Thorsten Biermann

Dealing with Backhaul Network Limitations in Coordinated Multi-Point Deployments

Dissertation

accepted by the

Faculty of Electrical Engineering,
Computer Science, and Mathematics

in partial fulfillment of the requirements for the degree of

Doctor rerum naturalium (Dr. rer. nat.)

Munich, July 2012

Referees:

Professor Dr. Holger Karl, University of Paderborn, Germany
Dr.-Ing. habil. Slawomir Stanczak, Technical University Berlin, Germany

Additional committee members:

Junior-Professor Dr. Christian Plessl, University of Paderborn, Germany
Professor Dr. Franz Josef Rammig, University of Paderborn, Germany
Professor Dr. Heike Wehrheim, University of Paderborn, Germany

Submission: July 25, 2012
Examination: October 30, 2012

Abstract

Future mobile access networks, like Long Term Evolution-Advanced (LTE-A), target wireless speeds of 1 Gbit/s per cell. Such high data rates require to increase the cell density by extending today's 3-sector Base Station (BS) deployments to host 6 sectors, and adding additional small pico and femto cells to cover hot spot locations. The resulting high cell density, however, also increases interference, which limits the performance of the mobile network and prevents to reach the targeted data rates, especially for User Equipments (UEs) located at the cell borders. Therefore, interference management has become the key factor to determine the overall wireless system performance.

The most promising method for managing interference is to coordinate neighbor cells. This approach is called Coordinated Multi-Point (CoMP) transmission/reception. Multiple BSs form a cluster and cooperate by exchanging, e.g., signaling and/or UE data. To enable CoMP, the backhaul network must fulfill stringent capacity and latency requirements, which are on the order of several Gbit/s backhauling capacity per BS and round-trip latencies between BSs down to 1 ms.

This thesis addresses CoMP's feasibility from the backhaul network perspective. It first evaluates whether current and future backhaul infrastructures can support certain CoMP techniques. It turned out that even standard future optical technologies, like Wavelength Division Multiplexing (WDM) Passive Optical Networks (PONs), do not allow CoMP in a satisfactory way. Therefore, we investigate mechanisms to deal with limitations caused by the backhaul shortcomings and mechanisms that improve CoMP feasibility in general and in particular in PON environments. The proposed mechanisms are combined to an overall system architecture. Finally, the influence of the metro-wide network topology and architecture is evaluated and migration guidelines are provided.

Based on the gained insights, several conclusions can be drawn. First, CoMP's feasibility is strongly limited, even with standard future optical backhauling technologies. This requires a cross-domain approach, which decides when, where, and how to cooperate based on wireless channel properties *and* on the backhaul network status. The presented techniques allow a ubiquitous CoMP deployment for BSs connected to the same backhaul access network branch. The presented topologies and architectures extend CoMP support to a metro-wide scale while even reusing existing metro ring deployments.

Zusammenfassung

Zukünftige mobile Netze, wie Long Term Evolution-Advanced (LTE-A), planen drahtlose Geschwindigkeiten von 1 Gbit/s pro Zelle. Solche hohen Datenraten erfordern eine erhöhte Zelldichte, beispielsweise durch die Erweiterung heutiger 3-Sektor-Basisstationen (BS) auf 6 Sektoren und durch zusätzliche, kleine Pico- und Femto-Zellen, um lokale Lastmaxima abzudecken. Die daraus resultierende hohe Zelldichte erhöht jedoch auch die Interferenz, was wiederum die mögliche Datenrate im System beschränkt, insbesondere für Benutzer, die sich an Zellgrenzen befinden. Daher ist Interferenz-Management zur Schlüsseltechnik geworden, um die gewünschten Datenraten zu erreichen.

Die vielversprechendste Technik für Interferenz-Management is das Koordinieren benachbarter Zellen. Diese Methode wird als Coordinated Multi-Point (CoMP) Transmission/Reception bezeichnet. Mehrere BS formen eine Gruppe und kooperieren, indem sie beispielsweise Signalisierungs- und/oder Benutzerdaten austauschen. Um CoMP zu ermöglichen, muss das Backhaul-Netzwerk strikte Kapazitäts- und Latenzanforderungen, in der Größenordnung von mehreren Gbit/s Datenrate pro BS und einer Umlaufzeit zwischen BS von bis zu 1 ms, erfüllen.

Diese Arbeit betrachtet die Realisierbarkeit von CoMP aus der Backhaul-Perspektive. Zunächst wurde untersucht, inwiefern heutige und zukünftige Backhaul-Infrastrukturen bestimmte CoMP-Techniken unterstützen können. Es stellte sich heraus, dass sogar zukünftige optische Technologien, wie Wavelength Division Multiplexing (WDM) Passive Optical Networks (PONs), CoMP nicht zufriedenstellend ermöglichen. Daher haben wir Techniken entwickelt, um mit Einschränkungen umzugehen und um die Realisierbarkeit von CoMP allgemein, und insbesondere in PON-Umgebungen, zu verbessern. Die entwickelten Einzeltechniken wurden zu einer Gesamtarchitektur kombiniert. Zusätzlich wurde der Einfluss der Topologie und Architektur des Metro-Netzes untersucht und Richtlinien zur Migration bisheriger Netze erarbeitet.

Die gewonnenen Einblicke liefern mehrere Schlussfolgerungen. CoMP ist, sogar unter Verwendung zukünftiger, optischer Backhaul-Technologien, stark eingeschränkt. Daher ist es notwendig, bei der Entscheidung wann, wo und wie BS kooperieren, die drahtlosen Kanaleigenschaften *und* den Zustand des Backhaul-Netzes zu berücksichtigen. Die vorgestellten Techniken zur Verbesserung der Realisierbarkeit ermöglichen CoMPs

flächendeckenden Einsatz, solange BS über den gleichen Backhaul-Netzwerk-Strang verbunden sind. Sobald das Metro-Netzwerk involviert ist, sind weitere Vorkehrungen nötig. Die vorgestellten Topologien und Architekturen ermöglichen CoMP metro-weit einzusetzen und verwenden dabei existierende Glasfaser-Installationen wieder.

Acknowledgment

First of all, I would like to thank Holger Karl for supervising me during my work that led to this thesis. He taught me how to solve problems, was always open for interesting discussions, had the patience to listen to my (probably sometimes stupid) proposals, and was always available when I needed advice. I am especially thankful for his steady support after I left Paderborn and continued research in Munich.

I also would like to thank Slawomir Stanczak for being available as second examiner of this thesis. The feedback of him and his group helped to improve this work.

On my way to complete this thesis I got support from many brilliant colleagues. First, thanks to my former colleagues from the Research Group Computer Networks at the University of Paderborn. In particular, Christian Dannewitz who shared his office with me and was always available for discussions of any type. Thanks for the great time! Working together with Martin Dräxler for many joint experiments and simulations was always a pleasure. Thanks for this productive collaboration! In addition, thanks a lot to Tanja Langen and Hajo Kraus for the great support in solving the administrative hassle at universities and for providing the right equipment and technical support.

I would like to thank Jörg Widmer for bringing me to DOCOMO Euro-Labs. Working together with so many professional people in such a friendly atmosphere made work feel like fun. The joint project together with Luca Scalia, Changsoon Choi, Kazuyuki Kozu, and Wolfgang Kellerer was very successful. Without them and their support finishing this thesis would not have been half as pleasurable as it was. Thanks for the fantastic last two years! In addition, I would like to thank various office mates for having tolerated me. There are Hauke Holtkamp, Serkan Uygungelen, Bo Fu, Jelena Frtunikj, Sandra Herker, Marwa El Hefnawy, and Emmanuel Ternon. Together with many other colleagues they made my time at Euro-Labs unforgettable.

Finally, I am feeling deeply grateful for my family. You allowed me to get where I am. Without your continuous support, all this would not have been possible. I am also very thankful for Flo being there when I need her, for her patience, and her love.

Contents

Chapter 1

Introduction

Contents

1.1 Current trends in mobile networks

The spread of smartphones has changed the habits of mobile users, who are nowadays more and more interested in multimedia content and social network services. Such services require much more capacity compared to today's mobile user behavior. According to predictions, mobile data traffic is expected to grow at a Compound Annual Growth Rate (CAGR) of 95 %. This results in a 28-fold increase between 2010 and 2015, consisting of approximately 58 % video content in 2015 [1].

To address the high data rate requirements of such services, targets for future cellular networks, like 3GPP's Long Term Evolution-Advanced (LTE-A), have been defined [2]. Wireless data rates of 1 Gbit/s per cell are expected. In addition, today's typical 3-sector Base Station (BS) deployments are going to be extended to host 6 sectors per BS. Additional small pico and femto cells for covering hot spot locations further increase the cell density and decrease the cell size to provide the required capacity.

As result of the high BS density, interference increases in the network as systems like LTE-A operate at a frequency reuse factor of 1, i.e., all BSs use the same spectrum. The increased interference in turn limits the performance of the mobile network and prevents the targeted data rates, especially for UEs located at the border between multiple cells. But also for UEs in the center of small cells, interference is present due to the overlap between macro cells and pico or femto cells.

To achieve the targeted performance of LTE-A, interference management has become a key technique. In many existing wireless communication systems, interference is

dealt with by making transmissions to/from UEs orthogonal in time or frequency or by increasing transmission power and treating each other's interference as noise. Recently, the paradigm has shifted to focus on how to more intelligently reduce or even exploit interference through the use of BS cooperation techniques. Such a system, where multiple cells (or, equivalently, sectors) are coordinated to reduce destructive interference among each other, is called CoMP transmission/reception system. In a CoMP system, multiple BSs form a cluster and cooperate by exchanging, e.g., signaling or UE data. CoMP systems have proven to be a very effective solution for managing interference in current and upcoming wireless systems [3, 4, 5].

