\mathrm{v}}, l_{22}^{\mathrm{v}}, l_{23}^{\mathrm{v}}\}\}$, $\mathcal{G}_4 = \{\{N_3, N_{12}^{\mathrm{v}}\}, \{l_{24}^{\mathrm{v}}, l_{25}^{\mathrm{v}}, l_{26}^{\mathrm{v}}\}\}$, $\mathcal{G}_5 = \{\{N_3\}, \{l_5\}\}$, $\mathcal{G}_6 = \{\{N_3\}, \{l_6\}\}$, $\mathcal{G}_7 = \{\{N_3\}, \{l_7\}\}$, $\mathcal{G}_8 = \{\{N_1, N_7^{\mathrm{v}}\}, \{l_8^{\mathrm{v}}\}, l_9^{\mathrm{v}}, l_{10}^{\mathrm{v}}\}\}$, $\mathcal{G}_9 = \{\{N_1, N_2\}, \{l_2, l_4\}\}$, $\mathcal{G}_{10} = \{\{N_1\}, \{l_1\}\}$, $\mathcal{G}_{11} = \{\{N_2, N_8^{\mathrm{v}}\}, \{l_{11}^{\mathrm{v}}, l_{12}^{\mathrm{v}}, l_{13}^{\mathrm{v}}\}\}$ and $\mathcal{G}_{12} = \{\{N_2\}, \{l_3\}\}$.

**Algorithm 1** Algorithm of the collision-free scheduler

```
procedure GetMultiEdgeColoring
    Calculate the degree of every node $deg(N_i)$
    Queue all links in descending order of their respective node degree
    Initialize an empty set $\mathcal{G}_k = (\mathcal{N}_k, \mathcal{L}_k)$ with $\mathcal{N}_k = \emptyset$ and $\mathcal{L}_k = \emptyset$
    Add first node from the queue to $\mathcal{G}_k$
    Iterate through remaining queue
    Check for each node if the half duplex and the interference constraints are violated
    If a constraint is violated go to next node
    Else add node to $\mathcal{G}_k$
    remove node from queue
    Repeat steps until queue is empty
```

## 3.3 Single-Rate Multi-Source Optimization Problem

In this section, an optimization problem is formulated, where multiple source nodes aim to serve multiple destination nodes. Furthermore, each source node serves its respective destinations with an equal rate and thus each source node has to adjust to the destination node with lowest achievable rate. This single-rate multi-source optimization problem takes into account the NET, MAC and PHY. The overall objective is to allocate the resources efficiently in the system, while each source tries to serve all its destination nodes. Therefore, the objective function is formulated in Eq. (3.6), where $r_s$ is the rate between source node $S_s$ to each of its respective destination nodes. The objective function in Eq. (3.6) maximizes the overall rate in the system, while each source serves every destination with the same rate.

$$\max \sum_{s=1}^{S} (r_s). \tag{3.6}$$

The flow from source node $S_s$ to destination node $D_d$ over link $l$ in the $k$-th subgraph is defined as $f_l^{(k)}(s,d)$. The outgoing flow from source node $S_s$ to destination node $D_d$ over the outgoing links $\mathcal{O}(S_s)$ is $r_s$. Hence, the flow constraint for each source is expressed in Eq. (3.7).

$$\sum_{k=1}^{K} \left( \sum_{l \in \mathcal{O}(S_s)} f_l^{(k)}(s,d) \right) = r_s, \quad \text{for} \quad s = \{1,...,S\}, d = \{1,...,D\}. \tag{3.7}$$

The flow conservation constraint for destination nodes is given in Eq. (3.8).

$$\sum_{k=1}^{K} \left( - \sum_{l \in \mathcal{I}(D_d)} f_l^{(k)}(s,d) \right) = -r_s, \quad \text{for} \quad s = \{1,...,S\}, d = \{1,...,D\}. \tag{3.8}$$

Moreover, every node in $\mathcal{N}^{\text{phy}} \cup \mathcal{N}^{\text{v}}$ must satisfy the flow conservation, which conveys that any incoming flow into a forwarding node must depart from the forwarding node. Thus, the flow conservation constraint for forwarding nodes is given in Eq. (3.9).

$$\sum_{k=1}^{K} \left( \sum_{l \in \mathcal{O}(N)} f_l^{(k)}(s,d) - \sum_{l \in \mathcal{I}(N)} f_l^{(k)}(s,d) \right) = 0, \quad \text{for} \quad s = \{1,...,S\}, d = \{1,...,D\}, \forall N \in (\mathcal{N}^{\text{phy}} \cup \mathcal{N}^{\text{v}}) \setminus \{D_d\}. \tag{3.9}$$

Furthermore, every flow in the network is upper bounded by a capacity constraint. The capacity of a link $l$ in subgraph $\mathcal{G}_k$ depends on the link capacity $c_l$ and the duration the link is utilized in the $k$-th subgraph, which is determined by the timeshare factor $\tau_k$. If a link is part of subgraph $\mathcal{G}_k$, the indicator function $\mathbf{I}_{\mathcal{L}_k}(l)$ is one, else the flow on that link is zero in $\mathcal{G}_k$. The indicator function is written as

$$\mathbf{I}_{\mathcal{L}_k}(l) = \begin{cases} 1, & \text{if } l \in \mathcal{L}_k, \\ 0, & \text{otherwise.} \end{cases} \tag{3.10}$$

All flows over a link have to satisfy the capacity constraint, which is expressed in Eq. (3.11).

$$\sum_{k=1}^{K} f_l^{(k)}(s,d) \leq \tau_k \cdot c_l \cdot \mathbf{I}_{\mathcal{L}_k}(l), \quad \text{for} \quad k = \{1,...,K\}, s = \{1,...,S\}, d = \{1,...,D\}, \forall l \in \mathcal{L}_k. \tag{3.11}$$

Each flow in the network is positive, which is formulated in Eq. (3.12).

$$f_l^{(k)}(s,d) \geq 0, \qquad (3.12)$$
$$\text{for} \quad k = \{1,...,K\}, s = \{1,...,S\}, d = \{1,...,D\}, \forall l \in \mathcal{L}_k.$$

Furthermore, the timeshare factor $\tau_k$ is the relative measure of time allocated to the $k$-th subgraph, which is expressed in Eq. (3.13).

$$\sum_{k=1}^{K} \tau_k = 1, \quad k = \{1,...,K\}. \qquad (3.13)$$

Finally, each timeshare factor is bounded between zero and one, which is formulated in Eq. (3.14).

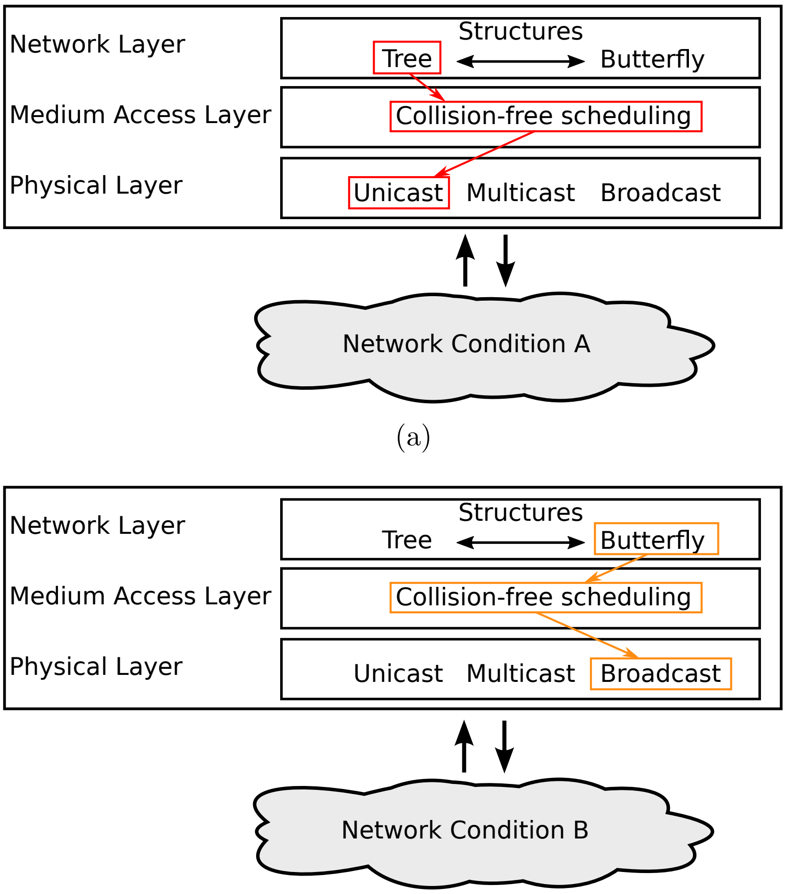
$$0 \leq \tau_k \leq 1, \quad k = \{1,...,K\}. \qquad (3.14)$$

Thus, the sum rate maximization problem for multiple sources and multiple destinations is expressed in Eq. (3.6) - Eq. (3.14).

## 3.4 Transitions in the Cross-Layer Framework

The transition-enabled cross-layer framework performs transitions at the NET and at the PHY. At the NET, the transition is performed between tree structures and butterfly structures. In a butterfly structure, network coding can be utilized, while in a tree structure, a simple store and forward is applied. At the PHY, transitions are performed between unicast, broadcast and multicast. Depending on the network conditions, the transitions-enabled cross-layer framework chooses the optimal combination at the lower three layers and as soon as the network conditions change, a transition is performed by finding a new optimal combination.

This is exemplified in Fig. 3.6, where in Fig. 3.6 (a) for network condition A, a combination of tree at the NET, collision-free scheduler at the MAC and unicast at the PHY is chosen. If the network conditions change, the combination of mechanisms may change, as for example in Fig. 3.6 (b), where for network condition B, a combination of butterfly at the NET, collision-free scheduler at the MAC and broadcast at the PHY is chosen. Thus, the transition-enabled cross-layer framework determines the optimal combination at the lower three layers for a given network condition and as soon as the network conditions change, it determines a new optimal combination. The capability to perform transitions at the PHY is due to the integration of all three PHY transmission



(a)

(b)

Figure 3.6: (a) For network condition A, tree is chosen at the NET and unicast at the PHY, (b) for network condition B, butterfly is chosen at the NET and broadcast at the PHY

mechanisms unicast, broadcast and multicast into the unified graph. The capability to perform transitions at the NET is due to the relaxation of the flow constraint in the optimization problem, where flows originating from the same source do not compete for resources. Moreover, the proposed collision-free scheduler at the MAC performs a coloring on the unified graph such that resource utilization is maximized while avoiding collisions occurring due to the broadcast nature.

| | NET mechanisms | | PHY mechanisms | | |
|---|---|---|---|---|---|
| | Butterfly | Tree | Unicast | Broadcast | Multicast |
| TUC | ✓ | - | ✓ | - | - |
| BBC | ✓ | ✓ | - | ✓ | - |
| PHY$_x$ | - | ✓ | ✓ | ✓ | - |
| TSOR | ✓ | ✓ | ✓ | ✓ | ✓ |

Table 3.1: Different schemes and their respective configurations.

## 3.5 Performance Analysis

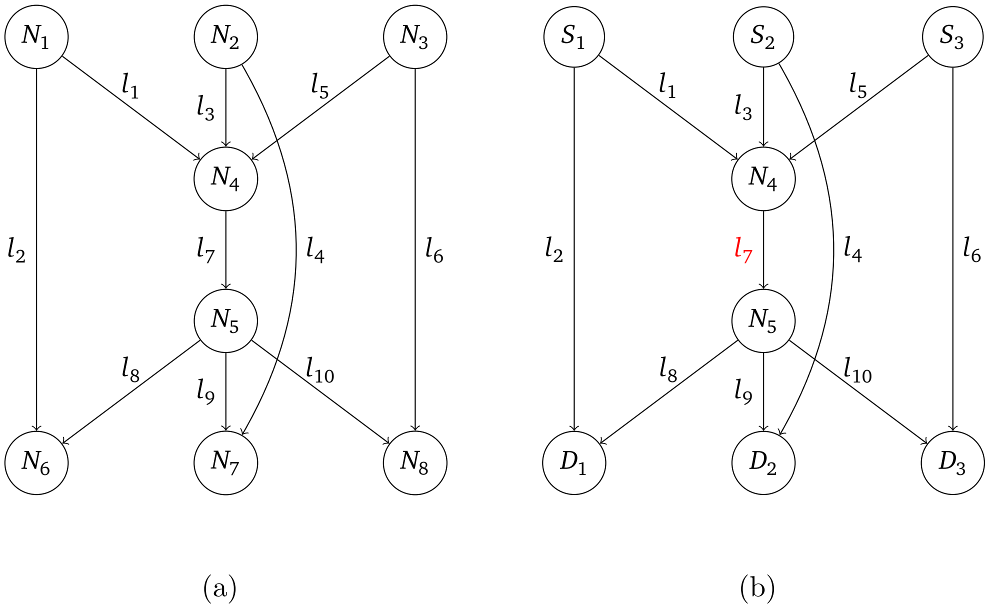
### 3.5.1 Introduction

The performance of the transition-enabled cross-layer framework is evaluated by first investigating static wireless multihop networks, where the number of nodes and links are fixed. In a second step, the performance of the transition-enabled cross-layer framework is evaluated by performing Monte-Carlo simulations. The performance is determined by the sum rate achieved in the network, which is the sum of rates between all source destination pairs. Furthermore, if not stated otherwise the utility function maximizes the sum rate in the network, while each source serves its respective destinations with the same rate. Furthermore, the performance of the transition-enabled cross-layer framework is compared with different schemes. The reference schemes can be broadly divided into two groups, the first group cannot perform transitions, while the second group can perform transitions, but only at a single layer. This is summarized in more detail in Table 3.1. The first scheme utilizes tree network support structures and unicast transmissions which is abbreviated TUC. The second scheme utilizes both butterfly and tree network support structures and broadcast transmissions which is abbreviated BBC. The third scheme utilizes tree at the NET and performs transitions between the PHY transmission mechanisms unicast and broadcast which is abbreviated PHY$_x$. Finally, the proposed transition-enabled cross-layer framework utilizes tree and butterfly at the NET, while it utilizes unicast, broadcast and multicast transmissions at the PHY, which is abbreviated TSOR.

### 3.5.2 Transitions in Fixed Wireless Multihop Networks

In this section, the schemes summarized in Table 3.1 are evaluated by studying their respective performances in a fixed wireless multihop networks. The fixed wireless multihop network is depicted in Fig. 3.7 (a) and consists of eight nodes, where $N_1$, $N_2$

and $N_3$ are source nodes, $N_4$ and $N_5$ are relay nodes and $N_6$, $N_7$ and $N_8$ are destination nodes, respectively. Each source node $S_1$, $S_2$ and $S_3$ aim to transmit data to the destination nodes $D_1$, $D_2$ and $D_3$. Moreover, all links in the network have a fixed capacity of one, which is shown in Fig. 3.7 (b). In order to observe a change in the network, one link is chosen as a bottleneck link, where the link capacity of the bottleneck link is tuned between a range from one to zero. As an example, in Fig. 3.7 (b) the link $l_7$ is marked as a bottleneck link. Since the location of the bottleneck plays a role different links are chosen and evaluated. Since the network shown in Fig. 3.7 (a) is symmetric and only one bottleneck link is present at a time, four bottleneck cases are identified and investigated. The first bottleneck case is when the bottleneck link is located between a source node and one destination node. Thus, when either the link between $S_1$ and $D_1$, $S_2$ and $D_2$ or $S_3$ and $D_3$ is the bottleneck link. The second bottleneck case is given when the bottleneck is located between a source node and the relay node $N_4$, which is either the link $l_1$, $l_3$ or $l_5$. The third bottleneck case is when the bottleneck is between the two relay nodes $N_4$ and $N_5$, which is an essential link in the network. The final bottleneck case is when the bottleneck link is present between the relay node $N_5$ and one of three different destination nodes $D_1$, $D_2$, or $D_3$. The result for the first case is illustrated in Fig. 3.8, where the sum rate, measured in bits/s/Hz, is plotted against the link capacity of the bottleneck link. For this case, the scheme TUC has the lowest performance, the scheme PHY$_x$ has the second best performance, while



(a) (b)

Figure 3.7: (a) The fixed wireless network with eight nodes and ten links and (b) the fixed wireless network with three sources, two relays, three destinations and one bottleneck link ($l_7$).

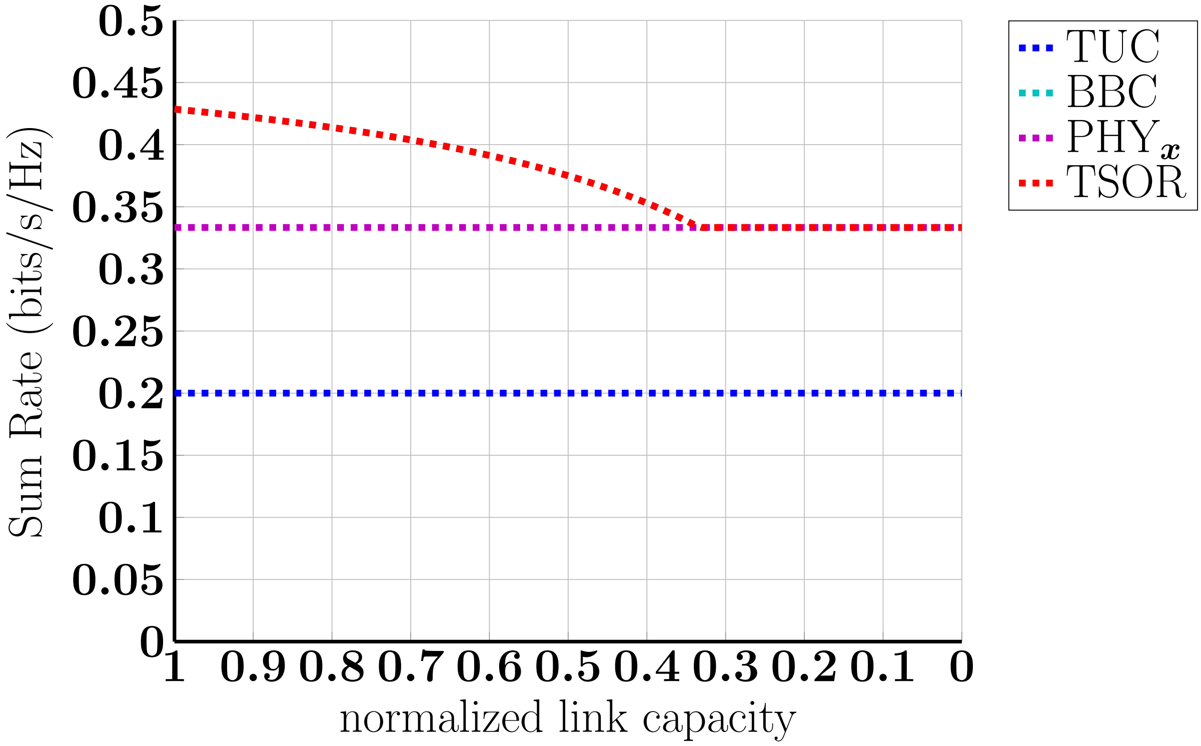


Figure 3.8: Simulation Result: Sum rate over normalized link capacity, the bottleneck is located between $S_1$ and $D_1$.

both BBC and TSOR have the best performance. In more detail, the scheme TUC achieves a constant sum rate of 0.2 bits/s/Hz. The reason for the low sum rate is that TUC utilizes only unicast transmissions, which require more transmissions compared broadcast transmissions. On the other hand, TUC achieves a constant sum rate of 0.2 bits/s/Hz regardless of the current link capacity of the bottleneck link. This can be explained by looking at the impact of the bottleneck link. The bottleneck link does not have an impact on the solution obtained by TUC, since all source nodes can serve the three destination nodes through the path of the two relay nodes $N_4$ and $N_5$. Hence, each source node serves its destination nodes with rate of 0.0667 bits/s/Hz and thus achieve in sum a rate of 0.2 bits/s/Hz.