There are many different ways how BSs actually can cooperate with each other. Referring to current 3GPP activities, Joint Scheduling (JS) [6], Joint Beam-Forming (JB) [7] and Joint Processing (JP) [8] are techniques which are under discussion for LTE-A. These techniques achieve different throughput gains [3, 4, 5] and have different impact on the wireline network infrastructure of the mobile network.

One BS usually operates multiple cells/sectors. In case CoMP is applied only among cells that belong to the same BS, no information has to be exchanged via the network interconnecting the BSs. However, when CoMP spans over cells belonging to different BSs, which easily happens in the general case, the network between the BSs is involved. This part of the network, which connects the BSs to the core network, i.e., interconnects and transports information between cooperating BSs, is called *backhaul network* and plays an important role in the feasibility of BS cooperation. To enable CoMP, the backhaul network part of the cellular system must fulfill all capacity and latency requirements imposed by the used cooperation technique, which are on the order of several Gbit/s backhauling capacity per cell and round-trip latencies between BSs of up to 1 ms.

Definition (Backhaul network)

The backhaul network is the network infrastructure that interconnects BSs among each other and connects them to the core network of the network operator. The backhaul network can be implemented using (combinations of) various technologies, like copper, fiber optics, and microwave links, and covers those parts of the network infrastructure that are often referred to as access and metro network.

There has been a lot of research in the area of implementing and evaluating the performance of different cooperation techniques, like joint signal processing [3, 4, 5] and inter-cell coordination [9, 10], from the wireless point of view. The results show how to integrate cooperation techniques into the wireless systems, demonstrate the possible performance gains, and partly scratch the surface of how to design an overall network architecture that incorporates cooperation techniques. Other work covers (i) the synchronization of the components involved in the cooperation schemes [11], (ii) the number of BSs that need to cooperate to achieve the desired gains [12], (iii) the management of inter-cluster interference [13], e.g., by introducing overlapping cluster configurations,

and (iv) the efficient collection of Channel State Information (CSI) [14]. In the context of the backhaul network influence, several evaluations show which requirements have to be fulfilled and how limited backhaul networks influence the efficiency of CoMP [15]. In turn, improvements for the wireless procedure to reduce the backhaul network requirements at the cost of a worse performance for the UEs are presented.

All these existing evaluations lack details, which prevent clear statements of the applicability of CoMP in real networks. Whereas the upgrade path for the wireless side, i.e., how to change the BSs to allow CoMP, is quite straightforward with the mechanisms that have been proposed so far, a proper idea on how to deal with the very strict requirements on the wireline network side is not clear. The main reason for this is that previous research work only took into account an abstracted, i.e., very simplified version of the backhaul network. Often, the whole network is modeled without taking into account individual links but just assuming a certain property, like a backhaul failure probability or a certain delay model. In reality, however, backhaul network infrastructures are very diverse. The used technologies range from low-capacity copper links over wireless microwave links up to optical links, all having different properties, like data rates and link latency, and hence different influence on the feasibility of CoMP.

1.2 Contribution

I focus on backhaul networks for CoMP. Instead of abstracting the backhaul network, as it has been done in related research work, I consider individual links between BSs as well as intermediate nodes, like routers, and their influence on the data exchange. This gives detailed insights on how different backhaul infrastructures influence CoMP.

Based on the outcome of the initial feasibility studies, I have developed mechanisms to deal with the observed shortcomings of the backhaul network. This allows to exploit CoMP techniques to an extent that the backhaul network supports and avoids arbitrary use of CoMP despite insufficient network capabilities. Furthermore, I developed approaches for improving CoMP feasibility in general and in particular in networks that use PON technologies. PONs are a promising technology for future backhaul networks as they can provide high capacity and are easy to deploy in the field.

Finally, I elaborate on how to design a backhaul network topology up to the metro scale and which technology to use to provide ubiquitous CoMP support. This covers clean-slate approaches as well as reusing existing infrastructures to a certain extent.

The results of my research work have been published in several journal and conference papers. In addition, various patents have been filed. The following lists give an overview of the publications and patents that have been published in the course of this thesis:

Journal papers

1. T. Biermann, L. Scalia, C. Choi, H. Karl, and W. Kellerer, "CoMP clustering and backhaul limitations in cooperative cellular mobile access networks," *Elsevier Pervasive and Mobile Computing*, 2012.

Conference papers

2. K. Miller, T. Biermann, H. Woesner, and H. Karl, "Network coding in passive optical networks," in *Proc. IEEE International Symposium on Network Coding (NetCod)*, IEEE, June 2010.

3. T. Biermann, L. Scalia, J. Widmer, and H. Karl, "Backhaul design and controller placement for cooperative mobile access networks," in *Proc. IEEE Vehicular Technology Conference (VTC Spring)*, IEEE, May 2011.

4. T. Biermann, L. Scalia, C. Choi, H. Karl, and W. Kellerer, "Backhaul network pre-clustering in cooperative cellular mobile access networks," in *Proc. IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, IEEE, June 2011.

5. T. Biermann, L. Scalia, C. Choi, H. Karl, and W. Kellerer, "Improving CoMP cluster feasibility by dynamic serving base station reassignment," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, IEEE, Sept. 2011.

6. C. Choi, L. Scalia, T. Biermann, and S. Mizuta, "Coordinated multipoint multiuser-MIMO transmissions over backhaul-constrained mobile access networks," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, IEEE, Sept. 2011.

7. T. Biermann, L. Scalia, and H. Karl, "Designing optical metro and access networks for future cooperative cellular systems," in *Proc. ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pp. 265–274, ACM, Oct. 2011.

8. C. Choi, Q. Wei, T. Biermann, and L. Scalia, "Mobile WDM backhaul access networks with physical inter-base-station links for coordinated multipoint transmission/reception systems," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, IEEE, Dec. 2011.

9. W. Kellerer, W. Kiess, L. Scalia, T. Biermann, C. Choi, and K. Kozu, "Novel cellular optical access network and convergence with FTTH," in *Proc. Optical Fiber Communication Conference and Exposition (OFC) and The National Fiber Optic Engineers Conference (NFOEC)*, Mar. 2012. Invited paper.

10. M. Dräxler, T. Biermann, H. Karl, and W. Kellerer, "Cooperating base station set selection and network reconfiguration in limited backhaul networks," in *Proc. IEEE Personal Indoor Mobile Radio Communications (PIMRC)*, 2012.

Patents

11. T. Biermann, L. Scalia, C. Choi, K. Kozu, and W. Kellerer, "Apparatus and method for determining a core network configuration of a wireless communication system," 2010.

12. L. Scalia, T. Biermann, C. Choi, K. Kozu, and W. Kellerer, "Apparatus and method for controlling a node of a wireless communication system," 2010.

13. C. Choi, Q. Wei, and T. Biermann, "Communication systems and method for directly transmitting signals between nodes of a communication system, node and optical multiplexer/demultiplexer device," 2010.

14. L. Scalia, T. Biermann, and C. Choi, "Apparatus and method for determining a control unit using feasibility requests and feasibility response," 2011.

15. C. Choi, L. Scalia, T. Biermann, and S. Mizuta, "Method for coordinated multipoint communication in wireless communication network, wireless communication system, base station and network controller," 2011.

16. C. Choi, T. Biermann, and Q. Wei, "Central node for a communication system, wireless communication system, optical multiplexer/demultiplexer device, and method for transmitting data to one or more groups of nodes in a communication system," 2011.

17. L. Scalia, T. Biermann, and C. Choi, "Apparatus and method for determining a cluster of base stations," 2011.

Submitted

18. T. Biermann, L. Scalia, C. Choi, H. Karl, and W. Kellerer, "On the feasibility of coordinated multi-point base station clusters in cellular networks," *IEEE Communications Magazine*, 2012. Submitted for publication.

1.3 Structure of this thesis

The remainder of this thesis is structured as follows:

Chapter 2: CoMP background, its requirements, and related work
This chapter introduces different CoMP techniques that are currently under discussion in the 3GPP community. It provides a deep discussion of Joint Transmission (JT) and its requirements that have to be fulfilled by the backhaul network infrastructure. Existing work on the backhaul network influence on CoMP feasibility are addressed. Parts of this chapter are contained in all publications mentioned earlier, although some papers provide more details [16, 24].

Chapter 3: Static clustering
The third chapter deals with the first way of deciding which BSs cooperate to jointly serve UEs in a CoMP system: the sets of cooperating BSs are chosen statically. Two approaches for finding optimal and quasi-optimal clusters for different input complexities are proposed. The content of this chapter has been published in two papers [18, 25].

Chapter 4: Dynamic clustering and its feasibility
To increase CoMP benefits, cooperating BS clusters can be selected dynamically based on the individual wireless channel conditions between the UEs and the BSs. This chapter first discusses requirements on the dynamic clusters, which come from the wireless side, and then evaluates the feasibility of dynamic clustering under realistic backhaul network scenarios. The results show that the clusters that are feasible according to the backhaul network capabilities are in most cases different from the ones desired according to the wireless channel properties. Addressing this mismatch, an approach is presented to deal with insufficient backhaul network capabilities when CoMP transmission or reception is required. The topics of this chapter have been published in two papers [16, 19].

Chapter 5: Improving CoMP cluster feasibility
As the backhaul network requirements of CoMP cannot be fulfilled in many cases, techniques are required to improve feasibility in such scenarios. Therefore, several techniques are presented in this chapter to mitigate backhaul network limitations in general and in networks that use PONs, a promising technology for future backhaul network deployments, to connect BSs to the operator's core network. Tech techniques have been published in several papers [17, 20, 23, 25].

Chapter 6: Deploying CoMP on metro scale
After having introduced several techniques for improving CoMP feasibility on the access network level in the previous chapters, this chapter first shows the necessity to also think about the feasibility on the metro network level. Thereafter, different metro network deployment approaches are proposed and evaluated for their ability to support CoMP. There is one published paper that deals with the content of this chapter [22].

Chapter 7: Combining clustering and feasibility improvement techniques
In this chapter, the individual techniques proposed in Chapter 4 and Chapter 5 are combined into an overall system.

Chapter 8: Conclusions and future research
The final chapter concludes the work presented in the previous chapters and gives an outlook on further work items.

Chapter 2

CoMP background, its requirements, and related work

Contents

The paradigm in cellular communication systems has shifted from interference avoidance among neighboring cells to intelligently exploiting the knowledge and/or the structure of interference in a certain area by exploiting Base Station (BS) cooperation techniques. In a Coordinated Multi-Point (CoMP) transmission/reception system, multiple BSs form a cluster and cooperate by exchanging signaling and/or UE data via the core and backhaul network. Such systems have proven to be very effective for interference management.

2.1 Overview of CoMP techniques

Following the 3GPP LTE-A terminology, CoMP techniques are grouped into two main categories, depending on whether UE data other than signaling is shared among the cooperating BSs:

- Joint Processing (JP): In the downlink, the UE receives either a Joint Transmission (JT), where multiple BSs simultaneously send on the same physical resources to

create constructive interference at the UE, or the UE performs a Dynamic Cell Selection (DCS) to select the best BS for the current data transmission. In the uplink, the UE data transmission is received by multiple BSs and sent from each cooperating BS to a central point (e.g., the serving BS, which is the BS with the best wireless channel conditions to the UE, or a central processing unit located in the network) where the received versions of the UE data are jointly decoded. In a JP system, UE data has to be shared among cooperating BSs.

Definition (Serving BS)

The serving BS of a certain UE is the BS that has the best wireless channel conditions to the UE compared to all other surrounding BSs. It is the BS that usually serves the UE in case no cooperation is used.

- Joint Scheduling (JS)/Joint Beam-Forming (JB): In the downlink case, UEs receive data transmissions only from one cell but scheduling and beamforming decisions are coordinated among the cells in the CoMP cluster. In the uplink, UEs of neighboring cells in a CoMP cluster are scheduled such that interference is reduced. UE data is *not* shared among cooperating BSs.

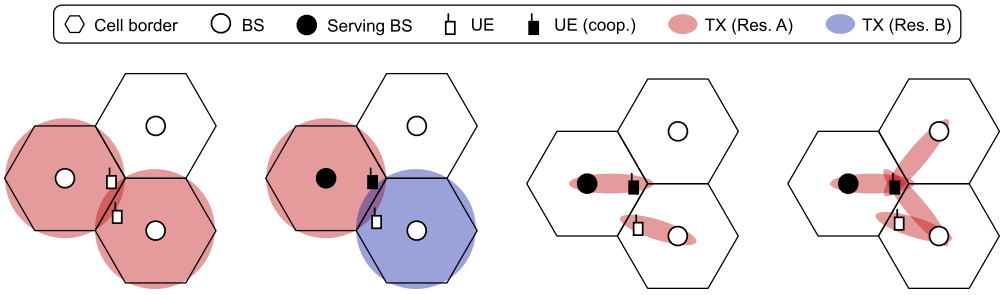
An overview of how the different CoMP techniques work and how they influence the transmission to/from the UEs is given in Figure 2.1 on the facing page. Note that JT is often combined with JB to further increase performance. Although this combining is not necessarily required, such an approach is shown in Figure 2.1d on the next page.

It has been shown that JT is the most promising CoMP approach as it can improve the average cell throughput up to 10 % when using Single-User MIMO (SU-MIMO) and up to 60 % when using Multi-User MIMO (MU-MIMO). Cell-edge UEs' performance improves for SU-MIMO up to 20 % and for MU-MIMO up to 50 % [34].

Besides the big gains that can be expected from JT, the drawback is that it is very demanding in terms of backhaul network capabilities. The backhaul network has to provide high capacity and has to have low latency. These trade-offs are further elaborated in the following.

2.2 Joint transmission and its implementation issues

The most promising CoMP scheme in the downlink is Multi-User MIMO (MU-MIMO) Joint Transmission (JT) [35]. JT in the downlink means that UE data is sent from multiple BSs at the same time using the same physical resources such that the signals interfere constructively at the UEs, thus maximizing the experienced Signal To Interference and Noise Ratio (SINR). This is achieved by individually precoding the UE data at



(a) No CoMP: BSs transmit on the same physical resources which causes interference.

(b) JS: BSs coordinate scheduling decisions to use different physical resources for cell-edge UEs.

(c) JB: BSs beamform to their respective cell-edge UEs to minimize interference. JS can be applied on top of this in order to maximize the use of physical resources.

(d) JT: BSs transmit on the same physical resources to create constructive interference at the UE.

Figure 2.1: Different CoMP techniques in the downlink transmission from BSs to UEs. The different colors of the wireless transmissions indicate different physical resources.

the cooperating BSs based on measured CSI from the UEs. This CSI describes the current wireless channel between the BSs and the UE, which is a very dynamic property. Depending on the mobility of the UEs, CSI loses its validity after a few milliseconds (typically after 1 to 10 ms).

To implement MU-MIMO JT, collected CSI, signaling, and UE data needs to be exchanged between cooperating BSs. Therefore, a considerable amount of capacity needs to be provided in the control plane and the transport plane of the mobile backhaul network infrastructure. Furthermore, the timely availability of the CSI and UE data at the cooperating BSs has a significant impact on the JT performance. Therefore, the mobile backhaul network infrastructure also has to guarantee that the cooperating BSs are able to exchange data within 1 ms in the worst case. Low-mobility scenarios relax this time constraint to multiple milliseconds.

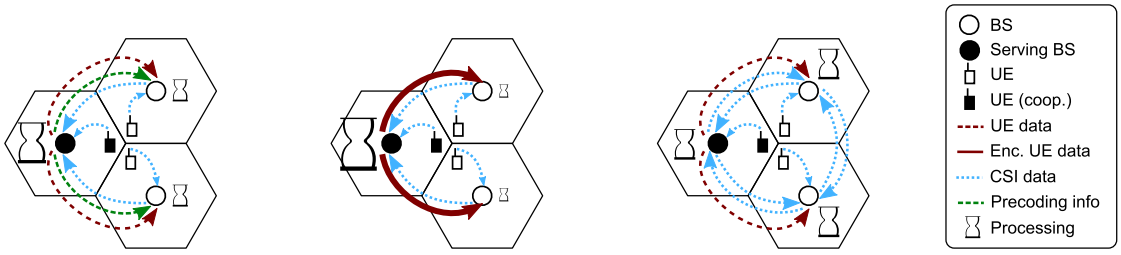
2.2.1 Implementation schemes

There are different ways of implementing a JT MU-MIMO system, each having different advantages and disadvantages resulting from the different ways of exchanging and processing UE data, CSI and precoding information. Such schemes can be divided into two categories:

- Distributed JT: All BSs in the cluster share their CSI and locally compute the precoding information using the same algorithm. In addition, raw UE data is shared from the serving BS, which is the BS with the best wireless channel conditions to an UE, or directly from a central entity located within the network.

- Centralized JT: CSI is aggregated at a central controller, like the serving BS of a UE. At this location, the precoding information is calculated for all BSs in the cluster and it is then distributed to all BSs in the cluster. Depending on where the UE data is eventually encoded using the centrally calculated precoding information, the approach is called fully centralized (encoding also takes place at the central controller and the encoded data is sent to the cooperating BSs, e.g., in form of IQ samples) or semi-centralized (raw UE data is encoded at each BS individually). The fully centralized approach is less convenient than the semi-centralized one since transferring the encoded data needs 10-20 times higher data rates than the transmission of the raw UE data.

The implementation approaches and the corresponding data and CSI and/or precoding information signaling exchange among the BSs are illustrated in Figure 2.2.



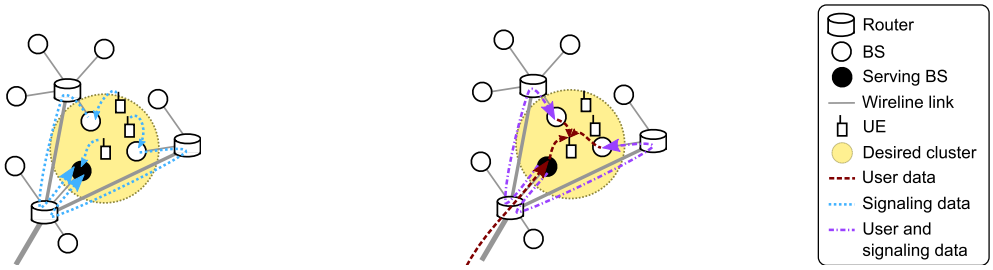
(a) Semi-centralized JT (b) Fully centralized JT (c) Distributed JT

Figure 2.2: Different JT implementation schemes assuming MU-MIMO. The lefthand cell represents the serving BS with the UE that needs cooperation. The two righthand cells represent the cooperating BSs. The size of the processing hourglass indicates the required amount of processing; the arrow line weight represents the amount of required capacity for the data exchange.