The scheme PHY$_x$ achieves a constant sum rate of 0.33 bits/s/Hz, which is a gain of 65 % compared to TUC. PHY$_x$ achieves a higher sum rate compared to TUC since PHY$_x$ utilizes broadcast transmissions. Moreover, the constant sum rate can be explained with the same logic applied for TUC. Here, each source node utilizes a unicast transmission to the relay node $N_4$, which forwards the data to relay node $N_5$ and finally relay node $N_5$ utilizes broadcast transmissions to forward the data from each source. This leads to a rate of 0.111 bits/s/Hz for each source node and to a sum rate of 0.33 bits/s/Hz.

The scheme BBC achieves a sum rate equal to 0.429 bits/s/Hz, when the link capacity of the bottleneck is one. As the link capacity of the bottleneck link decreases, the sum rate decreases from 0.429 bits/s/Hz to 0.33 bits/s/Hz. In the range from 1 to 0.33 BBC utilizes network coding and broadcast transmissions, while in the range of 0.33 and 0 BBC switches to a tree structure at the NET. Moreover, in addition to switching the NET mechanism also the source node with the bottleneck link stops serving the destination nodes $D_1$, $D_2$ and $D_3$.

Also, TSOR achieves the highest sum rate, which is 0.429 bits/s/Hz and thus two times higher compared to TUC and a gain of 30 % compared to PHY$_x$. In the range from 1 to 0.33 both TSOR utilizes network coding and broadcast transmissions. But also here as the link capacity of the bottleneck decreases from 1 to 0.33, the sum rate decreases from 0.429 bits/s/Hz to 0.33 bits/s/Hz. At a link capacity of 0.33, TSOR switches from a butterfly structure to a tree structure at the NET. However, in contrast to BBC it does not switch off the source node with the bottleneck link, but instead the source node switches from a broadcast transmission to a unicast transmission. The results of the second bottleneck case are shown in Fig. 3.9. As before, TUC achieves a constant sum rate of 0.2 bits/s/Hz, which is the lowest sum rate compared to the other

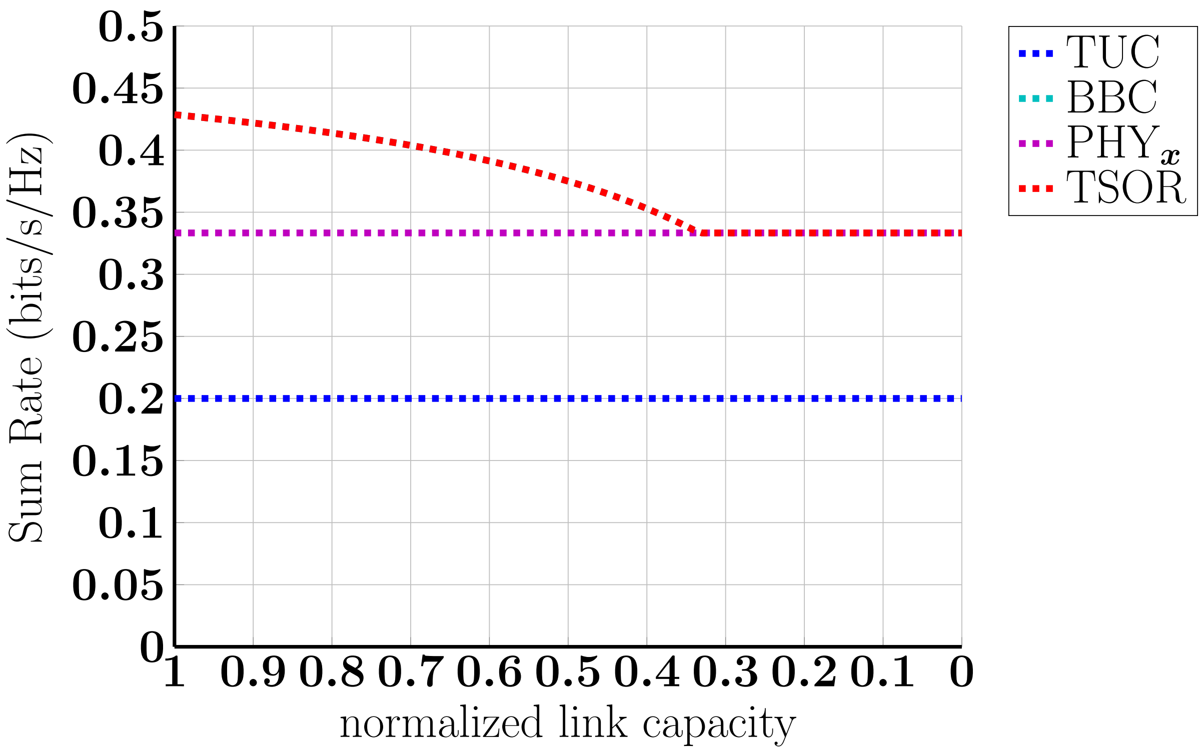


Figure 3.9: Simulation Result: Sum rate over normalized link capacity, the bottleneck is located between $S_1$ and $N_4$.

schemes. In contrast to the previous bottleneck case, not all source nodes serve the three destination nodes $D_1$, $D_2$ and $D_3$. However, the sum rate stays constant which can be explained by determining the maximum achievable rate in the network. This is given when only one source node serves the destination nodes $D_1$, $D_2$ and $D_3$. The maximum achievable rate with TUC is 0.2 bits/s/Hz. Thus, when the link capacity of the bottleneck link between $S_1$ and $D_1$ goes to zero, also the rate between $S_1$ and the destination nodes $D_1$, $D_2$ and $D_3$ goes to zero, while the rate of source nodes $S_2$ and $S_3$ increase from 0.0667 bits/s/Hz to 0.1 bits/s/Hz.

Since the performance of $\text{PHY}_x$ is the same as for the first case, one assumes the same logic can be applied as before. However, although $\text{PHY}_x$ achieves a constant sum rate of 0.33 bits/s/Hz over the complete range of the link capacity of the bottleneck link between $S_1$ and $N_4$, the constant performance does come from switching off the source node $S_1$. Moreover, $\text{PHY}_x$ the maximum achievable rate with $\text{PHY}_x$ is 0.33 bits/s/Hz, since any source node without the respective bottleneck can reach all destination nodes by utilizing three broadcast transmissions.

Also, BBC achieves the same results as in the first bottleneck case, where BBC at first achieves a sum rate of 0.429 bits/s/Hz which declines as the link capacity of the bottleneck link decreases. As in the first case, BBC utilizes network coding at the NET and broadcast transmissions at the PHY and as the link capacity of the bottleneck link decreases, $S_2$ and $S_3$ serves $D_1$, $D_2$ and $D_3$ by utilizing a tree structure and broadcast transmissions at the PHY. Thus, BBC performs in the same way in the second bottleneck case as in the first bottleneck case.

TSOR achieves at the highest a sum rate equal to 0.429 bits/s/Hz which drops as the link capacity of the bottleneck link decreases until it reaches it lowest value equal to 0.33 bits/s/Hz and stays constant for the remaining values of the link capacity at the bottleneck link. In the range from 1 to 0.33, TSOR utilizes network coding at the NET and broadcast transmissions at the PHY. In the second half from 0.33 to 0, TSOR switches to a tree structure at the NET and at $S_1$ to unicast transmissions at the PHY. Hence, TSOR achieves the same sum rate as BBC but instead of switching of a source node it performs a transitions. The third bottleneck case is shown in Fig. 3.10, where the bottleneck link lies between $N_4$ and $N_5$. Here, all four schemes have a downward trend and the sum rate drops to zero as the link capacity of the bottleneck link goes to zero. The reason is that the link between $N_4$ and $N_5$ is an essential link to serve the three destination nodes.

In more detail, at a link capacity of one TUC achieves a sum rate of 0.2 bits/s/Hz, but rapidly decreases until the sum rate reaches zero. The scheme $\text{PHY}_x$ achieves a higher

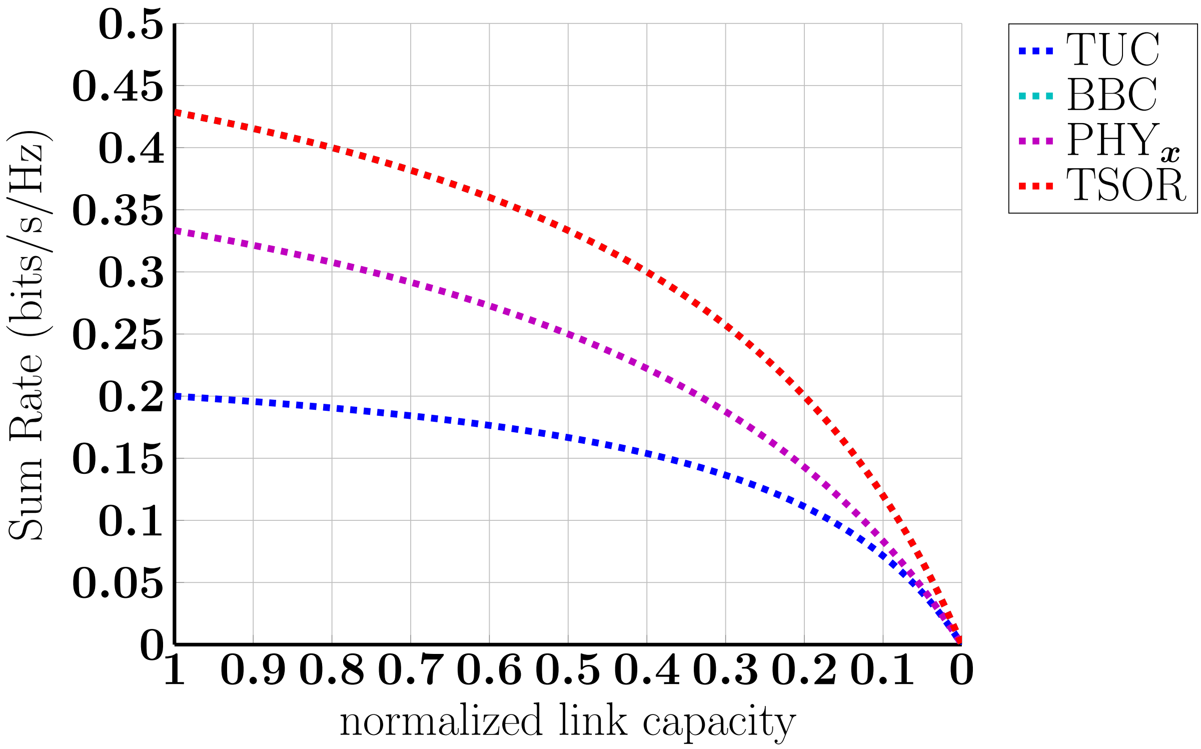


Figure 3.10: Simulation Result: Sum rate over normalized link capacity, the bottleneck is located between $N_4$ and $N_5$.

sum rate of 0.33 bits/s/Hz compared to TUC, but also here the sum rate declines as the link capacity of the bottleneck decreases.

Both BBC and TSOR start with a sum rate of 0.429 bits/s/Hz, but both schemes cannot compensate the decline of the link capacity at the bottleneck.

This bottleneck case shows that there are cases where performing transitions can be utilize to achieve an improvement or a stabilization in the network. The final bottleneck case is depicted in Fig. 3.11, where the bottleneck is located between $N_5$ and $D_1$. For this case, TUC achieves again a constant sum rate of 0.2 bits/s/Hz. TUC accomplishes this performance by deactivating the source nodes $S_2$ and $S_3$, since both source nodes are affected by the bottleneck link. At the same time, the source node $S_1$ can utilize unicast transmission directly to destination node $D_1$, while serving $D_2$ and $D_3$ through unicast transmissions performed by relay nodes $N_4$ and $N_5$.

PHY$_x$ achieves at first a sum rate of 0.33 bits/s/Hz, where all source nodes serve all destination nodes. But as the link capacity decreases the sum rate decreases. As soon as the link capacity goes below 0.5, PHY$_x$ switches off both $S_2$ and $S_3$. Instead,

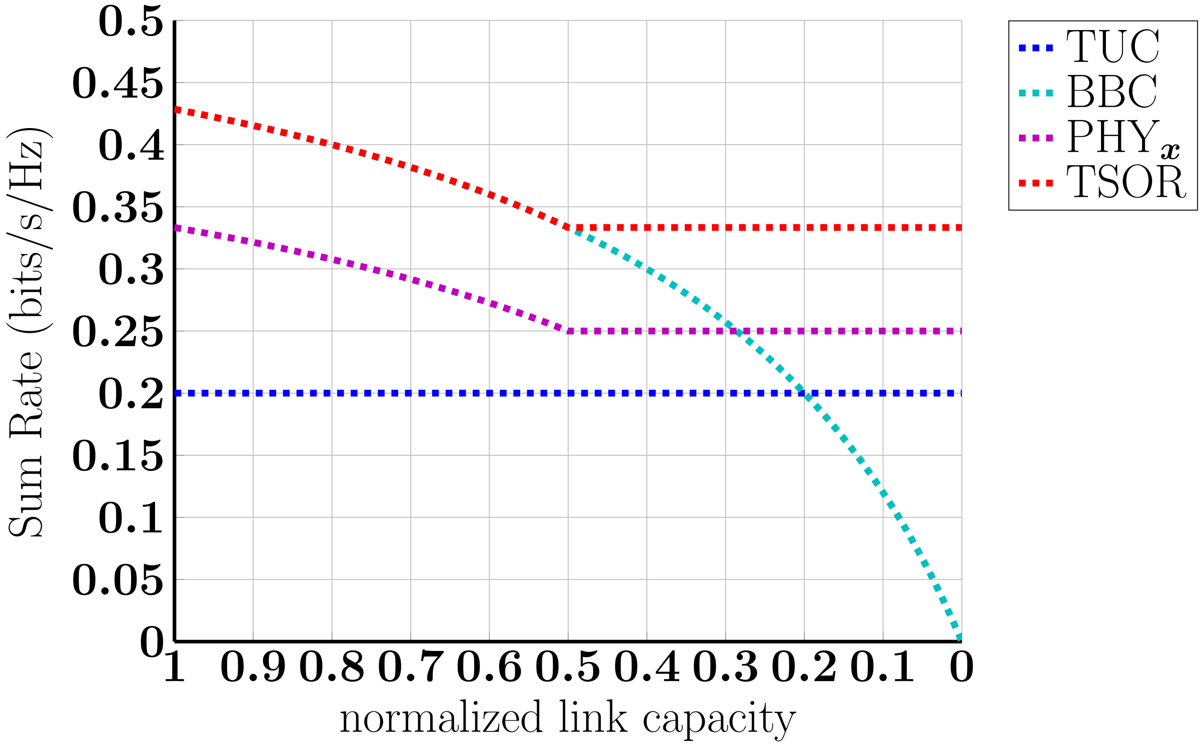


Figure 3.11: Simulation Result: Sum rate over normalized link capacity, the bottleneck is located between $N_5$ and $D_1$.

only source node $S_1$ serves all three destination nodes by utilizing both unicast and broadcast transmissions, which leads to a constant sum rate of 0.25 bits/s/Hz.

In contrast, BBC achieves a sum rate of 0.429 bits/s/Hz, but the sum rate declines with each decrease of the link capacity of the bottleneck until it reaches zero. This is due to the fact that the bottleneck between $N_5$ and $D_1$ is crucial bottleneck, since BBC can only utilize broadcast transmissions at the PHY.

The overall highest performance is achieved by TSOR, which utilizes all available mechanisms and transitions. As before, TSOR utilizes network coding at the NET and broadcast transmissions at the PHY, when the link capacity lies between one and 0.5. As soon as the link capacity goes below 0.5, TSOR switches to a tree structure and deactivates $S_2$ and $S_3$ as the other schemes. However, TSOR utilizes unicast, broadcast and multicast transmissions at the PHY, to achieve a sum rate of 0.33 bits/s/Hz.

The evaluation in the fixed wireless multihop networks provides an important conclusion. Transitions can provide significant gains, as long as a certain amount of equivalent mechanisms as well as the presence of alternative resources, e.g., paths, are available.

# Chapter 4

# Adaptive Video Content Distribution using DASH with AVC

## 4.1 Introduction

In this chapter, the possibilities of adaptive video content distribution in wireless multihop networks are examined. In more detail, the capabilities of DASH with AVC are investigated at the APP, where one source node utilizes DASH with AVC to serve multiple destination nodes. Hence, the transition-enabled cross-layer framework needs to be extended to incorporate the APP.

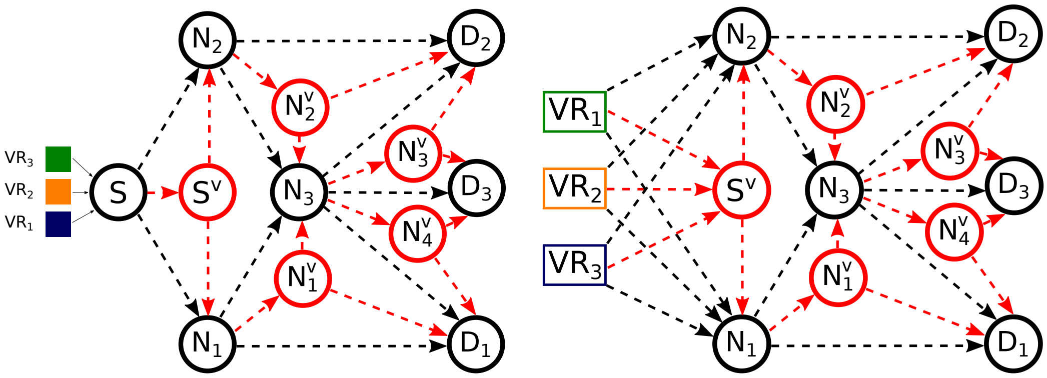
In the following sections, an application-aware transition-enabled cross-layer framework is developed, which incorporates jointly PHY, MAC, NET and APP. First, a brief description of DASH with AVC is given. Second, an extension of the unified graph is proposed to integrate the properties of DASH with AVC at the APP. Furthermore, based on the proposed extension and the requirements at the APP, a multi-source optimization problem is formulated. Next, a rate mapping problem is discussed, in order to map the bandwidth requirements at the APP to rate requirements at the PHY. Finally, the performance of the application-aware transition-enabled cross-layer framework is evaluated.

Some of the contributions presented in this chapter are published in [FAWK16].

## 4.2 Adaptive Bitrate Streaming

At the APP, DASH with AVC is a widely applied approach for adaptive video streaming. The video itself is non-adaptive, since it is encoded with AVC, which is a non-adaptive video codec. Instead, a video is encoded multiple times in various qualities, resulting in multiple representations of a video which are stored at a central instance. Thus, DASH with AVC provides $J$ different representations of a video, enabling adaption by switching between these different video representations. Each video representation has a different bandwidth requirement and has a different video quality. Depending on the desired video quality and the available bandwidth, a user requests the appropriate video representation. A different video representation is requested when the available bandwidth changes during the video stream. However, only one video representation can be requested at a time. Hence, DASH with AVC allows users to choose the most suitable video representation with respect to their own capabilities.

## 4.3 Graph Extension: Video Representations as Virtual Sources



(a) (b)

Figure 4.1: The source node $S$ in (a) is replaced in (b) by three virtual sources representing the three video representations $VR_1$, $VR_2$ and $VR_3$.