Figure 2.3 on the next page illustrates a downlink MU-MIMO JP scenario in detail by giving a chronological sequence of the individual actions that occur. The illustration assumes a semi-centralized JP implementation where each BS locally encodes the raw UE data based on the precoding information that has been calculated at a joint controller.

In the first phase (Figure 2.3a on the facing page), each cooperating BS collects the CSI measured by the UEs in its cell and sends it to a central point that coordinates the JP procedure, e.g., the serving BS of the UE that needs cooperation. I call the node that realizes this role the *CoMP controller* in the remainder of this thesis. This CSI exchange is needed to permit MU-MIMO processing when calculating the precoding information. In a second phase (Figure 2.3b on the next page), the precoding information is sent together with the raw user data to each cooperating BS where the simultaneous wireless transmission eventually takes place.

To understand the impact of MU-MIMO JT on the backhaul network design, I provide a more detailed description on the individual steps of the MU-MIMO JT procedure in the following subsections.



(a) Phase 1: UEs collect CSI and send it to the BSs. They forward it to the serving BS for processing.

(b) Phase 2: Serving BS forwards user data to cluster members (incl. scheduling, precoding info).

Figure 2.3: JP and corresponding data exchange.

2.2.2 Base station clustering

Ideally, a fully coordinated network could guarantee interference-free operation. In practice, however, this would require to exchange and process an enormous amount of UE data and signaling information. Therefore, it is reasonable to assume that only a limited number of BSs take part in the cooperative transmission. The question is *which BSs shall form a cooperative cluster to achieve reasonable performance gains*? A detailed overview of different clustering approaches can be found in [36].

There are two basic categories how to cluster BSs – static and dynamic clustering:

- Static clustering: In this case, clusters are chosen once, usually based on geographic criteria, and do not change over time. Hence, the drawback of this approach is that clusters are not optimized for individual UEs, i.e., they do not take into account current wireless channel conditions.

- Dynamic clustering: Here, the cooperating BS set changes over time such that the *best* BS set for a UE is selected. This selection is usually based on wireless channel measurements the UEs provide periodically to the BSs for hand-over purposes [37, 38], or based on more detailed CSI. The disadvantage of dynamic clustering is the additional overhead for deciding which cluster fits best for a certain UE.

It has been shown that CoMP performance is worse when using static clustering compared to dynamic clustering. However, when using more complex multi-layer overlapping cluster setups, attainable gains get closer to the dynamic approaches which exploit UE feedback [39].

To implement dynamic clustering schemes, the 3GPP standards provide mechanisms to access information required for selecting suitable BSs for cooperation. In this direction, Marsch et al. [40] suggest a dynamic clustering scheme leveraging the Measurement Report Messages (MRM) defined in the Long Term Evolution (LTE) specifications. The averaged Reference Signal Received Power (RSRP) values of the BSs, as reported by the UEs, are compared to a threshold to determine which of these BSs should cooperate.

In general, from the wireless system performance view it is better to have many BSs cooperate with each other as interference among cooperating BSs is reduced. Hence, the larger the cooperating set of BSs, the higher the expected wireless system throughput.

2.2.3 In-band and out-of-band signaling

The requirements to be fulfilled by the backhaul network are also influenced by the way *how* information is exchanged. A preliminary discussion on this issue can be found in [41]. Basically, there exist two approaches to exchange UE data and signaling between cooperating BSs. Considering the semi-centralized JT scheme (Section 2.2.1), once CSI is collected at the central processing unit co-located with the serving BS of the cooperatively served UE, the computed precoding information can be transported either together with the UE data (in-band signaling through piggybacking) or via a separate control plane (out-of-band signaling). The trade-offs arising from the two signaling schemes are quite different.

In the case of in-band signaling, all latency requirements arising from the limited CSI validity (which affect the temporal validity of the derived precoding information) also apply to UE data, thus imposing all strict latency demands on the transport plane.

On the other hand, in case of out-of-band signaling, the signaling information is separately carried via the control plane. As the relationship between UE data and precoding information, which is given inherently when piggybacking both things, is lost, additional information is needed in the out-of-band case to set up this relationship. Hence, the signaling information has to contain both the precoding information and an identifier of the UE data packet to which it has to be applied. To avoid having the same strict latency demands of the control plane also for the transport plane, UE data needs to be shared in advance between the controller and the cooperating BSs. This blindly increases the load on the backhaul network, regardless whether the transmission will be eventually done cooperatively or not, which further jeopardizes the limited backhaul network capacity. Another drawback is that the out-of-band signaling scheme requires higher complexity than in-band signaling: the cooperating BSs have to *search* and *identify* the packets to which a certain precoding information has to be applied in their buffers.

The choice to use in-band or out-of-band signaling does not influence the needed synchronization among cooperating BSs. This is caused by the fact that the signaling scheme does not change the actual joint transmission. Hence, the BSs need to be synchronized tightly to achieve interference reduction.

In summary, separating UE data and signaling of precoding information and UE data packet identification information trades off reduced delay requirements in the backhaul network transport plane with increased capacity requirements as well as increased buffer space and processing at the BSs.

Similar considerations apply to the distributed JT scheme. In this case, as precoding information is computed locally, the only signaling that needs to be shared among cooperating BSs is CSI and UE data packet identification information. Piggybacking UE data and signaling can be done if the UE data is distributed from one of the BSs in

the cooperating cluster. Therefore, the in-band or out-of-band signaling schemes result in the same trade-offs.

In the fully centralized JT implementation scheme, the data distribution inherently happens in-band. As the UE data is encoded at a central location, only the encoded data, e.g., IQ samples, has to be sent to the cooperating BSs. There is no separate precoding information necessary.

Due to the mentioned clear drawbacks resulting from out-of-band signaling, I focus on in-band signaling in the course of this thesis.

2.2.4 Backhaul requirements

Latency The duration for which CSI is valid, i.e., for which it accurately describes the state of the wireless channel, depends on the mobility of the UE. The minimum reasonable CSI validity in an LTE system is 1 ms, which corresponds to the LTE subframe duration and which is the scheduling interval in LTE. In scenarios with low UE mobility, CSI can be valid for a few tens of milliseconds. This means that for JP in the downlink, the duration from the point in time when the CSI is measured at UEs to the point in time when the cooperating BSs simultaneously transmit should be as low as possible. Simply speaking, the whole procedure summarized in Section 2.3 should be completed within a few milliseconds depending on UE mobility.

In an LTE system, feeding back the measured CSI from the UE to the BS already introduces up to 5 ms delay due to scheduling [42]. Hence, additional link latency and processing delay, e.g., at routers between cooperating BSs, play an important role. The CSI and UE data exchange needs to be as fast as possible, if possible within 1 ms delay, to guarantee an effective CoMP operation.

Note that the delay requirements of other CoMP techniques, like joint beamforming or joint scheduling, are similar. Hence, the following considerations are not only valid for JP but also for CoMP schemes that do not exchange user data.

For JP in the uplink, i.e., a joint reception at multiple BSs of a UE transmission, latency constraints are less strict as there is no deadline for CSI validity within which a sending operation has to happen. Received data copies and corresponding CSI can be processed at any time.

Capacity The second limiting factor is the increased capacity demand on the BSs' backhaul links. In LTE-Advanced, the downlink wireless peak data rate per BS sector is 1 GBit/s. This implies that for a foreseen 6 sector BS deployment, up to 6 GBit/s need to be sent wirelessly. From the backhauling perspective, the amount of user data which needs to be transported to a 6-sector BS is around 50 % of the wireless data rate, i.e., 3 GBit/s. The wireless rate of 6 GBit/s results from wireless coding and other overhead [43].

Considering upcoming optical backhauling technologies, the 2.5 GBit/s capacity provided by one WDM channel [44] is already insufficient to serve a single *non-cooperating* BS. Furthermore, as shown in Figure 2.3 on page 11, BSs that use JT have to exchange

additional signaling and UE data. Hence, even more backhaul capacity is required to allow JT CoMP, where UE data needs to be shared among cooperating BSs.

The actual amount of additional capacity needed in the backhaul network needed for JT depends on the JT implementation aspects mentioned in the previous sections. In any case, the backhaul network's limitations, even when using near-future optical technologies, will limit the feasibility of JT.

Evaluation of backhaul network influence The wireless performance of CoMP systems has been evaluated in many different publications, assuming an optimal backhaul network infrastructure. Recognizing the importance of the requirements which CoMP schemes put on the backhaul network, in recent publications many authors tend to assume simplified capacity-constrained and/or unreliable backhaul networks for their evaluations. In [45], the authors consider a cooperative downlink transmission for a capacity-limited backhaul and partial channel knowledge at BSs and UEs. The performance of certain cooperation schemes in terms of rate/backhaul trade-off for different interference scenarios is evaluated. Authors in [46] study the overall capacity needs for a backhaul network for supporting different CoMP schemes. They show that when UE data sharing between BSs occurs, the capacity requirements on the backhaul get significantly higher, directly proportional to the number of BSs in the cluster. To achieve the desired data rates at a reasonable backhaul load, they propose to quantize the data exchanged between the cooperating BSs. In [15], authors discuss implementation options for CoMP, finding out a relation between the desired Signal to Noise Ratio (SNR) for the UE and the required backhaul capacity. In [47] authors evaluate the data rates the UE can achieve in a CoMP system with constrained backhaul, trying to quantify the UE data that needs to be exchanged between the cooperating BSs. Based on this analysis, they derive an "achievable rate" for which the backhaul load can be low. Finally, [48] and [49] evaluate the CoMP performance on a topologically constrained backhaul where links exist only between neighboring BSs.

2.2.5 Summary

The strict requirements for the JT procedure in terms of capacity and delay constraints have to be fulfilled by the backhaul network that interconnects the BSs. Hence, the possibility of using JT in a cluster of BSs strongly depends on the backhaul network capabilities and how JT is actually implemented in the network as this influences the requirements. Table 2.1 on the next page summarizes the influence of the individual architectural design choices discussed in the previous sections.

I pointed out that even without any CoMP mechanisms in place, the backhaul network requirements of future LTE-A systems will be difficult to implement even with optical technologies due to the targeted high wireless data rates. Therefore, in the remainder of this thesis, I will focus on those implementation approaches that are most promising, i.e., those that will provide a high performance gain while keeping the additional overhead in the network as low as possible (marked in gray in Table 2.1 on the facing page). First, this is the semi-centralized implementation strategy, which needs

Table 2.1: Summary of implementation aspects' influence. The symbol + means that the parameter influences the corresponding metric (repetitions indicate a stronger influence) and ◯ stands for no influence. T stands for transport plane, C for control plane. Configurations marked in gray are chosen in the remainder of this thesis unless stated otherwise. They are selected to achieve a system with high performance gain and low additional overhead.

		Perf. gain	Requirements Capacity	Requirements Latency	Complexity
Implementation	fully-centr.	◯	+ + +	+	◯
	semi-centr.	◯	+	+	◯
	distributed	◯	+ +	+	+
Clustering	static	+	+	+	+
	dynamic	+ +	+	+	+ +
Signaling	in-band	◯	T: +, C: ◯	T: +, C: ◯	◯
	out-of-band	◯	T: + +, C: +	T: ◯, C: +	+
MIMO	MU	+ +	+	+ +	+ +
	SU	+	+	+	+

the lowest additional capacity in the backhaul network compared to the two other alternatives. Among the possible signaling schemes, the in-band approach has advantages as the capacity requirements in the transport plane are lower as unnecessary UE data exchange can be easily avoided. The MU-MIMO approach promises higher performance gains for UEs. Finally, I decided to consider both clustering approaches as they are both lively discussed in the research community and both can achieve reasonable gains.

Compared to existing work discussed in the course of this chapter, my work presented in the following chapters is orthogonal. I investigate CoMP from the networking perspective, taking into account CoMP capacity and latency requirements in realistic backhaul network deployment scenarios. This way, I highlight aspects which have not been considered in the past literature.

Chapter 3

Static clustering

Contents

The common denominator of most CoMP techniques are controller/processing nodes within the network of the mobile operator. Although the impact of a limited wireline network has been roughly studied [50], the problem of *where* to optimally place these nodes inside the network and *how* to connect them to the BSs has not been addressed. For this, requirements of the CoMP techniques and properties of the usually wireline backhaul network both have to be taken into account. Otherwise, the wireless performance decreases or cooperation even becomes impossible, which results in a degraded service quality for the users.

This chapter first discusses the basic network planning approach for CoMP-enabled systems in Section 3.1. It takes into account backhaul network properties to position CoMP controllers and to form clusters by assigning BSs to them. In Section 3.2, this approach is extended to also take into account desired CoMP clusters as additional input. This way, the network is configured to support CoMP at specified locations.

3.1 Basic clustering and backhaul configuration

In a CoMP system that uses a static clustering approach, the BS clusters used for cooperative transmissions do not change over time. During operation, UEs are assigned to the best-fitting cluster in the system but clusters are not changed according to the wireless channel conditions. Hence, the positioning of CoMP controllers and the assignment of BSs to controllers can be done offline during the design or planning phase of the network. To address this issue, I first formulate the problem in detail in Section 3.1.1 before I present a Mixed Integer Linear Program (MILP) that solves this problem optimally (Section 3.1.2). Thereafter, I propose a heuristic algorithm in Section 3.1.3 that approximates the optimal solution while reducing the runtime and memory requirements. Section 3.1.4 presents an evaluation of both approaches.

3.1.1 Problem description

The placement of CoMP controllers, as introduced in Section 2.2.1, and their connection to the cooperating BSs play a crucial role in the overall backhaul architecture design. System parameters like propagation delay in the backhaul network, required link capacity for transporting UE data and signaling information, synchronization between cooperative BSs and the CoMP controllers, Capital Expenditure (CAPEX) and Operating Expense (OPEX) of the backhaul architecture need to be taken into account.

This leads to the following problem for designing backhaul networks for cooperative, cellular mobile access networks. As input, it is first known that BSs in a certain part of the mobile network should jointly operate using CoMP. Furthermore, the properties of the backhaul network interconnecting these BSs are known. Such properties are existing links, possible additional links, link capacities, link latencies, etc. In addition, the requirements of the intended CoMP technique are known, like the required backhaul capacity, delay constraints, etc. Based on these inputs, we need to know how to optimally position CoMP controllers in the network (co-located with BSs), how to assign BSs to the controllers, and how to route the data between them if there are multiple possibilities such that all CoMP backhaul requirements are fulfilled.

To solve this network design problem, we use the parameters in Table 3.1 on the next page to define the input scenario. These parameters define a graph that has a vertex for each BS $b \in B$ and an edge for each link $l \in L$ between the BSs. All vertices and edges are annotated with properties, like capacity and latency, that are important for the network design. Additionally, we add four global properties ($cs_{\min}$, $cs_{\max}$, t_{proc}, and $t_{\max}$) that are independent of BSs or links and can be used to limit valid solutions.

3.1.2 Optimal solution

This section introduces a Mixed Integer Linear Program (MILP) that describes the network design problem introduced in Section 3.1.1. Based on the input, the MILP calculates optimal positions for the controller/processing nodes and assigns BSs to these nodes while minimizing costs. This assignment leads to the optimal interconnection of

Table 3.1: Input scenario parameters

B	Set of BSs
L	Set of links, where $L \subseteq B \times B$
b_costF_u	Fixed costs for deploying a controller at BS $u \in B$
b_costC_u	Costs per controlled BS for a controller at BS $u \in B$
b_cap_u	Required capacity (e.g., bandwidth/data rate) at BS $u \in B$
$\text{l\_cap}_{u,v}$	Capacity (e.g., bandwidth/data rate) of link $(u, v) \in L$
$\text{l\_t}_{u,v}$	Latency of link $(u, v) \in L$
$\text{l\_cost}_{u,v}$	Costs of link $(u, v) \in L$
cs_{min}	Minimum required cluster size
cs_{max}	Maximum allowed cluster size
t_{proc}	Processing time at controller
t_{max}	Maximum total round trip time from BS to controller

all BSs. The MILP uses the parameters in Table 3.1 as input and uses the variables shown in Table 3.2 to optimize the metric. In the following, I will use monetary costs as optimization goal as this is the probably most important aspect for network operators. Nevertheless, the optimization goal can easily be adapted to a different cost metric, like energy consumption.

Table 3.2: Decision variables that are determined by the MILP solver.

$\text{l\_act}_{u,v}$	Determines whether link $(u, v) \in L$ is active, $\text{l\_act}_{u,v} \in \{0, 1\}$
b_act_u	Determines whether a controller/processing node is colocated at BS $u \in B$, $\text{b\_act}_u \in \{0, 1\}$
$\text{f}_{s,d,u,v}$	Determines whether a data flow from a controller/ processing node at $s \in B$ goes to $d \in B$ over link $(u, v) \in L$, $\text{f}_{s,d,u,v} \in \{0, 1\}$
b_actCostF_u	Actual fixed controller/processor costs at $u \in B$, $\text{b\_actCostF}_u \geq 0$. It is 0 if no controller is located at u and b_costF_u is there is a controller active.
b_actCostC_u	Actual controller/processor costs at $u \in B$ per each controlled BS, $\text{b\_actCostC}_u \geq 0$. It is 0 if no controller is located at u. Otherwise ts is $c \cdot \text{b\_costC}_u$, where c is the number of controlled BSs.
$\text{l\_actCost}_{u,v}$	Actual costs for link $(u, v) \in L$, $\text{l\_actCost}_{u,v} \geq 0$. If link (u, v) is required in the configuration, the variable gets $\text{l\_cost}_{u,v}$. Otherwise it is 0.

Setting $\text{l\_act}_{u,v}$ to 1 means that the link (u, v) transports some flow, i.e., there is at least one $\text{f}_{s,d,u,v}$ set to 1. Similar to this, if b_act_u is set to 1, a controller/processing

function is located at BS u.

The three variables b_actCostF_u, b_actCostC_u, and $\text{l\_actCost}_{u,v}$ contain the actual costs for controllers and links. These costs depend on whether a controller/processing node is actually colocated at a BS and on whether a link is active, i.e., used to connect clustered BSs, or not.