In order to maximize the resource utilization at the lower three layers, it is necessary to model the properties and requirements of DASH with AVC in the unified graph. Each video representation can be seen as an independent video source, since a video is encoded multiple times and stored as an independent video representation. Hence, each video representation is modeled as a virtual source to represent the $J$ different video representations. Thus, $VR_1, \ldots, VR_J$ contain the same video content, but each $VR_j$ provides the video content in a different quality. Furthermore, each $VR_j$, $j = \{1, \ldots, J\}$ has a data rate requirement $B_j$, $j = \{1, \ldots, J\}$. Hence, a user can request the video representation $VR_j$ by connecting with the appropriate virtual source symbolizing the video representation $VR_j$.

The unified graph is extended by replacing physical source nodes with $J$ virtual sources which represent the $J$ video representations, which are labeled as $VR_j$. A replacement of a physical source node by $J$ virtual sources is done as follows: First, the physical source node and its respective physical links, representing a connection to a neighbor node, and its respective virtual links to virtual nodes, where a virtual node represents either a broadcast or a multicast, are removed. In the second step, each virtual source $VR_j$ is connected through virtual links to the physical neighbor nodes of the replaced physical source node, which enables a virtual source to forward the video representation via a unicast transmission. Also, by connecting each virtual source $VR_j$, $j = \{1, \dots, J\}$ to virtual nodes of the former physical source node, each virtual source can then utilize either broadcast or multicast transmissions to forward a video representation. As an example, the source node $S$ of the graph depicted in Fig. 4.1 (a) is replaced, where $J = 3$, thus resulting in the virtual sources $VR_1$, $VR_2$ and $VR_3$. This results in an extended unified graph which is shown in Fig. 4.1 (b), where $S$ is replaced with $VR_1$, $VR_2$ and $VR_3$. Moreover, $VR_1$, $VR_2$ and $VR_3$ are connected to the nodes through the unicast and broadcast links of the original source node $S$ which is indicated through the black dashed and red dashed edges, respectively.

Now, the unified graph jointly models APP, NET, MAC and PHY.

## 4.4 Multi-Source Optimization Problem with Independent Sources

In this section, an optimization problem is formulated which takes into account the four layers APP, NET, MAC and PHY.

As mentioned before, a video is encoded multiple times and stored as $J$ video representations at a server. Hence, the extension of the unified graph introduces $J$ video representations as $J$ independent virtual sources. This implies that the optimization problem is a multi-commodity flow problem, where each virtual source may introduce a flow into the network. Moreover, a virtual source $VR_j$ only introduces a flow into the network if a user, hence a destination node, requests the video representation $VR_j$. Thus, the optimization problem has binary components. Furthermore, a user can receive a video representation only if the bandwidth requirement of the video representation is fulfilled. However, the bandwidth requirement of a video representation is given at the APP and hence this has to be translated to a physical layer rate. Thus, the physical layer rate has to be determined, based on the given bandwidth requirement at the APP, the amount of resources and the resource utilization by each destination.

Therefore, adaptive video streaming utilizing the adaption of DASH with AVC at the APP and mechanisms at the NET, MAC and PHY can be formulated as a binary nonlinear optimization problem with independent sources. The independence between the virtual sources leads to a competition for network resources, especially when destination nodes are served with different video representations. Moreover, the goal is to maximize the resource utilization in the network and hence to maximize the sum rate in the network. This is expressed in the utility function in Eq. (4.1), where the objective function maximizes the sum rate in the system by determining which virtual source $VR_j$, $j = \{1, \dots, J\}$ serves which destination node $D_d$, $d = \{1, \dots, D\}$, where $r(j, d)$ expresses the rate achieved between virtual source $VR_j$ and destination $D_d$.

$$\max \sum_{j=1}^{J} \sum_{d=1}^{D} r(j, d). \tag{4.1}$$

Since, the problem is a multi-commodity flow problem, flow and capacity constraints are required. The flow constraint which has to be satisfied by every virtual source is expressed in Eq. (4.2), where the sum of outgoing flows from virtual source $VR_j$ to destination $D_d$ equals $r(j, d)$.

$$\begin{aligned} &\sum_{k=1}^{K} \Bigg( \sum_{l \in \mathcal{O}(N)} f_l^{(k)}(j, d) \Bigg) = r(j, d), \\ &\text{for} \quad j = \{1, \dots, J\}, d = \{1, \dots, D\}. \end{aligned} \tag{4.2}$$

As before, every forwarding node has to fulfill the flow conservation constraint, where any incoming flow into a forwarding node must depart from the forwarding node. This is formulated in Eq. (4.3) as

$$\begin{aligned} &\sum_{k=1}^{K} \Bigg( \sum_{l \in \mathcal{O}(N)} f_l^{(k)}(j, d) - \sum_{l \in \mathcal{I}(N)} f_l^{(k)}(j, d) \Bigg) = 0, \\ &\text{for} \quad j = \{1, \dots, J\}, d = \{1, \dots, D\}. \end{aligned} \tag{4.3}$$

The flow traversing the network from virtual source $VR_j$ to destination node $D_d$ has to be finalized at destination node $D_d$, leading to Eq. (4.4), where the sum of incoming flows over the ingoing links of a destination node result in the rate $r(j, d)$.

$$\begin{aligned} &\sum_{k=1}^{K} \Bigg( - \sum_{l \in \mathcal{I}(N)} f_l^{(k)}(j, d) \Bigg) = -r(j, d), \\ &\text{for} \quad j = \{1, \dots, J\}, d = \{1, \dots, D\}. \end{aligned} \tag{4.4}$$

Again, the capacity in subgraph $k$ depends on the link capacity $c_l$ and the duration the link is utilized in the $k$-th subgraph, which is determined by the timeshare factor $\tau_k$. If a link is part of subgraph $G_k = (\mathcal{N}_k, \mathcal{L}_k)$ the indicator function $\mathbf{I}_{\mathcal{L}_k}(l)$ is one, else the indicator function is zero. This means that the link is not active in $\mathcal{G}_k$. The indicator function is written as

$$\mathbf{I}_{\mathcal{L}_k}(l) = \begin{cases} 1, & \text{if } l \in \mathcal{L}_k \\ 0, & \text{otherwise.} \end{cases} \tag{4.5}$$

Hence, each flow over a link is upper bounded by the capacity constraint, expressed in Eq. (4.6).

$$\begin{aligned}&\sum_{k=1}^{K} f_l^{(k)}(j,d) \leq \tau_k \cdot c_l \cdot \mathbf{I}_{\mathcal{L}_k}(l),\\ &\text{for} \quad j=\{1,\dots,J\}, d=\{1,\dots,D\}, \forall l \in \mathcal{L}_k.\end{aligned} \tag{4.6}$$

In contrast to the previously formulated multi-commodity flow problem, the properties and requirements of DASH with AVC at the APP are incorporated. First, every virtual source $VR_j$ only serves the destination node $D_d$, if the destination node requests the video representation $VR_j$. This is expressed in Eq. (4.7), where $x_{j,d}$ is either equal to one if video representation $VR_j$ should be received by destination node $D_d$ or zero otherwise.

$$r(j,d) = \begin{cases} r_j, & \text{if } x_{j,d}=1, \\ 0, & \text{if } x_{j,d}=0, \end{cases}, \quad \text{for} \quad j=\{1,\dots,J\}, d=\{1,\dots,D\}. \tag{4.7}$$

Moreover, the requirement that a destination node can receive only one video representation at a time is given by the constraint described in Eq. (4.8), where the sum over all binary variables $x_{j,d}$ for each destination node can only be either zero or one.

$$\sum_{j}^{J} x_{j,d} \leq 1, \quad \text{for} \quad d=\{1,\dots,D\}. \tag{4.8}$$

Based on the bandwidth requirements $B_j$, $j=\{1,\dots,J\}$ at the APP, relative weights $w_j$, $j=\{1,\dots,J\}$ can be determined. These weights express that a video representation $VR_j$ has a higher bandwidth requirement than $VR_{j-1}$. This is specified in Eq. (4.9).

$$r_{j+1} = w_j \cdot r_j, \quad \text{for} \quad j=\{1,\dots,J-1\}. \tag{4.9}$$

Furthermore, it must be ensured that all flows in the network are positive, which is expressed in Eq. (4.10).

$$\begin{aligned}&f_l^{(k)}(j,d) \geq 0,\\ &\text{for} \quad k=1,\dots,K, j=\{1,\dots,J\}, d=\{1,\dots,D\}, l \in \mathcal{L}_k.\end{aligned} \tag{4.10}$$

Each subgraph is active for a certain amount of time. This allocation of time is described by a timeshare factor, which is a relative measure and is formulated in Eq. (4.11).

$$\sum_{k=1}^{K} \tau_k = 1, \quad \text{for} \quad k=\{1,...,K\}. \tag{4.11}$$

Moreover, the timeshare factor is bounded between zero and one, which is expressed in Eq. (4.12).

$$0 \leq \tau_k \leq 1, \quad \text{for} \quad k = \{1, ..., K\}. \tag{4.12}$$

The constraints expressed in Eq. (4.7), Eq. (4.8) and in Eq. (4.9) reveal that the optimization problem is a binary non-linear problem, which is difficult to solve in general. This is due to the fact that $r_j$ is unknown and $x_{j,d}$ is binary. Therefore, a rate mapping problem is discussed next, which maps the bandwidth requirements at the APP to physical layer rates. The solution of the rate mapping problem are physical layer rates $r_j$, $j = \{1, \ldots, J\}$ and thus the binary non-linear problem can be rewritten into a binary linear problem. That can be achieved by omitting the constraint expressed in Eq. (4.9) and using the solutions obtained from solving the rate mapping problem as an input for $r_j$ in Eq. (4.7).

## 4.5 Rate Mapping Problem

As mentioned before, every video representation $VR_j$ has a bandwidth requirement $B_j$ at the APP, but how this bandwidth requirement relates to a physical layer rate is unknown. In order to achieve a mapping from bandwidth requirements $B_j$, $j = \{1, \ldots, J\}$ at the APP to the physical layer rate $r_j$, $j = \{1, \ldots, J\}$, the following needs to be considered. First, the bandwidth requirement $B_j$ of video representation $VR_j$ is smaller than the bandwidth requirement $B_{j+1}$ of video representation $VR_{j+1}$. This is due to the fact that the higher the order of the video representation, the higher the video quality. Second, the mapping from bandwidth requirement $B_j$ to physical layer rate $r_j$ needs to be adapted regularly, to incorporate the change in the PHY resources.

Therefore, a rate mapping problem is formulated, which determines the physical layer rates based on the ratios between bandwidth requirements and the achievable rates at the destination nodes. As a first step, an optimization problem is formulated based on the known ratios of bandwidth requirements $w_j$, $j = \{1, \ldots, J\}$ and the achievable rates $c_d$, $d = \{1, \ldots, D\}$ at the destination nodes. The goal of the optimization problem consists of two general parts: first, to allocate as many destination nodes with a rate $r_j$, $j = \{1, \ldots, J\}$ and second, to maximize the allocated rates such that the sum of all rates is maximized. This objective is formulated in Eq. (4.13), where $\alpha_{j,d}$ is a binary variable which is one when physical layer rate $r_j$ is allocated to destination node $D_d$, else it is zero. The utility is non-linear, since both $\alpha_{j,d}$ and $r_j$ are unknown.

$$\max \sum_{j=1}^{J} \sum_{d=1}^{D} \alpha_{j,d} \cdot r_j \tag{4.13}$$

The physical layer rate $r_j$ can only be allocated to destination node $D_d$, if it does not exceed the achievable rate $c_d$ at destination node $D_d$. This constraint is expressed in Eq. (4.14).

$$\alpha_{j,d} \cdot r_j \leq c_d, \quad \text{for} \quad j = \{1, \ldots, J\}, d = \{1, \ldots, D\}. \tag{4.14}$$

Moreover, only one physical layer rate $r_j$ can be allocated to a destination node $D_d$ at a time, since a destination node can only receive one video representation $VR_j$ at a time. Therefore, Eq. (4.15) expresses that for each destination node $D_d$, only one binary variable $\alpha_{j,d}$ can be active at a time.

$$\sum_{j=1}^{J} \alpha_{j,d} = 1, \quad \text{for} \quad d = \{1, \ldots, D\}. \tag{4.15}$$

All physical layer rates $r_j$, $j = \{1, \ldots, J\}$ at the PHY must satisfy the same ratios as all bandwidth requirements $B_j$, $j = \{1, \ldots, J\}$. This is ensured by the constraint in Eq. (4.16).

$$r_{j+1} = w_j \cdot r_j, \quad \text{for} \quad j = \{1, \ldots, J-1\}, \tag{4.16}$$

Lastly, Eq. (4.17) sets $\alpha_{j,d}$ to be binary.

$$\alpha_{j,d} \in \{0, 1\}, \quad \text{for} \quad j = \{1, \ldots, J\}, d = \{1, \ldots, D\}. \tag{4.17}$$

Finally, Eq. (4.18) ensures that the physical layer rates are non-negative.

$$r_j \geq 0 \quad \text{for} \quad j = \{1, \ldots, J\}. \tag{4.18}$$

The optimization problem formulated in Eq. (4.13) - (4.18) maps the bandwidth requirements at the APP to physical layer rates at the PHY. However, the optimization problem is also a non-linear binary problem, which has a smaller problem size compared to the overall problem formulated in Sec. 4.4, but is still difficult to solve. Instead of solving the non-linear binary problem, a heuristic is proposed which efficiently solves the rate mapping problem. This is achieved by utilizing the structure of the rate mapping problem, which has similarities to a container loading problem.

The achievable rates $c_d$, $\{1, \ldots, D\}$ can be illustrated as containers, which need to be filled with a physical layer rate $r_j$. The size of the physical layer rate $r_j$ can be varied, but the size must be in compliance with the ratios of all physical layer rates. Thus, the heuristic aims to fill the containers as much as possible with physical layer rate $r_j$ $j = \{1, \ldots, J\}$, which can be imagined as being a liquid. A solution is valid as long as no container is overfilled and the ratios between the physical layer rates are satisfied. The heuristic searches in a greedy fashion to obtain a solution, which is done as follows.

**Algorithm 2** Algorithm of the rate mapping heuristic

```
1: procedure GetRateMappingHeuristic
2:     Sort containers in descending order
3:     Set sum fill level equal to zero
4:     Set $r_j$, where $j = J$, equal to the container with the highest fill level capacity
5:     Calculate $r_1, r_2, \ldots, r_{J-1}$
6:     Fill containers and calculate the sum over all fill levels
7:     If the previous sum over all fill levels is smaller then the current sum over all fill levels set $j = j - 1$ and repeat 3-6
8:     Else stop
```

The Algorithm 2 starts by sorting all containers. Next, the biggest container is completely filled with the highest video representation, which has the highest bandwidth requirement. According to this allocation and the known ratios, the remaining physical layer rates $r_{J-1}, r_{J-2}, \ldots, r_1$ are determined. Subsequently, the containers are filled with a suitable physical layer rate. Furthermore, the levels of the containers are summed up. The above steps are repeated and another appropriate solution is determined and compared with the previous iteration. If the calculated sum over all fill levels of the previous iteration is lower than the sum calculated in the current iteration, then store the current solution and repeat the above steps, else the heuristic stops and chooses the solution of the previous step as the best solution. A rate mapping solution found by the heuristic contains one container filled completely with a physical layer rate $r_j$, based on this allocation and with the known ratios $w_1, w_2, \ldots, w_j$, the remaining physical layer rates are calculated and the remaining containers are filled accordingly.

## 4.6 Performance Analysis

In this section, the performance of the application-aware transition-enabled cross-layer framework (TSOR) is investigated. The scenario is as follows: One source node transmits a video to several destination nodes. Moreover, three video representations of the video are available, namely $VR_1$, $VR_2$ and $VR_3$. Furthermore, the relative bandwidth requirements are $B_1 = \frac{1}{3}$, $B_2 = \frac{2}{3}$ and $B_1 = 1$, respectively. Monte-Carlo simulations are performed for different network sizes and numbers of destination nodes. The performance of the framework is evaluated in terms of sum rate.

In a first step, the performance of the framework is compared against a lower bound (LB) and an upper bound (UP). The lower bound is determined by serving all destination nodes with the same rate. The upper bound is determined by serving those destination nodes which maximize the sum rate in the network. Both, the lower bound and upper bound are determined by utilizing transitions at the lower three layers, but not exploiting the properties of DASH with AVC.

In a second step, the framework is compared against three different schemes. The first scheme utilizes tree network support structures and unicast transmissions which is abbreviated TUC. The second scheme utilizes both butterfly and tree network support structures and broadcast transmissions which is abbreviated BBC. The third scheme utilizes tree at the NET and performs transitions between the PHY transmission mechanisms unicast and broadcast, which is abbreviated PHYx. Moreover, all three schemes utilize DASH with AVC at the APP. In Fig. 4.2, the performance of TSOR is compared with the lower bound and upper bound. The results are for a network consisting of 30 nodes and five destination nodes.

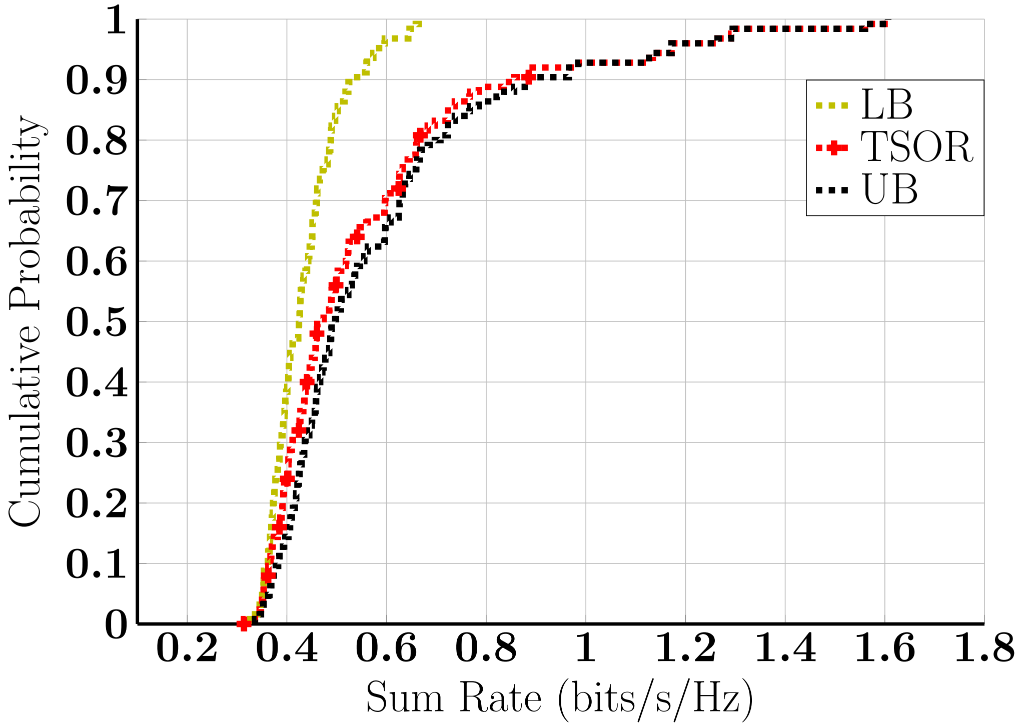


Figure 4.2: Simulation Result: Empirical cumulative distribution distribution function of the sum rate for lower bound, TSOR and upper bound.

The lower bound can achieves sum rate in the range between 0.32 bits/s/Hz and 0.67 bits/s/Hz. The median of the sum rate for the lower bound is 0.42 bits/s/Hz.

At the same time, TSOR achieves sum rates between 0.32 bits/s/Hz and 1.61 bits/s/Hz. Furthermore, TSOR achieves a median sum rate of 0.47 bits/s/Hz.

The upper bound achieves sum rates between 0.33 bits/s/Hz and 1.61 bits/s/Hz, while the median sum rate is 0.50 bits/s/Hz for the upper bound.

In general, the results depicted in Fig. 4.2 show that below, the median all three curves are performing close to each other, while above the median, TSOR and the upper bound outperform the lower bound, where the gain lies between 12 % and 140 %. One reason for these observations is the difference of the utility function formulated for the lower bound compared to the utility functions formulated for TSOR and the upper bound. The lower bound aims that all destination nodes are served, which reduces the utilization of the resources and thus the overall achievable sum rate. In contrast, TSOR and the upper bound are formulated in such a way that destination nodes with high resource utilization are allocated more resources and thus the overall achievable sum rate outperforms the achievable sum rate of the lower bound.

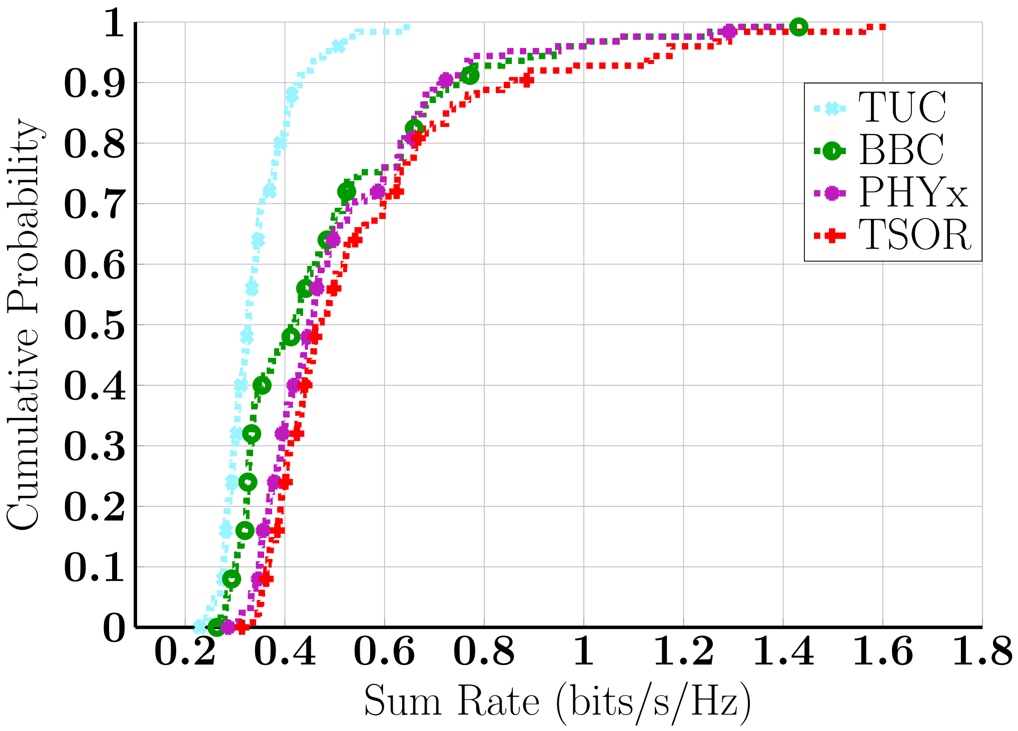


Figure 4.3: Simulation Result: Empirical cumulative distribution distribution function of the sum rate for TUC, BBC, PHYx and TSOR.

In Fig. 4.3, the performance of TSOR is compared with schemes which either cannot perform transitions or can only perform transitions at one layer.

The overall lowest performance is provided by TUC, where the sum rate achieved by TUC ranges between 0.23 bits/s/Hz and 0.65 bits/s/Hz, while the median of the sum rate is 0.33 bits/s/Hz for TUC. BBC achieves sum rates between 0.26 bits/s/Hz and 1.43 bits/s/Hz, while the median is 0.42 bits/s/Hz for BBC. PHYx achieves sum rates between 0.29 bits/s/Hz and 1.43 bits/s/Hz, while the median is 0.45 bits/s/Hz for PHYx.

In comparison, TSOR achieves sum rates in the range between 0.31 bits/s/Hz and 1.61 bits/s/Hz, while at the median TSOR achieves a sum rate of 0.47 bits/s/Hz. Thus, TSOR clearly outperforms TUC, which cannot perform transitions, at the lowest TSOR achieves a gain of 35 % compared to TUC, while at the highest a gain of 148 % can be achieved. Moreover, TSOR in comparison to BBC achieves in the low region a gain of 19 %, while at the highest the gain is 13 %. Furthermore, a gain of 7 % is achieved at the lowest by TSOR with respect to PHYx, while at the highest a gain of 13 % is achieved by TSOR compared to PHYx.

The results shown in Fig. 4.3, demonstrate that exploiting the broadcast nature of the wireless medium leads to gains, which is reasonable since a broadcast transmission between one node and its neighbor nodes requires one transmission, while in comparison to achieve the same with unicast transmissions requires more transmissions at least the number of neighbor nodes. In addition, the results also show that the utilization of transitions introduces gains, which is due to the fact that the goal of a transition is to choose the optimal combination of mechanisms.

In Fig. 4.4, the average sum rate for the four different schemes over increasing number of destination nodes and a network with 30 nodes is investigated. As seen in Fig. 4.4, as the number of destination nodes increases the average sum rate increases. In more detail, for the case of three destination nodes TSOR achieves an average sum rate of 0.34 bits/s/Hz which is a gain of 31 % compared to TUC, a gain of 17 % compared to BBC and a gain of 6 % compared PHYx, respectively.

For the case of four destination nodes TSOR achieves an average sum rate 0.47 bits/s/Hz, which is a gain of 52 % compared to TUC, a gain of 12 % compared to BBC and a gain of 4 % compared to PHYx, respectively.

In the case of five destination nodes TSOR achieves an average sum rate of 0.56 bits/s/Hz, TUC achieves an average sum rate of 0.34 bits/s/Hz, BBC achieves an

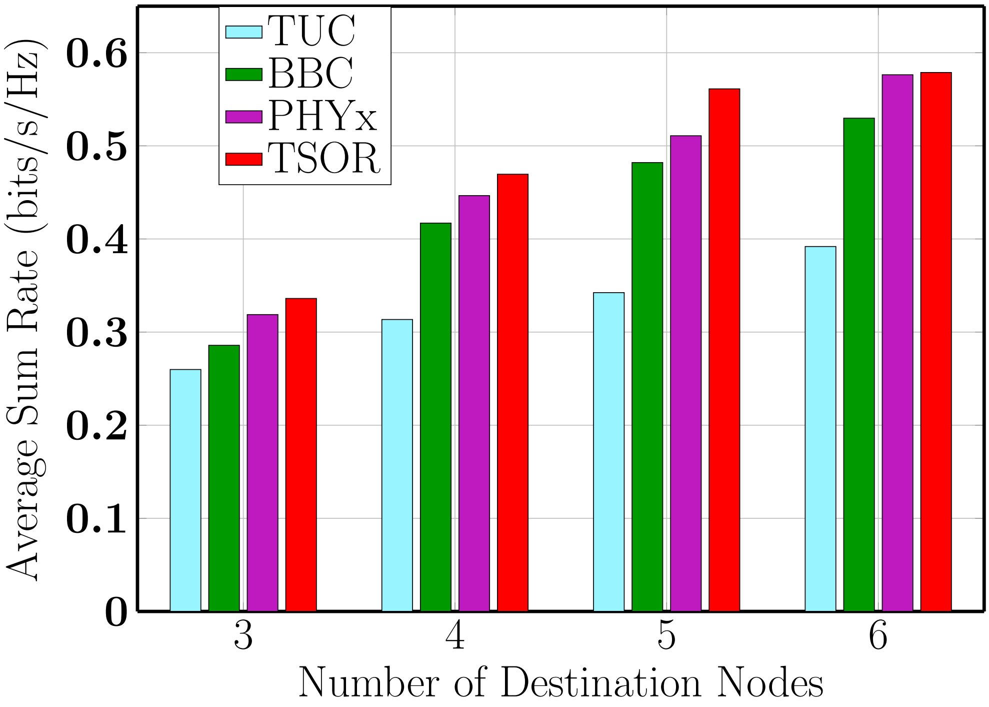


Figure 4.4: Simulation Result: Average sum rate of TUC, BBC, PHYx and TSOR over changing number of destination nodes.

average sum rate of 0.48 bits/s/Hz and PHYx achieves an average sum rate of 0.51 bits/s/Hz, which are gains between 10 and 65 %.

In the final case, six destination nodes are served. Here, TSOR achieves an average sum rate of 0.58 bits/s/Hz, TUC reaches an average sum rate of 0.39 bits/s/Hz, BBC achieves an average sum rate of 0.53 bits/s/Hz and PHYx achieves the same average sum rate as TSOR. Thus, TSOR can achieve a gain of 49 % compared to TUC and a gain of 10 % compared to BBC.

In general, an increase in sum rate with an increasing number of destination nodes is plausible, since the destination nodes are randomly chosen within the network, the probability increases that destination nodes with a high resource utilization are chosen. Furthermore, TUC is outperformed by BBC, PHYx and TSOR, because the three schemes utilize broadcast transmissions. In addition, PHYx and TSOR outperform TUC and BBC due to the utilization of transitions at the PHY.

In Fig. 4.5, the average sum rate for the four different schemes over increasing number of nodes in the network is examined, where the number of destination nodes is five. As shown in Fig. 4.5, as the number of nodes increases, the average sum rate stays on the same level for all schemes. In more detail, TUC achieves an average sum rate of 0.36

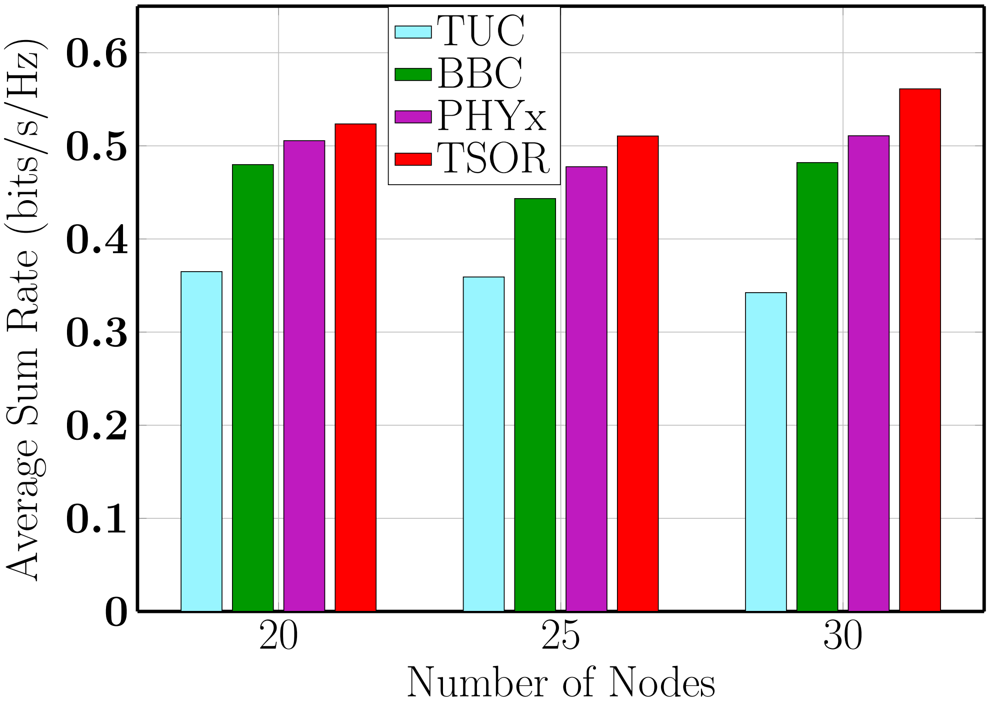


Figure 4.5: Simulation Result: Average sum rate of TUC, BBC, PHYx and TSOR over increasing number of nodes.

bits/s/Hz for a network size of 20 and 25 and an average sum rate of 0.34 bits/s/Hz for a network size of 30. In contrast, BBC achieves an average sum rate of 0.48 bits/s/Hz for a network sizes 20 and 30, while for a network size of 25 an average sum rate of 0.44 bits/s/Hz is achieved.

PHYx achieves an average sum rate of 0.50 bits/s/Hz for network size of 20, an average rate of 0.48 bits/s/Hz for a network size 25 and for a network size of 30 an average sum rate 0.51 bits/s/Hz.

In comparison, TSOR achieves the highest rates, which are 0.52 bits/s/Hz, 0.51 bits/s/Hz and 0.56 bits/s/Hz for network sizes 20, 25 and 30 respectively. Hence, TSOR achieves a gain between 44 % and 65 % compared to TUC, gains between 8 % and 16 % compared to BBC and gains between 4 % and 10 % compared to PHYx.

The results show that the three schemes achieve an almost constant sum rate over increasing number of nodes in the network. However, TSOR achieves a slight increase with an increasing network size, because TSOR utilizes transitions to find optimal combination of mechanisms on the lower three layers.

# Chapter 5

# Adaptive Video Content Distribution using RTP with SVC

## 5.1 Introduction

In this chapter, another way of combining adaptive video content distribution at the APP and available mechanisms at the NET, MAC and PHY is investigated. Here, the transition-enabled cross-layer framework is extended in such a way to incorporate RTP with SVC at the APP. In contrast, RTP with SVC provides adaptivity through video coding instead of storing and providing multiple representations of a video.

In the following sections, the application-aware transition-enabled cross-layer framework is adapted to utilize RTP with SVC at the APP. First, a short introduction of RTP with SVC is given. Second, the unified graph is extended to reflect the capabilities of RTP with SVC at the APP. Third, a multi-source optimization problem with dependent sources is formulated, which incorporates RTP with SVC at the APP and transitions at the lower three layers. Next, a rate mapping heuristic is proposed to map bandwidth requirements at the APP to physical layer rates. Finally, the performance of the application-aware transition-enabled cross-layer framework is compared with schemes which either cannot perform transitions or are not application-aware.

Some of the contributions presented in this chapter are published by the author of this thesis in [FAWK15].

## 5.2 Layered Coding Streaming

RTP with SVC is an adaptive video-streaming approach, where a video is encoded into multiple layers. Hence, the adaption in RTP with SVC is achieved through the scalable video codec and not by storing multiple copies of the video as compared to DASH with AVC. SVC separates a video into $J$ video layers with fixed bandwidth requirements. During the video streaming RTP with SVC can either scale up the video quality by transporting additional video layers or scale down by dropping some video layers. However, the reception or canceling of video layers has to be done in a successive order. Because, SVC generates each video layer based on the preceding video layers. This means a user needs to receive the lowest layer first and then the second lowest and soon and so fourth. In the same manner the cancelling has to be performed starting with the highest layer and the second highest layer and so on. The lowest video layer is referred to the base layer, while the higher video layers are referred as enhancement layers.

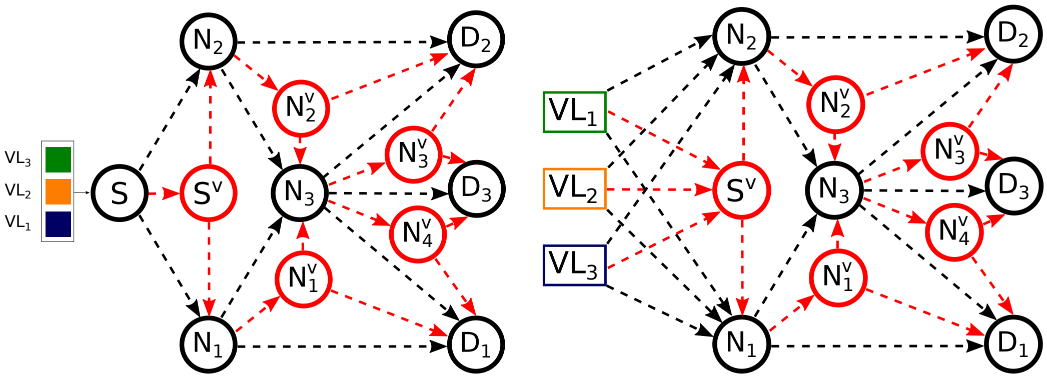
## 5.3 Graph Extension: Video Layers as Virtual Sources

RTP with SVC introduces $J$ video layers, where video layer $J$ can only be received if all previous video layers $1, \ldots, J-1$ have been received by a user. Hence, RTP with SVC at the APP is integrated into the unified graph by first introducing for each video layer a virtual source. Thus, every video layer $VL_j$, $j = \{1, \ldots, J\}$ a virtual source $VL_j$, $j = \{1, \ldots, J\}$ is introduced into the unified graph. The replacement follows the same procedure as in Ch. 4.3, the physical source nodes and its respective links are removed and instead virtual sources are introduced into the graph, where each virtual source is then connected via virtual links to the nodes in the graph. This is illustrated in Fig. 5.1, where source node $S$, is replaced by the three virtual sources $VL_1$, $VL_2$ and $VL_3$ which represent the base layer and two enhancement layers, respectively.

Furthermore, it must be ensured that video source $VL_J$ forwards video layer $VL_J$ only when the preceding video sources $VL_1, VL_2, \ldots, VL_{J-1}$ are received by the user. However, this successive order requirement cannot be expressed through an extension of the unified graph. Therefore, the APP requirement is formulated as a constraint in the multi-source optimization problem with dependent sources, which is discussed next.