To get the total link and controller/processing node costs for a network configuration, we need to sum up the individual costs. This is done in Equation (3.1) and Equation (3.2), respectively.

$$\text{l\_cost}_{\text{total}} = \sum_{(u,v)\in L} \text{l\_actCost}_{u,v} \tag{3.1}$$

$$\text{b\_cost}_{\text{total}} = \sum_{u\in B} \text{b\_actCostF}_u + \text{b\_actCostC}_u \tag{3.2}$$

The goal is to minimize the total costs while taking into account the constraints of the applied wireless cooperation scheme, defined by the parameters in Table 3.1 on the preceding page. The following MILP achieves that:

$$\text{minimize } \text{b\_cost}_{\text{total}} + \text{l\_cost}_{\text{total}} \tag{3.3}$$

$$\text{s. t.} \sum_{s\in B,(u,d)\in L} \text{f}_{s,d,u,d} = 1,\ \forall d \in B, \tag{3.4}$$

$$\text{s. t.} \sum_{s\in B,(s,u)\in L} \text{f}_{s,d,s,u} = 1,\ \forall d \in B, \tag{3.5}$$

$$\text{s. t.} \sum_{(v,u)\in L} \text{f}_{s,d,v,u} = \sum_{(u,w)\in L} \text{f}_{s,d,u,w}, \forall u \in B, (s,d) \in B \times B, u \neq s, u \neq d, \tag{3.6}$$

$$\text{s. t. } \text{f}_{s,d,u,v} = 0, \forall (s,d,u,v) \in B \times B \times B \times B, s \neq d, u = v, \tag{3.7}$$

$$\text{s. t. } \text{l\_act}_{u,v} \cdot \mathcal{M} \geq \sum_{(s,d)\in B\times B} \text{f}_{s,d,u,v},\ \forall (u,v) \in L, \tag{3.8}$$

$$\text{s. t. } \text{b\_act}_s \cdot \mathcal{M} \geq \sum_{d\in B,(s,v)\in L} \text{f}_{s,d,s,v},\ \forall s \in B, \tag{3.9}$$

$$\text{s. t.} \sum_{(d,v)\in B\times B,(c,v)\in L} \text{f}_{c,d,c,v} \geq cs_{\min} \cdot \text{b\_act}_c, \forall c \in B, \tag{3.10}$$

$$\text{s. t.} \sum_{(d,v)\in B\times B,(c,v)\in L} \text{f}_{c,d,c,v} \leq cs_{\max},\ \forall c \in B, \tag{3.11}$$

$$\text{s. t.} \sum_{(s,d)\in B\times B} \text{f}_{s,d,u,v} \cdot \text{b\_cap}_d \leq \text{l\_cap}_{u,v}, \forall (u,v) \in L, \tag{3.12}$$

$$\text{s. t.} \sum_{(s,d)\in B\times B} \text{f}_{s,d,u,v} \cdot \text{b\_cap}_d \leq \text{l\_cap}_{v,u}, \forall (v,u) \in L, \tag{3.13}$$

$$\text{s. t.} \sum_{(u,v)\in L} \mathrm{f}_{s,d,u,v} \cdot (\mathrm{l\_t}_{u,v} + \mathrm{l\_t}_{v,u}) \leq t_{\max} - t_{\mathrm{proc}}, \forall (s,d) \in B \times B, \tag{3.14}$$

$$\text{s. t. } \mathrm{b\_actCostF}_c = \mathrm{b\_act}_c \cdot \mathrm{b\_costF}_c, \ \forall c \in B, \tag{3.15}$$

$$\text{s. t. } \mathrm{b\_actCostC}_c = \sum_{d\in B,(c,v)\in L} \mathrm{f}_{c,d,c,v} \cdot \mathrm{b\_costC}_c, \forall c \in B, \tag{3.16}$$

$$\text{s. t. } \mathrm{l\_actCost}_{u,v} = \ \mathrm{l\_act}_{u,v} \cdot \mathrm{l\_cost}_{u,v}, \ \forall (u,v) \in L, \tag{3.17}$$

The first constraint in Equation (3.4) ensures that each BS is assigned to exactly one controller. This is done by fixing the number of flows that end at each BS to 1. The second constraint, shown in Equation (3.5), guarantees that each flow that ends at a BS also has a start BS – the controller. Equation (3.6) contains the third constraint, which is necessary to create the flows in the network. It guarantees flow balance, i.e., whenever a flow enters a node it also has to leave it again, except if the node is the flow's source or destination. Finally, Equation (3.7) forbids local loops for each flow.

After having created the flows between the controllers and the BSs, the helper constraint in Equation (3.8) activates all links in the network that are required to transport one of the flows. Similarly, Equation (3.9) activates the controller functionality at each BS that is source of at least one flow. Both constraints use a "big-M constant" to achieve this. In this method, an artificial constant $\mathcal{M}$ (larger than the maximum possible number of flows passing over a single link and larger than the maximum number of BSs controlled by a single CoMP controller) is used to set the binary variables $\mathrm{l\_act}_{u,v}$ and $\mathrm{b\_act}_s$ to one whenever at least one flow traverses the link (u,v) or at least one BS is controlled by the CoMP controller at BS s, respectively.

As many cooperation schemes, like joint precoding, require a minimum amount of BSs being jointly controlled to achieve the desired gains, the constraint in Equation (3.10) requires at least $cs_{\min}$ BSs to be connected to the same controller. At the same time, Equation (3.11) limits the maximum cluster size to $cs_{\max}$. This might be useful in case the controller capacities are limited and hence clusters cannot exceed a certain size.

The constraints in Equation (3.12) and Equation (3.13) ensure that the link capacities are not exceeded in the downlink (from controller to the BS) and in the uplink.

The next constraint in Equation (3.14) ensures that the round-trip delay between a controller and a BS is below an upper bound. This bound is calculated by subtracting the required processing time t_{proc} at the controller from the maximum allowed delay $t_{\max}$. Such a constraint is important, e.g., for joint precoding. Here, the encoded data must be sent while the CSI, based on which the encoding was done, is still valid.

Finally, the last three constraints calculate the link and controller costs for the resulting network configuration. Equation (3.15) and Equation (3.16) define the fixed and dynamic controller costs. Equation (3.17) sets the costs for each link depending on whether it is active or not.

Depending on the configuration of the MILP solver, the presented MILP is able to return the *optimal* solution for the problem according to a certain cost metric. The drawback, however, is that solving the MILP is NP-hard. This causes very long runtimes of the MILP solver for finding solutions to the problem. Evaluations show that for an

input scenario consisting of 50 BSs, the solving time easily exceeds 20 hours on today's computing hardware. More details will be given in Section 3.1.4.

Not only solving the problem using a MILP but also the described problem itself is NP-hard. The problem can be seen as a variant of the constrained multicast routing problem, which is known to be NP-hard [51, 52]. In this problem, multiple sources intend to send data to multiple destinations via a given network while a certain cost metric should be minimized. Moreover, the link capacities in the network are limited. In the described CoMP problem, the CoMP controllers correspond to the multicast sources and the BSs in the cooperative clusters are the multicast destinations. The overall costs should be minimized while all CoMP requirements, i.e., link capacities and latency constraints, need to be fulfilled. Hence, the CoMP optimization problem is NP-hard, too, which indicates the need to develop a heuristic solution approach to be able to deal with large input scenarios.

3.1.3 Heuristic approach

The high complexity for solving the problem optimally using the proposed MILP might be feasible for calculating a network configuration offline, i.e., before deploying it, for small input scenarios. More advanced applications, like reacting to problems in the network (e.g., link failures), which require to change the configuration, are impossible.

To circumvent the MILP's runtime problem, we propose a heuristic algorithm. The heuristic takes the same input as the MILP (i.e., a property graph containing the BSs and potential interconnections) and produces the same kind of output (locations of controllers/processors and interconnections between these components).

To exploit the proximity properties of the BS clusters, we have chosen a greedy approach based on a modified Breadth First Search (BFS) algorithm to select clusters of BSs. The resulting heuristic is summarized in Algorithm 3.1.

Algorithm 3.1 configureCoopNetwork($G_{\text{in}}(B, L)$)

```
1: G_out ← ∅ // initialize output graph with empty set
2: while |B| > 0 do
3:     s ← chooseController(G_in)
4:     G_newClust ← BFS'(G_in, s) // determine new cluster
5:     G_out ← G_out ∪ G_newClust // add new cluster to G_out
6:     G_in ← G_in \ G_newClust // remove new cluster from G_in
7: end while
8: return G_out
```

The method configureCoopNetwork expects the input parameter G_{in}, which is the input graph that consists of the set of BSs B and the potential set of links L that interconnect the BSs. The algorithm first initializes the output graph G_{out}. After this, it traverses a loop where in each iteration one new cluster is created and added to G_{out}. Furthermore, BSs that have been assigned to a cluster are removed from G_{in}.

At the beginning of each iteration, one of the BSs in G_{in} is selected that will act as a cluster controller. Starting from this new controller s, the modified BFS algorithm BFS' is executed. This algorithm differs from the conventional BFS such that in addition to checking for the existence of a link, it additionally checks if backhaul requirements are fulfilled before adding a new node to the BFS tree. This way, the algorithm determines the maximum feasible cluster around s and returns the resulting cluster graph G_{newClust}. This graph contains all BSs that are member of the cluster and links that connect these BSs to the controller s.

The way how to choose s from G_{in} plays an important role for the quality of the heuristic's output. If s is chosen arbitrarily, it can happen that a lot of cutoff is generated, i.e., there are isolated BSs or small groups of BSs left, that cannot be joined to a larger cluster. As this reduces the gain of CoMP, such an approach is not desirable. To reduce cutoff, the method chooseController in Algorithm 3.2 selects a BS from G_{in} such that the number of hops to the border of G_{in} equals the expected CoMP cluster radius in hops. We define the border of G_{in} as the outer nodes in the planar representation of G_{in}. As G_{in} shrinks in each iteration of the algorithm (already assigned BSs are removed), the border of G_{in} changes as well after each iteration step.

In case of all BSs in G_{in} are closer to the graph's border than the expected cluster radius, any BS can be chosen as new controller. This is because all nodes are probably able to control all remaining BSs in G_{in}.