## 5.4 Multi-Source Optimization Problem with Dependent Sources

After introducing the SVC video layers as virtual sources into the unified graph, an optimization problem can be formulated. As before, a utility function is formulated which aims to maximize the sum rate in the system. The utility function formulated



(a) (b)

Figure 5.1: The source node $S$ in (a) is replaced in (b) by three virtual sources representing the three video layers $VL_1$, $VL_2$ and $VL_3$.

in Eq. (5.1) maximizes the sum of rates between virtual sources and destination nodes, where $r(j,d)$ is the rate between virtual source $VL_j$ and destination node $D_d$.

$$\max \sum_{j}^{J} \sum_{d}^{D} r(j,d) \tag{5.1}$$

If virtual source $VL_j$ provides destination node $D_d$ with the respective video layer, then virtual source $VL_j$ introduces a flow into the network. This is expressed in Eq. (5.2), where the rate $r(j,d)$ between virtual source $VL_j$ and destination node $D_d$ is determined by the sum of flows $f_l^{(k)}(j,d)$ over all outgoing links originating at virtual source $VL_j$.

$$\sum_{k=1}^{K} \left( \sum_{l \in \mathcal{O}(N)} f_l^{(k)}(j,d) \right) = r(j,d), \quad \text{for} \quad j = \{1,\dots,J\}, d = \{1,\dots,D\}. \tag{5.2}$$

As before, nodes which are neither a virtual source or a destination node have to forward the incoming flow. This conservation constraint is formulated in Eq. (5.3), where the sum of incoming flows and the sum of outgoing flows at a forwarding node are equal.

$$\sum_{k=1}^{K} \left( \sum_{l \in \mathcal{I}(N)} f_l^{(k)}(j,d) - \sum_{l \in \mathcal{O}(N)} f_l^{(k)}(j,d) \right) = 0, \quad \text{for} \quad j = \{1,\dots,J\}, d = \{1,\dots,D\}. \tag{5.3}$$

Finally, flows introduced by virtual source $VL_j$ are finalized at the respective destination node, which is conveyed in Eq. (5.4).

$$\sum_{k=1}^{K}\Bigg(-\sum_{l\in\mathcal{I}(N)} f_l^{(k)}(j,d)\Bigg) = -r(j,d), \tag{5.4}$$
$$\text{for} \quad j=\{1,\ldots,J\}, d=\{1,\ldots,D\}.$$

As before, at any given time flows transported over link $l$ must be smaller or equal to the respective link capacity $c_l$. This is reflected in the capacity constraint given in Eq. (5.5).

$$\sum_{k}^{K} f_l^{(k)}(j,d) \leq \tau_k \cdot c_l \cdot \mathbf{I}_{\mathcal{L}_k}(l), \tag{5.5}$$
$$\text{for} \quad k=\{1,\ldots,K\}, j=\{1,\ldots,J\}, d=\{1,\ldots,D\}, l\in\mathcal{L}_k.$$

As mentioned before, video layers have to be received in a successive order. Eq. (5.6) ensures this, where $x_{j-1,d}$ is binary and is set to one if $VL_{j-1}$ is received by $D_d$ and zero otherwise. Thus, destination node $D_d$ can only receive video layer $VL_j$, if it receives $VL_{j-1}$ which is only received if $VL_{j-2}$ is received and so forth.

$$x_{j,d} \geq x_{j+1,d}, \quad \text{for} \quad j=\{1,...,J-1\}, d=\{1,\ldots,D\}. \tag{5.6}$$

Moreover, if $x_{j,d}$ is one, then the rate $r(j,d)$ is set equal to the PHY rate $r_j$, which is expressed in Eq. (5.7).

$$r(j,d) = \begin{cases} r_j, & \text{if } x_{j,d}=1, \\ 0, & \text{if } x_{j,d}=0, \end{cases}, \quad \text{for} \quad j=\{1,\ldots,J\}, d=\{1,\ldots,D\}. \tag{5.7}$$

As before, flows in the network can only be positive, which is in expressed in Eq. (5.8).

$$f_l^{(k)}(j,d) \geq 0, \tag{5.8}$$
$$\text{for} \quad k=\{1,...,K\}, j=\{1,\ldots,J\}, d=\{1,\ldots,D\}, l\in\mathcal{L}_k.$$

The timeshare factor $\tau_k$, which is the duration subgraph $G_k$ is active, is normalized in Eq. (5.9).

$$\sum_{k=1}^{K}\tau_k = 1, \quad \text{for} \quad k=\{1,...,K\}. \tag{5.9}$$

Finally, all timeshares are bound between zero and one, which is given Eq. (5.10).

$$0 \leq \tau_k \leq 1, \quad \text{for} \quad k=\{1,...,K\}. \tag{5.10}$$

## 5.5 Rate Mapping Problem

Similar to DASH with AVC, RTP with SVC has bandwidth requirements for each video layer at the APP which have to be mapped to physical layer rates. Therefore, similarly to the rate mapping for the video representation, an optimization problem for mapping bandwidth requirements of video layers at the APP to physical layer rates is formulated. As before, the achievable rates of each user are seen as containers, where the objective is to fill the containers as much as possible.

Again, a utility function is formulated which allocates video layers such that the sum of fill levels of all containers is maximized. This is formulated in Eq. (5.11), where $r_j$ is the physical layer rate of video layer $j$ and $\alpha_{j,d}$ is binary, which is set to one if video layer $j$ is allocated to user $i$ and is zero otherwise.

$$\max \sum_{i=j}^{J} \sum_{d=1}^{D} \alpha_{j,d} \cdot r_j \tag{5.11}$$

A user can get all video layers as long as two conditions are met. First, the container cannot overfill which is assured as long as the sum of all allocated physical layer rates is smaller or equal to the height of the container. This constraint is expressed in Eq. (5.12), where $\alpha_{j,d}$ is the binary variable $r_j$ is the physical layer rate of video layer $j$ and $c_d$ is the container height of user $d$.

$$\sum_{j=1}^{J} \alpha_{j,d} \cdot r_j \leq c_d, \quad \text{for} \quad d = \{1, \ldots, D\}. \tag{5.12}$$

Second, it must be ensured that video layers are assigned in successive order. Thus, Eq. (5.13) expresses that each user $i$ can only get a higher video layer if all lower video layers are also received.

$$\alpha_{j,d} \geq \alpha_{j+1,d}, \quad \text{for} \quad j = \{1, \ldots, J-1\}. \tag{5.13}$$

Finally, Eq. (5.14) expresses the relative relationship between video layer $j$ and the next higher video layer $j+1$, which has to be satisfied also by the respective physical layer rates $r_j$ and $r_{j+1}$, where $w_j$ expresses the relative ratio between them.

$$r_{j+1} = w_j \cdot r_j, \quad \text{for} \quad j = \{1, \ldots, J-1\}. \tag{5.14}$$

Lastly, Eq. (5.15) sets $\alpha_{j,d}$ to be binary.

$$\alpha_{j,d} \in \{0, 1\}, \quad \text{for} \quad j = \{1, \ldots, J\}, d = \{1, \ldots, D\}. \tag{5.15}$$

The rate mapping problem for RTP with SVC formulated in Eq. (5.11) - Eq. (5.15) is a non-linear binary problem. Hence, a heuristic is utilized to efficiently solve the rate mapping problem. Since, the rate mapping problem for RTP with SVC has a similar structure as the rate mapping problem formulated for DASH with AVC, the same approach is followed as in Sec. 4.5.

## 5.6 Performance Analysis

In this section, the benefits of RTP with SVC at the APP combined with transitions at the lower three layers is evaluated. First, a wireless multihop network consisting of 30 nodes with one source node and five destination nodes is investigated. The source node has a SVC encoded video available to serve the five destination nodes. Moreover the SVC encoded video is composed of three video layers, one base layer and two enhancement layers. Second, the performance is assessed for a fixed network size but changing number of destination nodes. Third, the number of destination nodes is fixed and the size of the network is changed. The performance is analyzed in terms of the achieved sum rate in the network.

As before, the proposed application-aware transition-enabled cross-layer framework is compared with the lower bound and upper bound. Again, schemes which either cannot perform transitions or can only perform transitions at one single layer are investigated. In order to asses the benefits of the proposed framework. In Fig. 5.2, the cdf of the lower bound, TSOR and the upper bound are depicted, respectively. As it can be seen, the lower bound achieves sum rates in the range between 0.33 bits/s/Hz and 0.67 bits/s/Hz. Furthermore, the median for the lower bound is 0.43 bits/s/Hz. In comparison, TSOR achieves sum rates in the range between 0.34 bits/s/Hz and 1.61 bits/s/Hz. Moreover, at the median TSOR achieves a sum rate of of 0.5 bits/s/Hz. Similarly, the upper bound achieves sum rates between 0.34 bits/s/Hz and 1.61 bits/s/Hz, while the median sum rate is 0.5 bits/s/Hz for the upper bound.

Hence, TSOR performs very close to the upper bound and achieves a gain up to 140 % with respect to the lower bound. The reason for this observation is that the utility of the lower bound is aiming to solve all destination nodes, while the upper bound and TSOR aim to maximize the sum rate in the system.

Fig. 5.3 shows the performance of TUC, BBC, PHYx and TSOR for a network with 30 nodes, where one source nodes serves five destination nodes. The overall lowest performance is achieved by TUC, where the range of sum rates with TUC is between

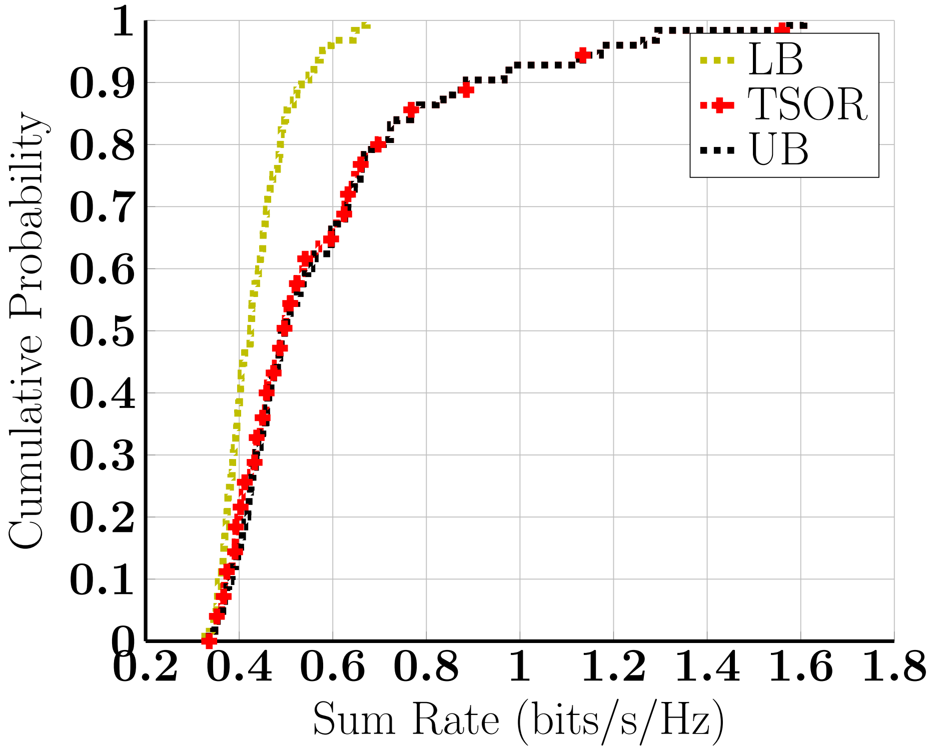


Figure 5.2: Simulation Result: Empirical cumulative distribution distribution function of the sum rate for lower bound, TSOR and upper bound.

0.26 bits/s/Hz and 0.97 bits/s/Hz. At the median, TUC reaches a sum rate of 0.39 bits/s/Hz. In comparison, BBC achieves sum rates between 0.26 bits/s/Hz and 1.43 bits/s/Hz, while at the median BBC reaches a sum rate of 0.42 bits/s/Hz. Hence, achieves a gain up to 47 % compared to TUC. The scheme PHYx achieves sum rates between 0.33 bits/s/Hz and 1.43 bits/s/Hz, which is a gain between 17 % up to 47 % with respect to both TUC and BBC. At the median a sum rate of 0.49 bits/s/Hz can be observed for PHYx, which is a gain of 26 % and 17 % compared to TUC and BBC, respectively.

TSOR achieves sum rates between 0.33 bits/s/Hz and 1.61 bits/s/Hz, while the median sum rate for TSOR is 0.50 bits/s/Hz. Thus, TSOR achieves gains between 28 % and 66 % compared to TUC, while the gains are between 13 % and 27 % with respect to BBC. Moreover, even though the scheme PHYx and TSOR perform close to each other, TSOR achieves higher sum rates above the median compared to PHYx, which leads to gains up to 13 %.

The reason TUC is outperformed by BBC, PHYx and TSOR is that these three schemes at some level utilize transitions, which allows to utilize the optimal mechanisms for a given network. Moreover, BBC is outperformed in lower region which is due to

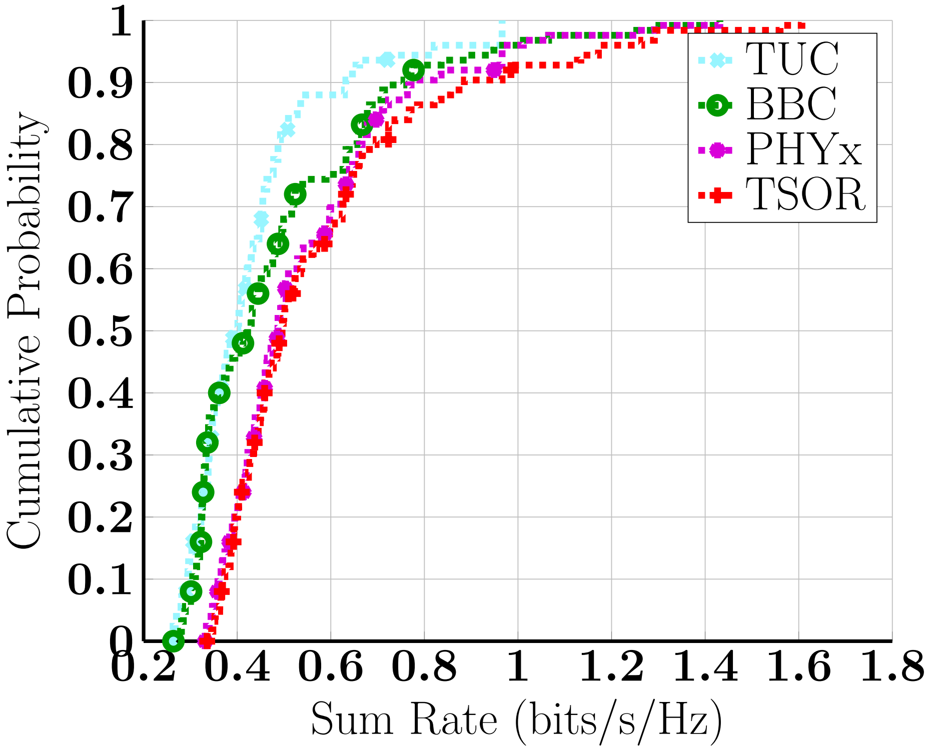


Figure 5.3: Simulation Result: Empirical cumulative distribution distribution function of the sum rate for TUC, BBC, PHYx and TSOR.

the fact that both PHYx and TSOR utilize transitions at the PHY, where switching from broadcast transmissions to unicast transmissions is beneficial to avoid bottlenecks which can occur already with one low capacity link in the network. Furthermore, TSOR can outperform all schemes, since it utilizes transitions on multiple layers, hence TSOR finds an optimal combination of mechanisms.

In Fig. 5.4, the average sum rate for the four different schemes over increasing number of destination nodes and a network with 30 nodes is investigated. As seen in Fig. 5.4, as the number of destination nodes increases the average sum rate increases. In more detail, for the case of three destination nodes TSOR achieves an average sum rate of 0.41 bits/s/Hz which is a gain of 10 % compared to TUC, a gain of 28 % compared to BBC and a gain of 2.5 % compared PHYx, respectively.

For the case of four destination nodes TSOR achieves an average sum rate 0.51 bits/s/Hz, which is a gain of 28 % compared to TUC, a gain of 21 % compared to BBC and a gain of 4 % compared to PHYx, respectively.

In the case of five destination nodes TSOR achieves an average sum rate of 0.59 bits/s/Hz, TUC achieves an average sum rate of 0.43 bits/s/Hz, BBC achieves an

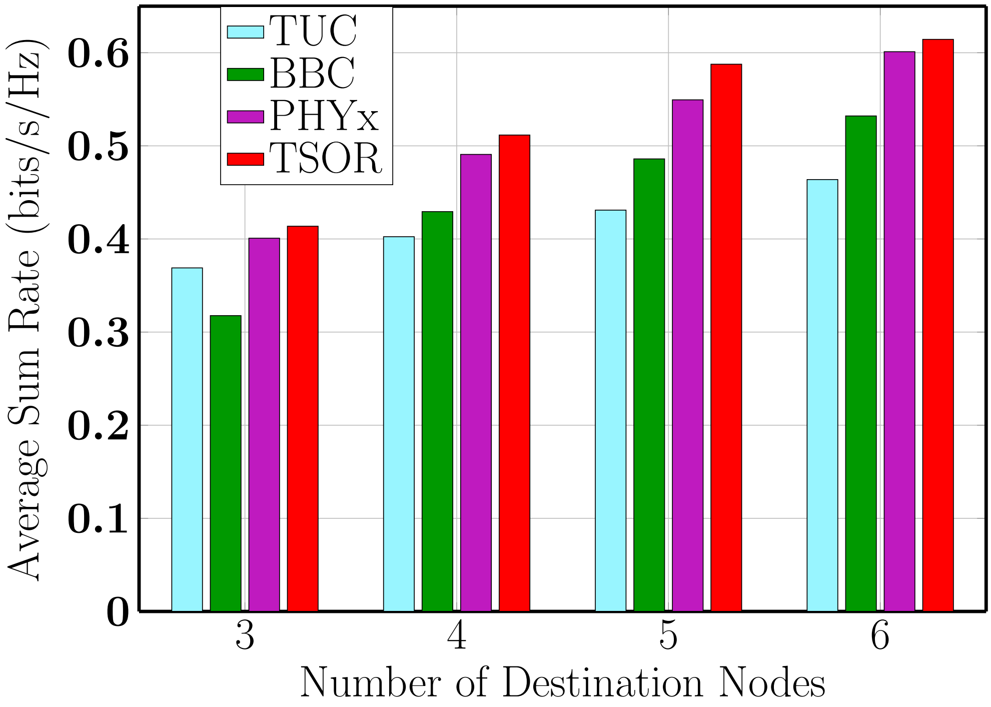


Figure 5.4: Simulation Result: Average sum rate of TUC, BBC, PHYx and TSOR over changing number of destination nodes.

average sum rate of 0.49 bits/s/Hz and PHYx achieves an average sum rate of 0.55 bits/s/Hz, which are gains between 7 and 37 %.

In the final case, six destination nodes are served. Here, TSOR achieves an average sum rate of 0.61 bits/s/Hz, TUC reaches an average sum rate of 0.46 bits/s/Hz, BBC achieves an average sum rate of 0.53 bits/s/Hz and PHYx achieves an average sum rate of 0.60 bits/s/Hz. Thus, TSOR can achieve a gain of 33 % compared to TUC and a gain of 15 % compared to BBC.

The results illustrated in Fig. 5.4, show that for an increase in destination nodes results in an increase of sum rate, which makes sense since more destination nodes leads to a potential increase in resource utilization. Furthermore, this increase is facilitated by transitions, where the optimal mechanisms are chosen to achieve maximum sum rates in the system. In Fig. 5.5, the average sum rate for the four different schemes over increasing number of nodes in the network is examined, where number of destination nodes is five. TUC achieves an average sum rate between 0.43 and 0.45 bits/s/Hz for network sizes of 20, 25 and 30. In contrast, BBC achieves an average sum rate of 0.49 bits/s/Hz for a network sizes 20 and 30, while for a network size of 25 an average sum rate of 0.46 bits/s/Hz is achieved.

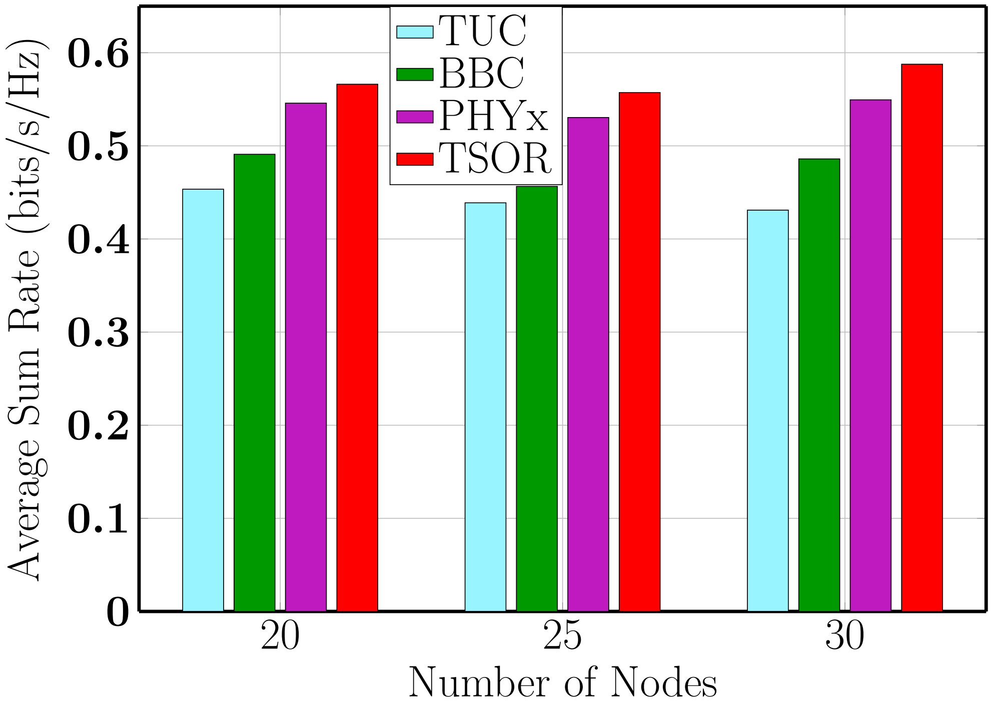


Figure 5.5: Simulation Result: Average sum rate of TUC, BBC, PHYx and TSOR over increasing number of nodes.

PHYx achieves an average sum rate of 0.53 bits/s/Hz for network size of 25, an average rate of 0.55 bits/s/Hz for a network size 25 and 30.

In comparison, TSOR achieves the highest rates, which are 0.57 bits/s/Hz, 0.56 bits/s/Hz and 0.59 bits/s/Hz for network sizes 20, 25 and 30 respectively. Hence, TSOR achieves a gain between 27 % and 37 % compared to TUC, gains between 16 % and 22 % compared to BBC and gains between 3 % and 7 % compared to PHYx.

As shown in Fig. 5.5, as the number of nodes increases the average sum rate stays on the same level for all schemes. However, TSOR always achieves higher sum rates, since it finds the optimal combination of mechanisms at the lower three layers.

# Chapter 6

# Aggregation of Distributed Cached Content

## 6.1 Introduction

In this chapter, a scenario is considered where content is not only available at one central instance, but in addition, content is cached in a distributed fashion over the complete network. The aim is to investigate the advantage of aggregating content over the network instead of single source delivery. Furthermore, the gain of combining content caching and cross-layer transitions at the lower three layers is examined.

Hence, a short introduction into caching in wireless multihop networks is given first in Section 6.2. Second, in Section 6.3, an extension to the graph based model of the transition-enabled cross-layer framework is proposed to incorporate caching. Next, in Section 6.4, a binary linear optimization problem is formulated, which solves the problem of aggregating content from multiple sources to multiple destination nodes. Finally, in Section 6.5, the performance of content caching combined with lower layer transitions is compared against schemes which utilize cached content, but cannot perform transitions.

Several parts of this chapter have been originally published by the author in [FMAK17].

## 6.2 Caching in Wireless Networks

The availability of content at multiple points in a wireless network can improve resource utilization, since the content is cached closer to the user. In this thesis, a scenario is considered where mobile devices with limited caches have either the complete content or a fraction of a content in their cache. Furthermore, the mobile devices form a wireless multihop network in which the content is simultaneously requested by multiple destination nodes within the network. Moreover, it is assumed that the content is already cached in a distributed manner among nodes in the wireless multihop network, which requested the content previously.

Hence, all nodes in the network with cached content are potential sources, which improves the access of content in the network. However, the complexity to obtain an optimal solution increases as well, since a destination can only receive the content or a subset of it from one source at a time. Therefore, a graph extension is proposed next, which represents content as virtual sources and connects nodes with cached content with the respective virtual sources.

## 6.3 Graph Extension: Super-Source Virtualization for Distributed Content

As before, the network consists of nodes $\mathcal{N} = \{N_1, \dots, N_I, D_1, \dots, D_M\}$ and links $\mathcal{L} = \{1, \dots, L\}$. A content $F$ is not only available at one source, but is also available at other nodes which have either the complete content or parts of it in their caches.

It is assumed that a content $F$ is separable and composed of $P$ parts, $F = \bigcup\limits_{p=1}^{P} F_p$, where the parts are disjoint, i.e., $F_p \cap F_q = \emptyset$, $\forall p \neq q$. Furthermore, a node $N_i$ caches $p$ parts of $F$, where $0 \leq p \leq P, p \in \mathbb{N}_0$ . Moreover, destination nodes $\{D_1, \dots, D_M\} \subset \mathcal{N}$ request the complete content or missing parts of it.

In order to illustrate the scenario of cached content in a wireless multihop network, an example is depicted in Fig. 6.1. In Fig. 6.1, the content $F$ is composed of four disjoint parts, denoted by $F_1, F_2, F_3, F_4$. Five nodes, $N_1, N_2, N_3, N_4, N_5$, have parts of the content cached and three destination nodes $D_1, D_2, D_3$ request the complete content $F = \{F_1, F_2, F_3, F_4\}$. As shown in Fig. 6.1, different nodes have different parts of the content, e.g. $N_2$ cached $F_1$ and $F_2$, while $N_4$ has $F_3$ and $F_4$ in the cache. Destination nodes may request any part of a content from any node which has the respective part of the content in its cache. However, a destination node cannot request a part of the content from more than one node. Therefore, the goal is to determine for each destination node and each part of the content from which node that part of the content should be retrieved and through which paths it should be delivered.

It has to be noted that these decisions are coupled and depend on the available paths, resources and channel qualities with respect to all destination nodes. The number of possible forwarders as well as the number of possible paths to the destination nodes are high when the number of nodes with cached content is high, which enlarges the complexity of the problem. Thus, a modeling is required which can cope with a high

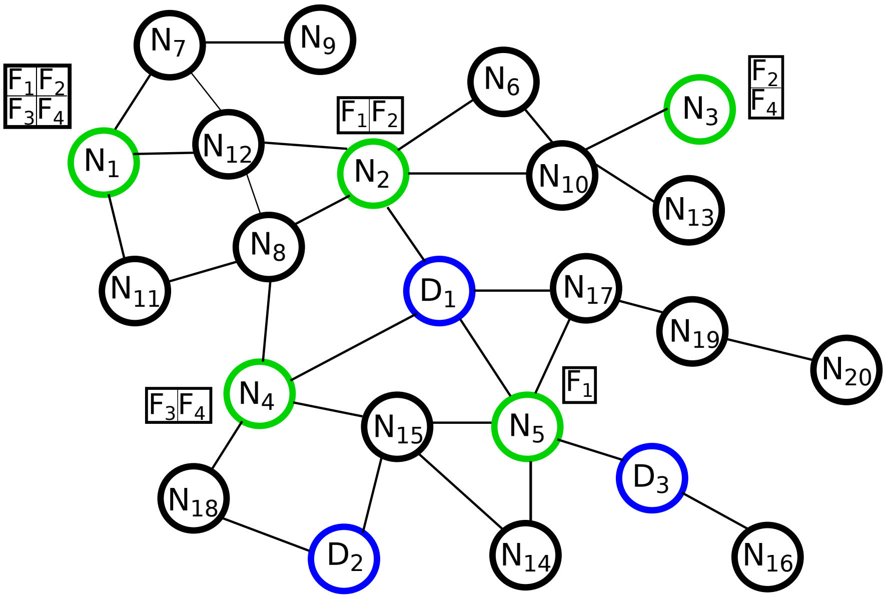


Figure 6.1: Destination nodes $D_1$, $D_2$ and $D_3$, marked in blue, request content $F$, which is distributed over the wireless multihop network, where nodes with the complete or part of the content in the cache are marked in green.

number of nodes with cached content. Specifically, the model needs to take into account the possibility that nodes cache only parts of the content.

Since every part $F_p$ of content $F$ can be cached at multiple nodes, each part $F_p$ is modeled as a virtual source $S_p^{\mathrm{v}}$. Each $S_p^{\mathrm{v}}$ is connected through virtual links to nodes that have the content cached. The model is illustrated in Fig. 6.2, where the four parts are modeled as virtual sources $S_1^{\mathrm{v}}$, $S_2^{\mathrm{v}}$, $S_3^{\mathrm{v}}$ and $S_4^{\mathrm{v}}$, respectively. The virtual sources are connected through virtual links with nodes that have the part $F_p$ in their cache, indicated by the green dotted lines. For example, $S_1^{\mathrm{v}}$ is connected to $N_1$, $N_2$ and $N_5$.

By introducing super-sources, the number of sources to be considered will be fixed and instead, only the number of available paths will increase with every additional node with cached content in the network. This reduces the complexity, where before a node with cached content represented an additional source and additional paths in the network. Furthermore, this approach allows to formulate a multi-source optimization problem similar to those before. Hence, a multi-source multi-destination optimization problem is discussed next.

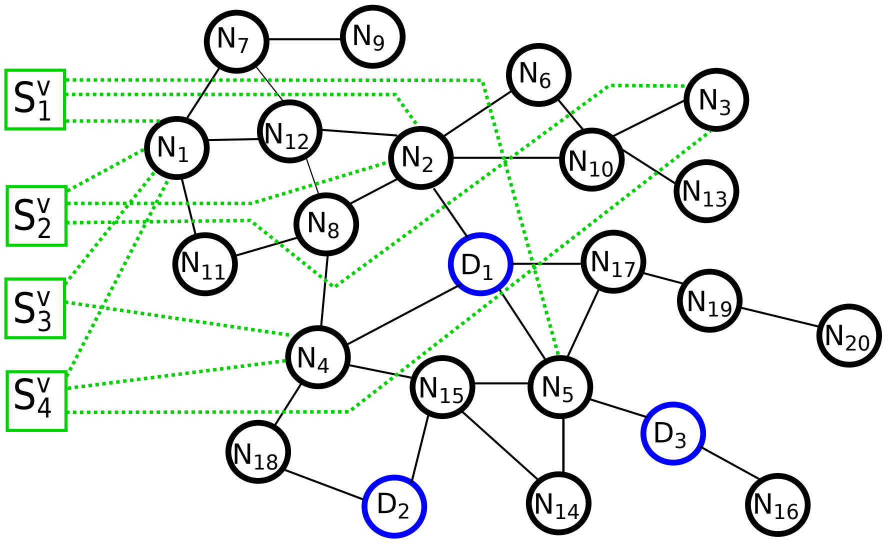


Figure 6.2: For the four parts $F_1$, $F_2$, $F_3$ and $F_4$ of content $F$, four virtual sources $S^{\mathrm{v}}_1$, $S^{\mathrm{v}}_2$, $S^{\mathrm{v}}_3$ and $S^{\mathrm{v}}_4$ are introduced, which are connected via virtual links with nodes that have the respective part in their cache, indicated by green dotted lines.

## 6.4 Multi-Source Multi-Destination Optimization Problem

In this section, a multi-source multi-destination sum rate maximization problem for content delivery is formulated. As already mentioned, it is assumed that the current cache content of the nodes in the network is known. Hence, the proposed transition-enabled cross-layer framework for caching can be combined with any caching policy. In the proposed optimization problem, the multiple sources correspond to the virtual sources which represent the content. The transition-enabled cross-layer framework for caching aims at maximizing the sum rate in the system by choosing which nodes forward which parts in their cache to the destination nodes. At the same time, the best combination of mechanisms to forward the content at the lower three layers is determined. The utility function is formulated as

$$\max \sum_{d=1}^{D} \min_{p} r(p,d), \tag{6.1}$$

where $r(p,d) \geq 0$ is the rate achieved between virtual source $S^{\mathrm{v}}_p$ and destination node $D_d$. The utility function expresses *i)* that each destination node receives each part with the same rate, which is achieved by setting all rates between virtual sources $S^{\mathrm{v}}_p$, $p = \{1,\ldots,P\}$ and a destination node $D_d$ to $\min\limits_{p} r(p,d)$, ensuring that the complete content is received and *ii)* the maximization of the sum rate in the system.

The flow from virtual source $S^{\mathrm{v}}_p$ to destination node $D_d$ over a link $l$ in the $k$-th subgraph is defined as $f_l^{(k)}(p,d)$. The outgoing flow from virtual source node $S_p^{\mathrm{v}}$ to destination node $D_d$ over the outgoing virtual link $l$ is $r(p,d)$ if the link is activated, else it is zero. The activation of a virtual link $l$ between virtual source node $S_p^{\mathrm{v}}$ and destination node $D_d$ is indicated with the binary variable $y_l(p,d)$. Hence, the flow constraint for every virtual source is expressed in Eq. (6.2).

$$\sum_{k=1}^{K} f_l^{(k)}(p,d) = \begin{cases} r(p,d), & \text{if } y_l(p,d)=1, \\ 0, & \text{otherwise,} \end{cases}, \tag{6.2}$$
$$\text{for} \quad p \in \{1,...,P\}, d \in \{1,...,D\}, l \in \mathcal{O}(S_p^{\mathrm{v}}).$$

Since we assume that for a given destination node, a given part $F_p$ of the content can only be retrieved from one node, only one outgoing virtual link of a virtual source node can be activated for serving destination node $D_d$. This is written in Eq. (6.3).

$$\sum_{l \in \mathcal{O}(S_p^{\mathrm{v}})} y_l(p,d) \leq 1, \quad \text{for} \quad p = \{1,...,P\}, d = \{1,...,D\}. \tag{6.3}$$

The rate between $S_p^{\mathrm{v}}$ and $D_d$ is constrained by the maximum flow in the network [AMO93]. Every node in $\mathcal{N} \cup \mathcal{N}^{\mathrm{v}}$ must satisfy the flow conservation, which conveys that any incoming flow into a node must depart from the node. The flow conservation constraint for forwarding nodes which is expressed in Eq. (6.4).

$$\sum_{k=1}^{K} \left( \sum_{l \in \mathcal{O}(N)} f_l^{(k)}(p,d) - \sum_{l \in \mathcal{I}(N)} f_l^{(k)}(p,d) \right) = 0, \tag{6.4}$$
$$\text{for} \quad p = \{1,...,P\}, d = \{1,...,D\}, \forall N \in (\mathcal{N} \cup \mathcal{N}^{\mathrm{v}}) \setminus \{D_d\}.$$

The flow constraint for destination nodes is formulated in Eq. (6.5).

$$\sum_{k=1}^{K} \left( - \sum_{l \in \mathcal{I}(D_d)} f_l^{(k)}(p,d) \right) = -r(p,d), \tag{6.5}$$
$$\text{for} \quad p = \{1,...,P\}, d = \{1,...,D\}.$$

Every flow in the network is upper bounded by a capacity constraint. The capacity of a link $l$ in subgraph $\mathcal{G}_k$ depends on the link capacity $c_l$ and the duration the link is utilized in the $k$-th subgraph, which is determined by the timeshare factor $\tau_k$. If a link is part of subgraph $\mathcal{G}_k$, the indicator function $\mathbf{I}_{\mathcal{L}_k}(l)$ is one, else the flow on that link is zero in $\mathcal{G}_k$. The indicator function is written as

$$\mathbf{I}_{\mathcal{L}_k}(l) = \begin{cases} 1, & \text{if } l \in \mathcal{L}_k, \\ 0, & \text{otherwise.} \end{cases} \tag{6.6}$$

The capacity constraint is expressed by

$$0 \leq \sum_{p=1}^{P} f_l^{(k)}(p,d) \leq \tau_k \cdot c_l \cdot \mathbf{I}_{\mathcal{L}_k}(l), \tag{6.7}$$
$$\text{for} \quad k = \{1, ..., K\}, d = \{1, ..., D\}, \forall l \in \mathcal{L},$$

which bounds all flows through a link. The timeshare factor $\tau_k$, which is the duration during which subgraph $G_k$ is active, is normalized as

$$\sum_{k=1}^{K} \tau_k = 1, \quad \text{for} \quad k = \{1, ..., K\}. \tag{6.8}$$

Lastly, all timeshares are bound between zero and one, which is expressed by

$$0 \leq \tau_k \leq 1, \quad \text{for} \quad k = \{1, ..., K\}. \tag{6.9}$$

The proposed multi-source multi-destination optimization problem is expressed in Eq. (6.1) - Eq. (6.9). This optimization problem is a non-linear binary problem.

We rewrite the above problem as a binary linear problem. First, the objective function formulated in Eq. (6.1) is a piecewise-linear function and can be reformulated [BV04]. This is done in Eq. (6.10), where $t_d$ is an auxiliary variable.

$$\max \sum_{d=1}^{D} t_d, \tag{6.10}$$

Moreover, $t_d$ is constrained by the rate, which is expressed in Eq. (6.11).

$$t_d \leq r(p,d), \quad \text{for} \quad p = \{1, ..., P\}, d = \{1, ..., D\}. \tag{6.11}$$

Furthermore, the constraint in Eq. (6.2) can be written either as one binary non-linear constraint or as a set of three binary linear constraints by applying the big-M method [GNS09]. Here, the latter approach is chosen. The first constraint sets the outgoing flows of $S_p^{\mathrm{v}}$ to zero if $y_l(p,d)$ is equal to zero which is expressed in Eq. (6.12), where $M_1$ is a constant that should be chosen sufficiently large but close enough to the upper bound of $\sum_{k=1}^{K} f_l^{(k)}(p,d)$.

$$\sum_{k=1}^{K} f_l^{(k)}(p,d) \leq M_1 y_l(p,d), \tag{6.12}$$

Since the maximum flow is bounded by the link capacities in the network and the maximum physical capacity in the unified graph is normalized to one, we can set $M_1 = 1$. Next, the second part in Eq. (6.2), where the outgoing flows are set equal to $r(p,d)$ when $y_l(p,d)$ is equal to one, is enforced by two constraints. This is formulated in Eq. (6.13) and Eq. (6.14), where $M_2$ is set as the upper bound of the left-hand side expression in Eq. (6.13) and $M_3$ is set as the upper bound of the left-hand side expression in Eq. (6.14).

$$\sum_{k=1}^{K} f_l^{(k)}(p,d) - r(p,d) \leq M_2(1 - y_l(p,d)), \tag{6.13}$$

$$r(p,d) - \sum_{k=1}^{K} f_l^{(k)}(p,d) \leq M_3(1 - y_l(p,d)), \tag{6.14}$$

Since the link capacities in our model are normalized to one, the maximum rate and flows in the network are upper bounded to one and hence $M_2 = M_3 = 1$.

Thus, after rewriting the objective function in Eq. (6.1) and the non-linear binary constraint in Eq. (6.2), the multi-source multi-destination optimization problem becomes a binary linear problem.

## 6.5 Performance Analysis

In this section, the performance of caching and transitions at the lower three layers is evaluated in terms of the achievable sum rate.

The proposed transition-enabled cross-layer framework for caching is abbreviated TSOR. TSOR is compared against to two schemes which cannot perform transitions. The first one is using the butterfly structure at the NET and broadcast at the PHY, which is abbreviated BBC. The second scheme is utilizing the tree structure at the NET and unicast at the PHY, which is abbreviated with TUC.

Therefore the following scenario is considered: The content consists of four parts and it is assumed that all four parts are available in at least one node. Furthermore, some nodes have $P_i$ parts of the content in their cache according to a uniform distribution. Moreover, four cases are investigated *i*) each node has one part of the content ($X = 1$) in its cache, *ii*) each node has less than or equal to two random parts ($X \leq 2$) in its cache, *iii*) each node has less than or equal to three random parts ($X \leq 3$) in its cache and *iv*) each node has less than or equal to four random parts ($X \leq 4$) in its cache.