The expected cluster radius can be calculated depending on the current backhaul network properties and depending on the CoMP capacity and latency requirements, as shown in lines 1-3 of Algorithm 3.2. The higher the error during radius estimation, the more the system behaves like one that randomly picks s, i.e., the cutoff increases and CoMP performance decreases.

In the following, we give an example how this can be done for a JT CoMP system that uses MU-MIMO as cooperation technique (Algorithm 3.2).

Algorithm 3.2 chooseController($G(B, L)$)

1: $cr_{\text{lat}} \leftarrow (t_{\max} - t_{\text{proc}}) \,/\, \left(2 \cdot \sum_{(u,v)\in L} \text{l\_t}_{u,v}/|L|\right)$
2: $cr_{\text{cap}} \leftarrow \left(\sum_{(u,v)\in L} \text{l\_cap}_{u,v}/|L|\right) \,/\, \left(\sum_{u\in B} \text{b\_cap}_{u}/|B|\right)$
3: $cr_{\text{overall}} \leftarrow \text{MIN}(cr_{\text{lat}}, cr_{\text{cap}})$ // expected cluster radius (in hops)
4: $s \leftarrow$ getNodeNHopsFromBorder(G, cr_{overall})
5: **return** s

The first major backhaul constraint for JT is the delay from measuring the CSI until the point in time where the jointly encoded data is sent. Accordingly, the expected cluster radius regarding the latency cr_{lat} can be calculated as shown in the first line of Algorithm 3.2. The parameters $t_{\max}$ and t_{proc} contain the maximum allowed time span from measuring the CSI to the actual sending of the data (i.e., 1 ms in LTE), and the processing delay at the controller, respectively (as defined in Table 3.1 on page 19). Their difference is divided by the mean link latency. Using the maximum link latency

here is not appropriate as this would lead to expected clusters that are always too small compared to the actual feasible ones.

The second limitation for JT is that the links must have enough capacity to transport the UE and signaling data from the controller to the cooperating BSs. So the amount of bandwidth that is required at each BS and the link capacities from the controller to the BSs (possibly involving multiple hops) limit the maximum possible cluster size, too. This is taken into account in line 2 of Algorithm 3.2 on the previous page. There, the expected cluster radius depending on the capacity properties of the input graph cr_{cap} is calculated. The radius depends on the mean link capacity and the mean capacity requirement of the BSs. Hence the expected cluster radius changes for different CoMP schemes as they have different constraints that have to be fulfilled.

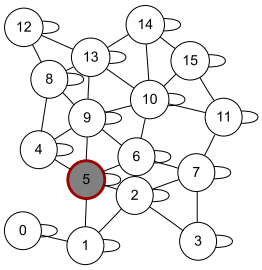
The overall expected cluster radius cr_{overall} is the minimum of cr_{lat} and cr_{cap}. Using cr_{overall} as input, the method getNodeNHopsFromBorder returns a BS from the input graph that is exactly cr_{overall} hops away from the graph's border. This BS is the new controller. In case there are multiple BSs that are cr_{overall} hops away from the graph border, one of them is chosen randomly. This step might be improved in the future for a more clever tie braking to further improve the overall gain.

Now that the heuristic has selected a new controller s from the input graph G_{in} using chooseController, a modified BFS is started with s as start node. Compared to a standard BFS algorithm, whenever the BFS' algorithm traverses a link and finds a new BS x in G_{in}, it checks whether x can be added to the current cluster without violating any constraint. For the capacity constraint this is done by checking if each link between s and x provides enough free capacity; the maximum tolerable latency is checked on the full path between s and x. If all requirements are fulfilled, x is added to the current cluster and to the BFS's queue of BSs to continue the search from later on. If adding x to the cluster is not possible, the search algorithm does not continue searching for new BSs on this path of the search tree. As soon as the BFS' algorithm terminates, a new cluster has been created around the controller s.

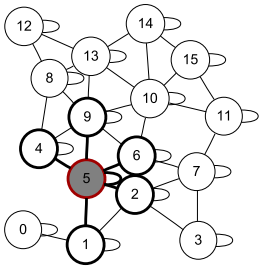
After BFS' has terminated, the new cluster (represented by G_{newClust}) is added to the output graph G_{out} and removed from the input graph G_{in}. The loop continues with the next iteration until all BSs have been assigned to a cluster.

The algorithm is illustrated in an example in Figure 3.1 on the facing page. The scenario consists of 16 BSs that shall be assigned to controllers such that all backhaul requirements are fulfilled.

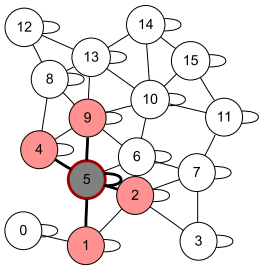
In the example, first, a controller is selected for the new (red) cluster. The algorithm randomly chooses BS 5 as it is 2 hops away from the border of the given input scenario (consisting of BSs 0, 1, 2, 3, 4, 7, 8, 11, 12, 13, 14, 15). The expected cluster radius cr_{overall} of 2 has been chosen as example for this illustration. Thereafter, the backhaul network conditions to the neighbors of BS 5 are checked if they can fulfill CoMP requirements. Those BSs that can participate are added to the red cluster in Figure 3.1c on the next page. The same happens in the next four steps, where the cluster is further extended. After no more cluster extension is possible in Figure 3.1h on the facing page, a new cluster (blue) is created with BS 15 as controller. It is again 2 hops away from the border of the remainder of G_{in} after the (now assigned) nodes of the red cluster have



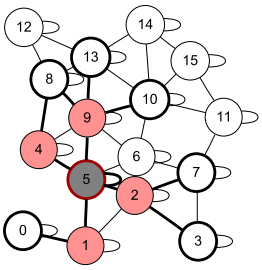
(a) BS 5 is selected as controller for the new cluster (red).



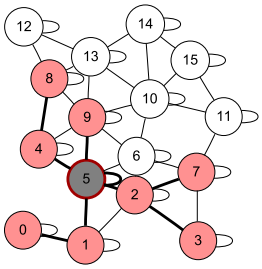
(b) Check neighbors (BSs 1, 2, 4, 6, 9) if backhaul network fulfills requirements.



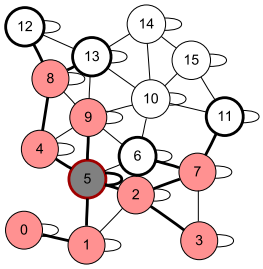
(c) Add BSs to new cluster where requirements are fulfilled (BSs 1, 2, 4, 9).



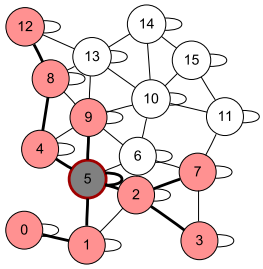
(d) Check cluster extensions (BSs 0, 3, 7, 8, 10, 13) from newly added BSs.



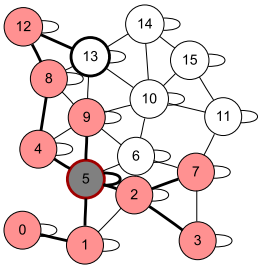
(e) Add BSs to new cluster where requirements are fulfilled (BSs 0, 3, 7, 8).



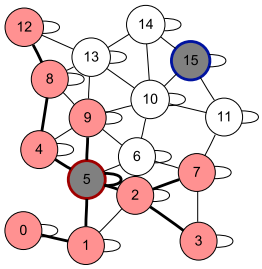
(f) Check cluster extensions (BSs 6, 11, 12, 13) from newly added BSs.



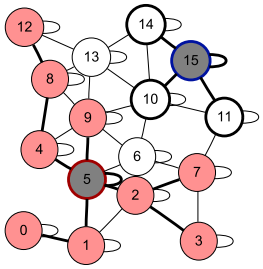
(g) Add BSs to new cluster where requirements are fulfilled (only BS 12).



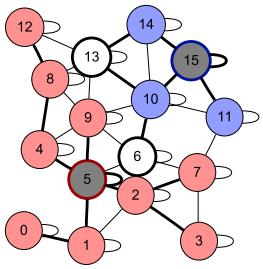
(h) Check cluster extensions (BS 13) from newly added BSs.



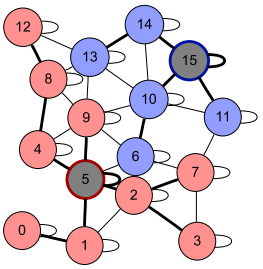
(i) First BFS terminates and BS 15 is selected as controller for second cluster (blue).



(j) Check neighbors (BSs 10, 11, 14) if backhaul network fulfills requirements.



(k) Add BSs 10, 11, 14 to new cluster; then check for further extension (BSs 6, 13).



(l) Add BSs 6 and 13 to new cluster. The algorithm terminates (no BSs remaining).

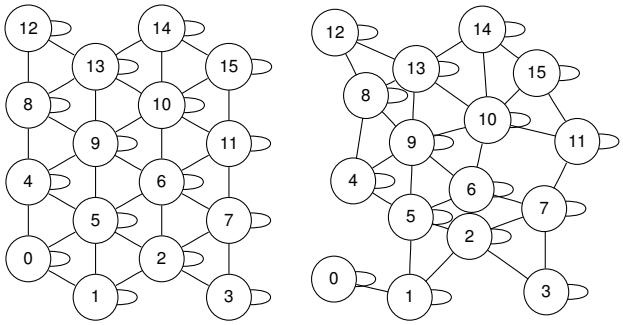
Figure 3.1: Example algorithm run for 16 BSs. Expected cluster radius $cr_{\text{overall}} = 2$ hops. Unassigned BSs are white, controllers gray; clusters members are red or blue. Links being checked or finally used to communicate within the clusters are marked with thick lines.

been removed. The border now consists of all remaining nodes, from which 15 has been chosen randomly (it is 2 hops away from node 6 and 13). The cluster is then extended as it was done for the red cluster until no unassigned BSs are left in Figure 3.1l.