Fig. 6.3 shows the average sum rate achieved by TSOR over the percentage of nodes with at most $X$ parts of the content in their cache for the four cases of $X$. In all four cases, it can be observed that the average sum rate increases as the number of

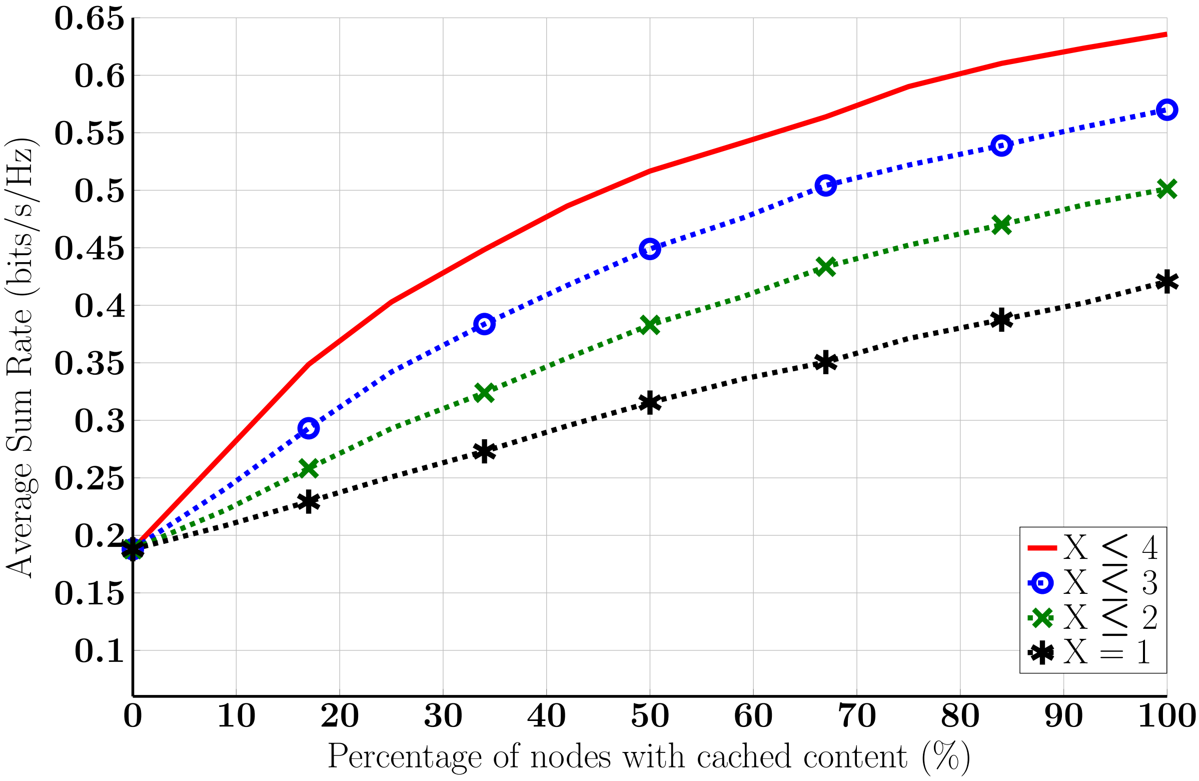


Figure 6.3: Simulation Result: Sum rate achieved by TSOR vs. relative number of nodes with cached content for different X parts cached at nodes.

nodes with cached content increases. This is due to the fact that with more nodes with cached content in the network, the content is getting closer to the destination nodes. Additionally, it can be observed that the performance increases the more parts of a content are cached at each node. In the case of $X \leq 4$, the maximum sum rate achieved is 0.64 bits/s/Hz, which is a gain of 12% compared to $X \leq 3$, a gain of 28% compared to $X \leq 2$ and a gain of 52% compared to $X = 1$. The curves show a concave behavior, except for the case $X = 1$ where the curve follows a linear trend.

In Fig. 6.4, the average sum rate achieved by TSOR is shown for the cases when 75%, 50%, 25% of nodes in the network have parts of the content in their cache, as well as when the caches of the nodes are empty. TSOR achieves an average sum rate of 0.19 bits/s/Hz without caching. In comparison, the average sum rate increases by 33% when $X = 1$, 54% when $X \leq 2$, 80% when $X \leq 3$ and 112% when $X \leq 4$, for the case that 25% of nodes have cached some parts of the content. This shows the advantage of exploiting caching, but it also shows that the number of parts cached at each node has an impact on the achievable sum rate.

As an example, an average sum rate of 0.3 bits/s/Hz is achieved for the case $X = 1$ when 50% of the nodes have cached some parts of the content. Similarly, this can be

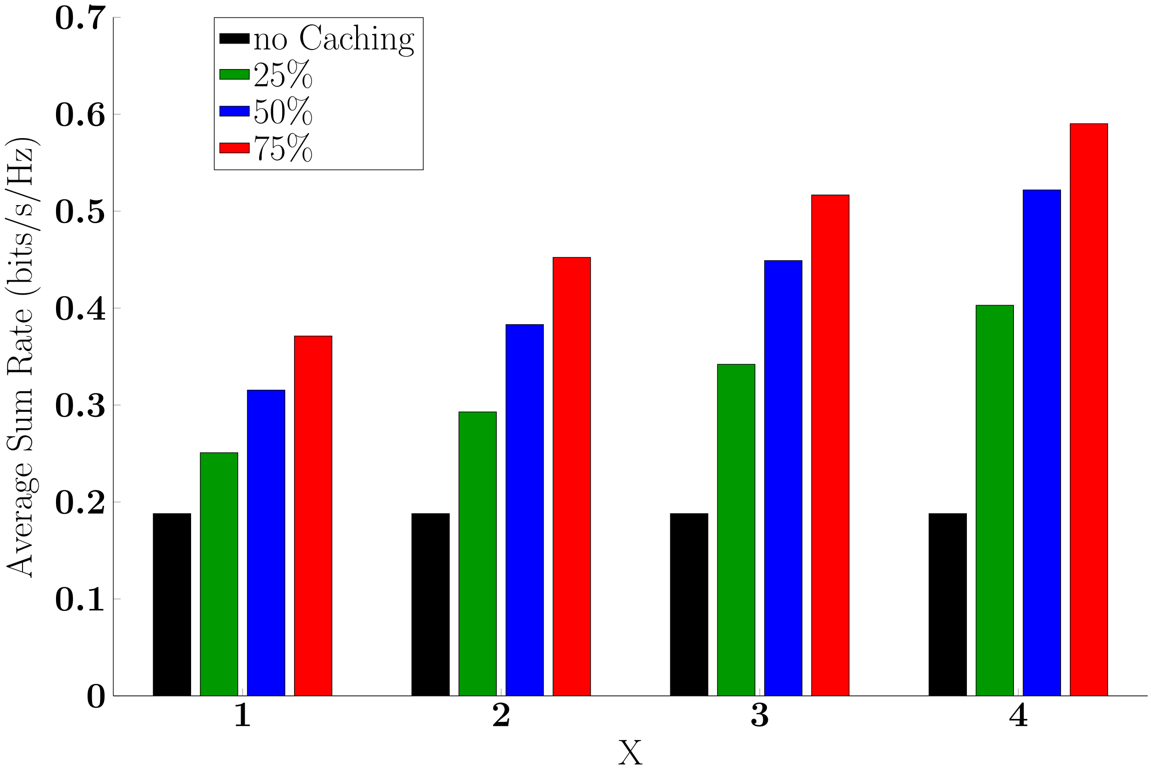


Figure 6.4: Simulation Result: Average sum rate for different $X$ of parts cached at nodes for 0%, 25%, 50% and 75% of caching nodes.

observed for the case $X \leq 4$ which requires 25% of the nodes with parts of the content in the cache. The reasons is that a nearby node with multiple parts of the content in the cache will provide all parts in the cache to a destination node if selected, thus reducing the number of active sources and increasing the amount of available resources. Hence, both factors, i.e., the number of nodes with content in the cache and the amount of parts cached at nodes play a role.

Next, the utilization of transitions at the lower three layers is investigated. Fig. 6.5 shows the average sum rate over the percentage of nodes with part of the content in the cache for BBC, TUC, and TSOR for the two cases $X = 1$ and $X \leq 4$. When the caches of the nodes are empty, TSOR achieves an average sum rate of 0.19 bits/s/Hz, which is two times higher than for TUC and 1.5 times higher compared to BBC. For the case $X = 1$, TSOR outperforms both TUC and BBC, where TSOR achieves an average sum rate of 0.42 bits/s/Hz, which is a gain of 50% and 40% compared to TUC and BBC, respectively. Moreover, TSOR outperforms both TUC and BBC for the case of $X \leq 4$, where TSOR achieves an average sum rate between 0.19 bit/s/Hz and 0.64 bits/s/Hz. This results in an average gain of 40% compared to TUC and BBC. Thus, Fig. 6.5 shows the benefit of exploiting caching and performing transitions, since TSOR can adapt to changing network conditions by switching between tree and butterfly at

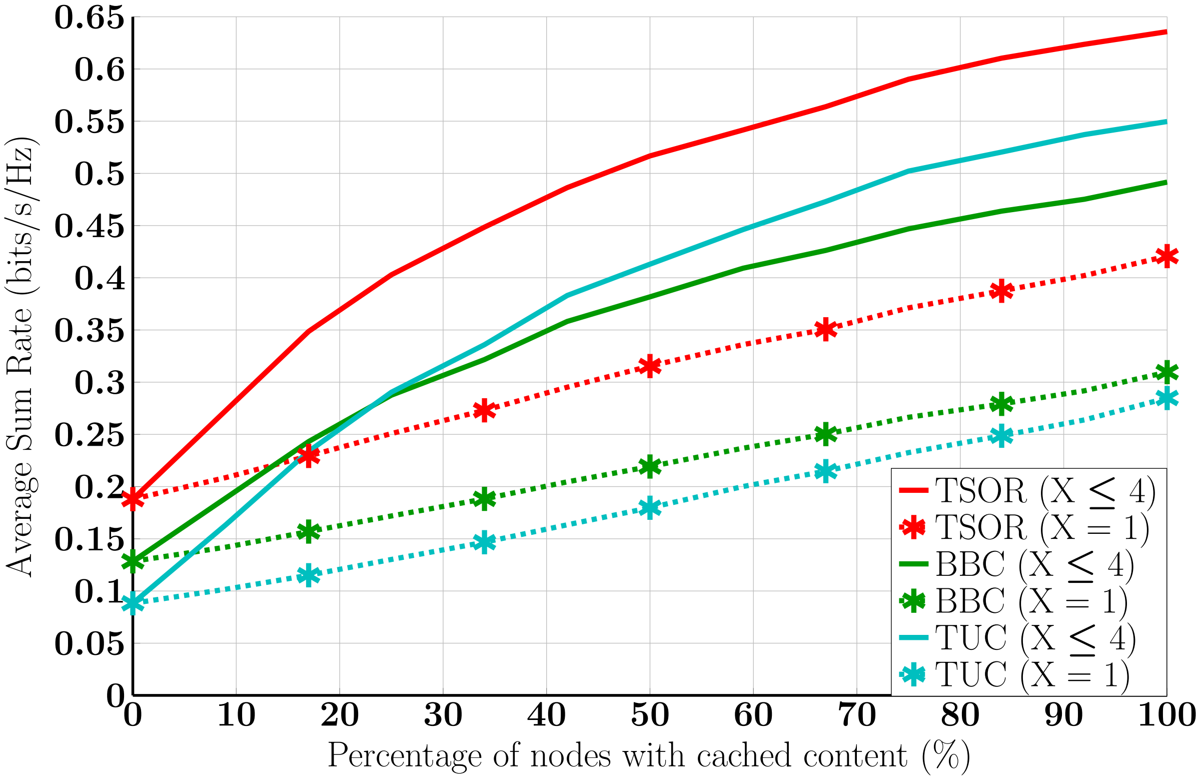


Figure 6.5: Simulation Result: Average sum rate for $X = 1$ and $X \leq 4$ for BBC, TUC and TSOR.

the NET and between unicast and broadcast transmissions at the PHY.

# Chapter 7

# Conclusions

## 7.1 Summary

In this thesis, a novel transition-enabled cross-layer framework was proposed, which models the PHY, MAC, NET and integrates properties and constraints from the APP. The main scenarios of this work are single-source multi-destination and multi-source multi-destination scenarios. These two scenarios cover the communication patterns in digital communication with the most growth and impact.

In Chapter 1, the challenge of content distribution and aggregation in wireless multihop networks is introduced. Also an overview of the state of the art is presented. Furthermore, the open issues are identified and formulated. Finally, the main contributions and an overview of the thesis are given.

In Chapter 2, the lower three layers PHY, MAC and NET are discussed, as well as adaptive video streaming and caching with respect to the APP layer.

In Chapter 3, a new unified graph-based model is presented, which integrates the mechanisms and properties from each layer. Moreover, the concept of transitions on the lower three layers is integrated into a cross-layer framework and a single-rate multi-source optimization problem is formulated. The performance analysis shows that the utilization of transitions leads to higher throughput and is also able to compensate the presence of bottlenecks by switching between different combination of mechanisms.

In Chapter 4, the transition-enabled cross-layer framework is combined with the flexibility of DASH at the APP. Further, it provides the optimal combination of video representation with network support structures at the NET and communication types at the PHY in a unified graph. Moreover, a novel heuristic for DASH is presented, which maps the relative data rate requirements of the video representations to the physical resources. A multi-source optimization problem is formulated, where each video representation is expressed as an independent virtual source. The performance analysis investigates the performance of the proposed transition-enabled cross-layer framework and compares it with two approaches which cannot perform transitions and one approach which only performs transitions on the PHY. The proposed solution achieves gains of up to 35 % in terms of outage capacity and up to 148 % with respect to the maximum achievable sum rate. Finally, a comparison of the different schemes where done with increasing number of nodes in the network. Again, the proposed solution outperformed the schemes not utilizing transitions, where gains up to 65 % were achieved.

In Chapter 5, the challenge of transporting adaptive video content using RTP with SVC is discussed. The underlying problem is similar to the problem discussed in Chapter 4. However, while the adaptive video streaming with DASH translates into an optimization problem where all video representations are independent from each other, this is not the case when adaptive video streaming using SVC is considered. Adaptive video streaming with SVC results in a optimization problem where multiple sources are present but are only utilized when all the lower quality layers are also transmitted, because each video layer depends on all it preceding video layers. Similar to the previous chapter, to solve the above multi-source optimization problem a rate mapping heuristic is formulated. The performance analysis demonstrates the advantages and gains possible when transitions are utilized. The proposed transition-enabled cross-layer framework achieves up to 66 % gains compared to two approaches which cannot perform transitions and up to 13 % compared to an approach which only performs transitions at the PHY. Furthermore, the proposed framework outperforms the compared approaches when the number of destination nodes is increased where gains between 28 % and 37 % are achieved. This also holds when the number of nodes in the network are increased, where the proposed solutions achieves gains up to 37 %.

In Chapter 6, the problem of aggregating distributed content is discussed. Here, the graph-based model is extended by representing distributed content as super-source nodes. This leads to the formulation of a multi-source multi-destination optimization problem. The performance analysis shows that the more content is cached at intermediate nodes, the higher the average sum rate, leading to gains of up to 52 %. Furthermore, the amount of nodes which are caching has a huge impact, which results into gains of up to 300 %. Also, the utilization of transitions shows that gains of up to 50 % can be achieved compared to approaches which do not perform transitions.

In conclusion, the proposed transition-enabled cross-layer framework shows that transitions provide significant gains. These can be further increased by incorporating requirements and properties of applications, e.g. adaptive video streaming, into a unified model, i.e., a graph-based model. Finally, by additionally incorporating caching into the scenario, it is possible to utilize aggregation of content from multiple devices.

## 7.2 Outlook

A novel application-aware transition-enabled cross-layer framework was proposed in this thesis. The proposed framework demonstrates that combing available mechanisms on different layers, enabling transitions and incorporating the requirements and capabilities of applications, e.g. adaptive video-streaming, can result in significant gains. Therefore, extending the framework to cover more services, e.g. telephony, environmental sensing, data processing, is a logical next step. In order to achieve an extension of services, two important aspects have to be studied further. The first aspect is to consider the absence of global information. The proposed framework and therefore the possible gains are based on the availability of global information at a central instance. Although, in a fairly simple and slow changing wireless multihop network assuming global information is reasonable. The framework can only scale, if the framework can handle scenarios where either information is available in a more distributed way, or where collected information, e.g. availability of links, becomes outdated or both. An approach to handle these kind of scenarios is to incorporate mechanisms which handle the signaling, collection and updating of information with respect to the state of the wireless multihop network. It should be mentioned that promising research is already conducted which proposes transition-enabled monitoring framework. Furthermore, it was shown that the framework can be formulated as an optimization problems. In some cases the optimization problem is a linear problem, but for most cases the problem is a binary linear problem, which does not scale well. This issue can be tackled by extending the framework in such way that a limited group of nodes determine an optimal combination of low-layer mechanisms. One approach to implement decision making with partial information and observability of local environment would be a reinforcement learning, where the actions are mechanisms or certain combination of mechanisms, the environment is represented by the unified graph-based model and the reward/loss function could be expressed as a relative relationship between the amount of data coming into the section of the network and the amount of data going out of the network. Another aspect of the thesis was the integration of the application layer, but the utility functions where chosen such that the resources at the PHY where utilized optimally. Alternatively, one could formulate a more user centric utility functions, e.g. a utility function which maximizes the quality of experience for all users. Finally, extending the proposed graph based approach to represent more mechanisms, e.g. policy-based routing or properties, e.g. buffers would be quite interesting.

# List of Acronyms

| | |
|---|---|
| **APP** | Application Layer |
| **AR** | Artificial Reality |
| **AVC** | Advanced Video Coding |
| **D2D** | Device-to-Device |
| **DASH** | Dynamic Adaptive Streaming via HTTP |
| **HTTP** | Hypertext Transfer Protocol |
| **LB** | Lower Bound |
| **MAC** | Medium Access Layer |
| **NET** | Network Layer |
| **NP-hard** | Nondeterministic Polynomial time hard |
| **OFDM** | Orthogonal Frequency Division Multiplexing |
| **PHY** | Physical Layer |
| **RTP** | Real-Time Transport Protocol |
| **SNR** | Signal-to-Noise Ratio |
| **SVC** | Scalable Video Coding |
| **UB** | Upper Bound |
| **WMN** | Wireless Multihop Network |

# List of Symbols

| | |
|---|---|
| $\mathcal{G}$ | directed graph |
| $\mathcal{N}$ | set of nodes |
| $\mathcal{L}$ | set of links |
| $N_i$ | i-th transmiting node |
| $N_j$ | j-th receiving node |
| $l$ | link between $N_i$ and $N_j$ |
| $\sum$ | |
| $\mathcal{O}(\cdot)$ | set of outgoing links of a node |
| $\lvert \cdot \rvert$ | cardinality of a set |

# Bibliography

[15] "White paper: Cisco visual networking index data traffic forecast update, 2014-2019", Cisco, Tech. Rep., 2015.

[ACLY00] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow", *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000, ISSN: 0018-9448. DOI: `10.1109/18.850663`.

[ADI+12] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking", *IEEE Communications Magazine*, vol. 50, pp. 26–36, 2012.

[AMO93] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network flows : theory, algorithms, and applications*. Prentice Hall, 1993.

[AWB+19] B. Alt *et al.*, "Transitions: A protocol-independent view of the future internet", *Proceedings of the IEEE*, no. 4, pp. 835–846, Feb. 2019, ISSN: 0018-9219. [Online]. Available: `http://tubiblio.ulb.tu-darmstadt.de/113662/`.

[BGW10] S. Borst, V. Gupta, and A. Walid, "Distributed caching algorithms for content distribution networks", in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.

[BV04] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004, ISBN: 0521833787.

[CQZ+12] H. Cui, D. Qian, X. Zhang, C. Jing, and Y. Sun, "Joint source-network coding optimization for video streaming over wireless multi-hop networks", in *Proc. IEEE Vehicular Technology Conference (VTC Spring)*, May 2012, pp. 1–5. DOI: `10.1109/VETECS.2012.6240202`.

[DTXL11] H. Deng, X. Tao, T. Xing, and J. Lu, "Resource allocation for layered multicast streaming in wireless OFDMA networks", in *Proc. IEEE International Conference on Communications (ICC)*, Jun. 2011, pp. 1–5. DOI: `10.1109/icc.2011.5963496`.

[FAWK15] M. Fasil, H. Al-Shatri, S. Wilk, and A. Klein, "Application-Aware Cross-Layer framework: Video content distribution in wireless multihop networks", in *Proc. IEEE 26th Annual International Symposium on Personal, Indoor and Mobile Radio Communications - (PIMRC)*, Hong Kong, P.R. China, Aug. 2015, pp. 1088–1093.

[FAWK16] M. Fasil, H. Al-Shatri, S. Wilk, and A. Klein, "A network-centric view on dash in wireless multihop networks", in *Proc. IEEE 84th Vehicular Technology Conference (VTC-Fall)*, Sep. 2016, pp. 1–6. DOI: `10.1109/VTCFall.2016.7880861`.

[FKK14] M. Fasil, A. Kuehne, and A. Klein, "Node virtualization and network coding: Optimizing data rate in wireless multicast", in *Proc. International Symposium on Wireless Communications Systems (ISWCS)*, Aug. 2014, pp. 573–578. DOI: `10.1109/ISWCS.2014.6933419`.

[FMAK17] M. Fasil, S. Mueller, H. Al-Shatri, and A. Klein, “Exploiting caching and cross-layer transitions for content delivery in wireless multihop networks”, in *15th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, May 2017, pp. 1–7. DOI: `10.23919/WIOPT.2017.7959877`.

[FRLB15a] A. Frömmgen, R. Rehner, M. Lehn, and A. Buchmann, “Fossa: Learning eca rules for adaptive distributed systems”, in *International Conference on Autonomic Computing (ICAC)*, Jul. 2015. [Online]. Available: `http://tubiblio.ulb.tu-darmstadt.de/74574/`.

[FRLB15b] A. Frömmgen, R. Rehner, M. Lehn, and A. Buchmann, “Fossa: Using genetic programming to learn eca rules for adaptive networking applications”, in *Proceedings of the 38th Annual IEEE Conference on Local Computer Networks (LCN’15), 2015*, 2015. [Online]. Available: `http://tubiblio.ulb.tu-darmstadt.de/77246/`.

[FYH+14] C. Fang, F. R. Yu, T. Huang, J. Liu, and Y. Liu, “A survey of energy-efficient caching in information-centric networking”, *IEEE Communications Magazine*, vol. 52, pp. 122–129, 2014.

[GNS09] I. Griva, S. G. Nash, and A. Sofer, *Linear and nonlinear optimization.* Siam, 2009.

[GPT13] S. Gitzenis, G. S. Paschos, and L. Tassiulas, “Asymptotic laws for joint content replication and delivery in wireless networks”, *IEEE Transactions on Information Theory*, vol. 59, no. 5, pp. 2760–2776, May 2013.

[Hoh20] F. Hohmann, “Cooperative communication in wireless OFDMA multihop networks”, Ph.D. dissertation, Technische Universität Darmstadt, Darmstadt, 2020. DOI: `https://doi.org/10.25534/tuprints-00011541`. [Online]. Available: `http://tuprints.ulb.tu-darmstadt.de/11541/`.

[JCM16] M. Ji, G. Caire, and A. F. Molisch, “Wireless device-to-device caching networks: Basic principles and system performance”, *IEEE Journal on Selected Areas in Communications*, vol. 34, pp. 176–189, 2016.

[JHJC15] S.-W. Jeon, S.-N. Hong, M. Ji, and G. Caire, “Caching in wireless multihop device-to-device networks”, in *Proc. IEEE International Conference on Communications (ICC)*, 2015, pp. 6732–6737.

[JTLC14] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, “On the average performance of caching and coded multicasting with random demands”, in *Proc. IEEE International Symposium on Wireless Communications Systems (ISWCS)*, 2014, pp. 922–926.

[KSL07] S. Kwack, H. Seo, and B. G. Lee, “Suitability-based subcarrier allocation for multicast services employing layered video coding in wireless OFDM systems”, in *Proc. IEEE Vehicular Technology Conference (VTC Fall)*, Sep. 2007, pp. 1752–1756. DOI: `10.1109/VETECF.2007.370`.

[LE13] S. Lakshminarayana and A. Eryilmaz, "Multirate multicasting with intralayer network coding", *IEEE/ACM Transactions on Networking*, vol. 21, no. 4, pp. 1256–1269, Aug. 2013, ISSN: 1063-6692. DOI: `10.1109/TNET.2012.2226909`.

[LLJL05] Z. Li, B. Li, D. Jiang, and L. C. Lau, "On Achieving Optimal Throughput with Network Coding", in *Proceedings IEEE INFOCOM 2005.*, vol. 3, Mar. 2005, 2184–2194 vol. 3. DOI: `10.1109/INFCOM.2005.1498493`.

[Loc15] A. Loch, "Enabling efficient, robust, and scalable wireless multi-hop networks: A cross-layer approach exploiting cooperative diversity", Ph.D. dissertation, Technische Universität Darmstadt, Darmstadt, Feb. 2015. [Online]. Available: `http://tuprints.ulb.tu-darmstadt.de/4470/`.

[LTGK13] J. Llorca, A. M. Tulino, K. Guan, and D. C. Kilper, "Network-coded caching-aided multicast for efficient content delivery", in *Proc. IEEE International Conference on Communications (ICC)*, 2013, pp. 3557–3562.

[LZLS12] L. Lei, Z. Zhong, C. Lin, and X. Shen, "Operator controlled device-to-device communications in LTE-advanced networks", *IEEE Wireless Communications*, vol. 19, no. 3, pp. 96–104, Jun. 2012.

[MK19] M. Mousavi and A. Klein, "Decentralized video streaming in multi-hop wireless networks: Incentive mechanism and energy efficiency", *IEEE Access*, vol. 7, pp. 51 329–51 345, 2019.

[MMA+16] M. Mousavi, S. Müller, H. Al-Shatri, B. Freisleben, and A. Klein, "Multi-hop data dissemination with selfish nodes: Optimal decision and fair cost allocation based on the shapley value", in *2016 IEEE International Conference on Communications (ICC)*, 2016, pp. 1–6.

[MN14] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching", *IEEE Transactions on Information Theory*, vol. 60, pp. 2856–2867, 2014.

[NBK12] R. Niati, A. Banihashemi, and T. Kunz, "Throughput and energy optimization in wireless networks: Joint mac scheduling and network coding", *IEEE Transactions on Vehicular Technology*, vol. 61, no. 3, pp. 1372–1382, Mar. 2012, ISSN: 0018-9545. DOI: `10.1109/TVT.2012.2183400`.

[NWGG14] J. Nightingale, Q. Wang, C. Grecos, and S. Goma, "Video adaptation for consumer devices: Opportunities and challenges offered by new standards", *IEEE Communications Magazine*, vol. 52, no. 12, pp. 157–163, Dec. 2014, ISSN: 0163-6804. DOI: `10.1109/MCOM.2014.6979968`.

[OBB+14] A. Osseiran *et al.*, "Scenarios for 5G mobile and wireless communications: The vision of the METIS project", *IEEE Communications Magazine*, vol. 52, no. 5, pp. 26–35, May 2014.

[PIT14] K. Poularakis, G. Iosifidis, and L. Tassiulas, "Approximation algorithms for mobile data caching in small cell networks", *IEEE Transactions on Communications*, vol. 62, no. 10, pp. 3665–3677, Oct. 2014.

[Rap09] T. S. Rappaport, *Wireless Communications: Principles and Practice* (Prentice Hall communications engineering and emerging technologies series). Dorling Kindersley, 2009, ISBN: 9788131728826. [Online]. Available: `https://books.google.de/books?id=11qEWkNFFwQC`.

[RLR+18] N. Richerzhagen, P. Lieser, B. Richerzhagen, B. Koldehofe, I. Stavrakakis, and R. Steinmetz, "Change as chance: Transition-enabled monitoring for dynamic networks and environments", in *Proc. of the 14th Wireless On-demand Network systems and Services Conference (WONS)*, IEEE/IFIP, Feb. 2018, pp. 1–8. [Online]. Available: `http://tubiblio.ulb.tu-darmstadt.de/94902/`.

[RRKS17] N. Richerzhagen, B. Richerzhagen, B. Koldehofe, and R. Steinmetz, "Towards transitions between role assignment schemes: Enabling adaptive offloading in challenged networks", in *Proc. of 12th Workshop on Challenged Networks (CHANTS) co-located with 23rd Annual International Conference on Mobile Computing and Networking (MobiCom) (accepted for publication)*, ACM, Oktober 2017, pp. 1–6, ISBN: 978-1-4503-5144-7/17/10. [Online]. Available: `http://tubiblio.ulb.tu-darmstadt.de/89725/`.

[RRL+16] N. Richerzhagen *et al.*, "Exploring transitions in mobile network monitoring in highly dynamic environments", in *Demonstrations of the 41st IEEE Conference on Local Computer Networks (LCN Demonstrations)*, Nov. 2016, pp. 1–3. [Online]. Available: `http://tubiblio.ulb.tu-darmstadt.de/86000/`.

[RRZ+16] B. Richerzhagen, N. Richerzhagen, J. Zobel, S. Schönherr, B. Koldehofe, and R. Steinmetz, "Seamless transitions between filter schemes for location-based mobile applications", in *Proc. 41st IEEE Conference on Local Computer Networks (LCN)*, Nov. 2016, pp. 348–356. [Online]. Available: `http://tubiblio.ulb.tu-darmstadt.de/85999/`.

[RSW05] A. Ramamoorthy, J. Shi, and R. Wesel, "On the Capacity of Network Coding for Random Networks", *IEEE Transactions on Information Theory*, vol. 51, no. 8, pp. 2878–2885, Aug. 2005, ISSN: 0018-9448. DOI: `10.1109/TIT.2005.851725`.

[SE07] Y. Sagduyu and A. Ephremides, "On Joint MAC and Network Coding in Wireless Ad Hoc Networks", *IEEE Transactions on Information Theory*, vol. 53, no. 10, pp. 3697–3713, Oct. 2007, ISSN: 0018-9448. DOI: `10.1109/TIT.2007.904980`.

[SMW07] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007, ISSN: 1051-8215. DOI: `10.1109/TCSVT.2007.905532`.

[SSH+11] Y. Sánchez de la Fuente *et al.*, “iDASH: Improved dynamic adaptive streaming over http using scalable video coding”, in *Proc. Second Annual ACM Conference on Multimedia Systems*, ser. MMSys ’11, San Jose, CA, USA: ACM, 2011, pp. 257–264, ISBN: 978-1-4503-0518-1. DOI: `10.1145/1943552.1943586`. [Online]. Available: `http://doi.acm.org/10.1145/1943552.1943586`.

[SSHE15] D. Stohr, M. Schulz, M. Hollick, and W. Effelsberg, “App and phy in harmony: Demonstrating scalable video streaming supported by flexible physical layer control”, in *2015 IEEE 16th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2015, pp. 1–3. DOI: `10.1109/WoWMoM.2015.7158183`.

[SSS19] K. Spiteri, R. Sitaraman, and D. Sparacio, “From theory to practice: Improving bitrate adaptation in the dash reference player”, *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 15, no. 2s, Jul. 2019, ISSN: 1551-6857. DOI: `10.1145/3336497`. [Online]. Available: `https://doi.org/10.1145/3336497`.

[Sto18] D. Stohr, “User-centric video in the future internet: Qoe in participatory video generation and distribution”, Ph.D. dissertation, Technische Universität Darmstadt, Darmstadt, 2018. [Online]. Available: `http://tuprints.ulb.tu-darmstadt.de/7616/`.

[VCO+07] J. Villalon, P. Cuenca, L. Orozco-Barbosa, Y. Seok, and T. Turletti, “Arsm: A cross-layer auto rate selection multicast mechanism for multi-rate wireless lans”, *IET Communications*, vol. 1, no. 5, pp. 893–902, Oct. 2007, ISSN: 1751-8628. DOI: `10.1049/iet-com:20060249`.

[VSR03] A. Valera, W. K. G. Seah, and S. V. Rao, “Cooperative packet caching and shortest multipath routing in mobile ad hoc networks”, in *Proc. IEEE International Conference on Computer and Communications (INFOCOM)*, vol. 1, 2003, pp. 260–269.

[WCT+14] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. Leung, “Cache in the air: exploiting content caching and delivery techniques for 5G systems”, *IEEE Communications Magazine*, vol. 52, pp. 131–139, 2014.

[Wil16] S. Wilk, “Quality-aware content adaptation in digital video streaming”, Ph.D. dissertation, Technische Universität Darmstadt, Darmstadt, 2016. [Online]. Available: `http://tuprints.ulb.tu-darmstadt.de/5819/`.

[XNI+15] L. Xiang, D. W. K. Ng, T. Islam, R. Schober, and V. W. S. Wong, “Cross-layer optimization of fast video delivery in cache-enabled relaying networks”, in *IEEE Global Communications Conference (GLOBECOM)*, 2015, pp. 1–7.

[YA14] Y. Ye and P. Andrivon, “The scalable extensions of hevc for ultra-high-definition video delivery”, *IEEE MultiMedia*, vol. 21, no. 3, pp. 58–64, Jul. 2014, ISSN: 1070-986X. DOI: `10.1109/MMUL.2014.47`.

[YLYL06] J. Yuan, Z. Li, W. Yu, and B. Li, “A Cross-Layer Optimization Framework for Multihop Multicast in Wireless Mesh Networks”, *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 11, pp. 2092–2103, Nov. 2006, ISSN: 0733-8716. DOI: `10.1109/JSAC.2006.881617`.

[ZAP+07] X. Zhu, P. Agrawal, J. Pal Singh, T. Alpcan, and B. Girod, “Rate allocation for multi-user video streaming over heterogenous access networks”, in *Proc. ACM 15th International Conference on Multimedia*, Augsburg, Germany: ACM, 2007, pp. 37–46, ISBN: 978-1-59593-702-5.

[ZGL+14] M. Zhao, X. Gong, J. Liang, W. Wang, X. Que, and S. Cheng, “Scheduling and resource allocation for wireless dynamic adaptive streaming of scalable videos over http”, in *IEEE International Conference on Communications (ICC), 2014*, Jun. 2014, pp. 1681–1686. DOI: `10.1109/ICC.2014.6883564`.

[ZGZ+09] L. Zhou, B. Geller, B. Zheng, A. Wei, and J. Cui, “System scheduling for multi-description video streaming over wireless multi-hop networks”, *IEEE Transactions on Broadcasting*, vol. 55, no. 4, pp. 731–741, Dec. 2009, ISSN: 0018-9316. DOI: `10.1109/TBC.2009.2032795`.

[ZYZ+06] J. Zhao, F. Yang, Q. Zhang, Z. Zhang, and F. Zhang, “Lion: Layered overlay multicast with network coding”, *IEEE Transactions on Multimedia*, vol. 8, no. 5, pp. 1021–1032, Oct. 2006, ISSN: 1520-9210. DOI: `10.1109/TMM.2006.879847`.

# Own Publications

[FAWK15] M. Fasil, H. Al-Shatri, S. Wilk, and A. Klein, "Application-Aware Cross-Layer framework: Video content distribution in wireless multihop networks", in *Proc. IEEE 26th Annual International Symposium on Personal, Indoor and Mobile Radio Communications - (PIMRC)*, Hong Kong, P.R. China, Aug. 2015, pp. 1088–1093.

[FAWK16] M. Fasil, H. Al-Shatri, S. Wilk, and A. Klein, "A network-centric view on dash in wireless multihop networks", in *Proc. IEEE 84th Vehicular Technology Conference (VTC-Fall)*, Sep. 2016, pp. 1–6. DOI: `10.1109/VTCFall.2016.7880861`.

[FKK14] M. Fasil, A. Kuehne, and A. Klein, "Node virtualization and network coding: Optimizing data rate in wireless multicast", in *Proc. International Symposium on Wireless Communications Systems (ISWCS)*, Aug. 2014, pp. 573–578. DOI: `10.1109/ISWCS.2014.6933419`.

[FMAK17] M. Fasil, S. Mueller, H. Al-Shatri, and A. Klein, "Exploiting caching and cross-layer transitions for content delivery in wireless multihop networks", in *15th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, May 2017, pp. 1–7. DOI: `10.23919/WIOPT.2017.7959877`.

# Supervised Student Theses

| Name | Title of the thesis | Thesis type | Date |
|---|---|---|---|
| Shehata, Mohamed | Routing in Wireless Multi-Hop Networks: Evaluation of Anypath, Multipath and Corridor Based Routing | Bachelor Thesis | 07/2013 |
| Ivanova, Izabela | Power Allocation in Wireless Multi-Hop Networks | Bachelor Thesis | 08/2013 |
| Zeraj, Alba | Multi-Objective Rate Optimization in Wireless Multihop Networks | Master Thesis | 10/2015 |
| Orenbakh, Mikhail | A Low-Complexity Scheduler for Physical Layer Multicast in Wireless Multihop Networks | Master Thesis | 12/2015 |
| Kumar, Seema | Integration of Network Coding and Topology Control | Master Thesis | 01/2016 |
| Striffler, Tobias | Analysing Caching Policies in a Cross-Layer Content Delivery Framework for Wireless Multihop Networks | Master Thesis | 02/2017 |

# Funding Acknowledgment

This work has been funded by the German Research Foundation (DFG) within Collaborative Research Center (CRC) 1053 “MAKI – Multi-Mechanisms Adaptation for the Future Internet” as part of project C01.

# Lebenslauf

| | |
|---|---|
| **Schulausbildung** | |
| 1995-2001 | Heinrich-Heine Schule in Dreieich<br>Schulabschluß: Mittlere Reife |
| 2001-2004 | Weibelfeld Schule Dreieich<br>Schulabschluß: Abitur |
| **Zivildienst** | |
| 07/2004-03/2005 | Zivildienstleistender im AWO Jugendheim, Dreieich |
| **Studium** | |
| 2005-2011 | Studium der Elektro- und Informationstechnik an der Technischen Universität Darmstadt<br>Studienabschluß: Diplom |
| **Berufstätigkeit** | |
| 07/2012-06/2017 | wissenschaftlicher Mitarbeiter am<br>Fachgebiet Kommunikationstechnik,<br>Institut für Nachrichtentechnik,<br>Technische Universität Darmstadt |
| 02/2018-03/2019 | Software Consultant bei<br>NEO Analytics,<br>Frankfurt am Main |
| 04/2019-04/2020 | Technolgy Consultant bei<br>Intelligent Data Analytics,<br>Frankfurt am Main |
| seit 06/2020 | Cloud Spezialist AWS bei<br>DB Systel GmbH,<br>Frankfurt am Main |

Darmstadt, 11.03.2021

Mousie Fasil

## Erklärungen laut Promotionsordnung

### §8 Abs. 1 lit. c PromO

Ich versichere hiermit, dass die elektronische Version meiner Dissertation mit der schriftlichen Version übereinstimmt.

### §8 Abs. 1 lit. d PromO

Ich versichere hiermit, dass zu einem vorherigen Zeitpunkt noch keine Promotion versucht wurde. In diesem Fall sind nähere Angaben über Zeitpunkt, Hochschule, Dissertationsthema und Ergebnis dieses Versuchs mitzuteilen.

### §9 Abs. 1 PromO

Ich versichere hiermit, dass die vorliegende Dissertation selbstständig und nur unter Verwendung der angegebenen Quellen verfasst wurde.

### §9 Abs. 2 PromO

Die Arbeit hat bisher noch nicht zu Prüfungszwecken gedient.

Datum und Unterschrift