3.1.4 Evaluation

This section covers the evaluation of solving the static clustering problem using the MILP described in Section 3.1.2 and the heuristic algorithm from Section 3.1.3. In the following figures, all confidence intervals have been calculated for a confidence level of 95 % and are shown in all plots in the following figures.

The input scenario consists of a matrix of n BSs that form a square. We look at two different arrangements of the BSs. The first is a regular arrangement where all BSs have identical inter-BS distances of $\overline{s} = 500\,m$, which corresponds to an urban deployment scenario, and have potential interconnections to all neighboring BSs. The second type of arrangement is irregular and adds randomness to the BS positioning. In this scenario, BSs still have a *mean* distance of $\overline{s}$ but a normally distributed random value with zero mean and standard deviation $\overline{s}/5$ is added to the X and Y coordinates of the BS locations. BSs are interconnected if their distance is below $1.25 \cdot \overline{s}$. The reason for having irregular input scenarios is to check whether our heuristic suffers from arbitrarily choosing new neighbors during the BFS. Figure 3.2 shows two examples of these two arrangement types for $n = 16$.



(a) Regular arrangement (b) Irregular arrangement

Figure 3.2: Example input scenarios for $n = 16$

The links are optical and have a latency of $s \cdot \frac{1.45}{c}$, where c is the speed of light and 1.45 the refraction index of the fiber. This corresponds to a typical Single-Mode Fiber (SMF) setup. These optical parameters will be used throughout the whole thesis.

We further assume a Radio over Fiber (RoF) setup from the controllers to the BSs.[1] The RoF bandwidth of each link is 3000 MHz, which corresponds to currently available RoF equipment. Every BSs has 3 sectors that have 4 antennas each, operated at a bandwidth of 100 MHz, which corresponds to a future wide band wireless system like LTE-A. Table 3.3 on the next page summarizes the input parameters.

We did experiments for $n \in \{3^2, 4^2, 5^2, 6^2, 7^2\}$ on a dual-core machine (3.33 GHz per core, 4 GB RAM) and set the optimality gap of the MILP solver to 0.1. This means

[1] Note that this only affects the evaluation. Our proposed methods are generic enough to support any technology.

Table 3.3: Chosen input parameters

Parameter	Value
C	$\{1..n\}$
L	$\{(u, v) \in C \times C\}$
b_costF_u	3
b_costC_u	0.5
b_cap_u	$3 \cdot 4 \cdot 100\,\text{MHz} = 1200\,\text{MHz}$
$\text{l\_cap}_{u,v}$	3000 MHz if $u \neq v$, else ∞
$\text{l\_t}_{u,v}$	$s \cdot 1.45/c$ ($\approx 13.4\,\mu$s in regular scenario)
$\text{l\_cost}_{u,v}$	$s/\overline{s} = 1$
cs_{min}	3
cs_{max}	50
t_{proc}	0.5 ms
t_{max}	1 ms

that the solution is at most 10 % worse than the optimal solution. Such a high gap was required to achieve solver runtimes below one week for the larger input scenarios.

Compared to solving the problem optimally with the MILP, the heuristic method has a much shorter runtime and lower memory requirements. The difference is illustrated in Figure 3.3 on the following page for the regular (*reg*) and irregular (*irreg*) input scenarios. The figure shows two versions of our heuristic method to later demonstrate the advantage of choosing the start nodes for the BFS according to our scheme. The *simple* one simply picks the first node at the border of the input graph. The *mhop* heuristic chooses new controller locations as proposed in Algorithm 3.2 on page 23.

The plot shows that while the MILP solver runtime increases up to 20 hours, the heuristic's runtime stays between 1.5 and 6 orders of magnitude below that (runtime always below 1 ms).

To get an idea of the quality of the heuristics' output, Figure 3.4 on the following page compares the costs of the solution given by the MILP solver to solution costs of the two versions of our proposed heuristic. The considered costs are the total costs for the returned network configuration, using the unit costs defined in Table 3.3.

The returned output of the heuristic is at most 5 % worse than the output provided by the MILP solver. Furthermore, the difference between the *simple* and *mhop* heuristic shows that our proposed way of choosing a new controller has a major influence, especially for irregular input scenarios, i.e., where BSs are distributed randomly in the plane and where links have different properties in terms of capacity and latency.

Note that in Figure 3.4a on the following page, the heuristic (mhop) even produces lower costs than the MILP. This is possible as we set the optimality gap of the MILP solver to 10 %. The result of the heuristic, however, is better than that.

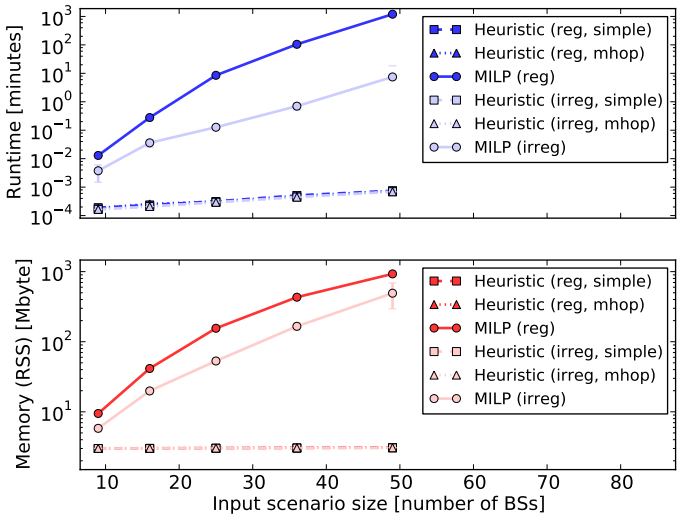
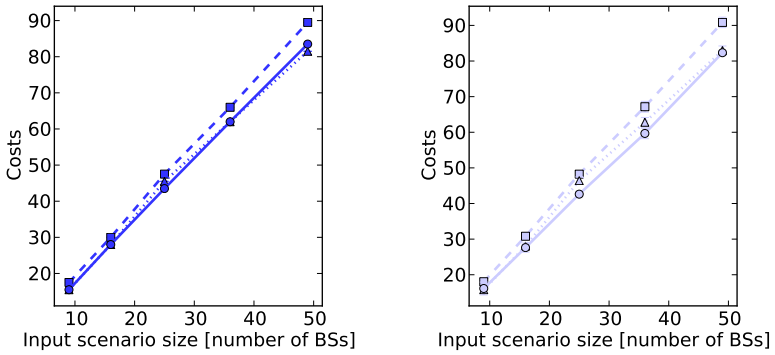


Figure 3.3: Solver and heuristic execution time and memory consumption depending on input scenario size. For the heuristic, the results are nearly identical for all parameter combinations. Hence the curves for the heuristics overlap.



(a) Regular input scenario (b) Irregular input scenario

Figure 3.4: Costs of the resulting network configuration depending on input scenario size. Please refer to the legend of Figure 3.3.

3.2 Clustering and backhaul configuration based on desired clusters

The approach in Section 3.1 assigns all BSs in the network to static clusters only based on backhaul network capabilities. This enables basic CoMP functions in the network but the resulting clusters will likely not provide the best possible performance as the wireless conditions are not taken into account. Hence, both domains have to be taken into account jointly, such that CoMP is available at locations where it is needed most. To achieve this, Martin Dräxler and I extended the basic approach presented in Section 3.1.

The extended problem statement is discussed in detail in Section 3.2.1 before I present an MILP that allows to solve this problem optimally (Section 3.2.2). Thereafter, I propose a heuristic algorithm in Section 3.2.3 that approximates the optimal solution while significantly reducing the runtime and memory requirements. Section 3.2.4 presents an evaluation of both approaches in different scenarios.

3.2.1 Problem description

As CoMP techniques are used to combat interference of neighboring BSs (downlink) or UEs (uplink) to increase the wireless spectrum reuse, the gain of applying CoMP strongly depends on the wireless channel conditions between the set of cooperating BSs and the UEs. Hence, the main decision criterion based on which BSs are selected for cooperation should be the wireless channel conditions. In the following, I will call these clusters *desired clusters*.

Definition (Desired Cluster)

A desired cluster is a set of BSs that has been selected according certain criteria like wireless channel conditions. Applying CoMP among the member BSs of the desired cluster is expected to improve service quality for selected UEs. A desired cluster should be selected such that it is a minimum set of cooperating BSs to achieve the required UE service quality improvement.

How to chose desired clusters in the context of static clustering is not focus of this work. It can be implemented in different ways, e.g., based on manual wireless measurements or based on coverage statistics gathered by operators. This information is then used as input for the network planning and complemented in a second step by backhaul network capabilities to make sure that desired BSs *can* cooperate.

Figure 3.5 shows a scenario where only some BS clusters are beneficial to improve the wireless service quality; others are not. This illustrates that taking into account the desired clusters is important even in the static clustering approach.

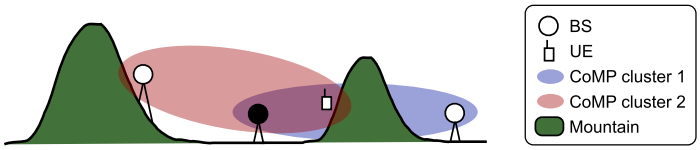
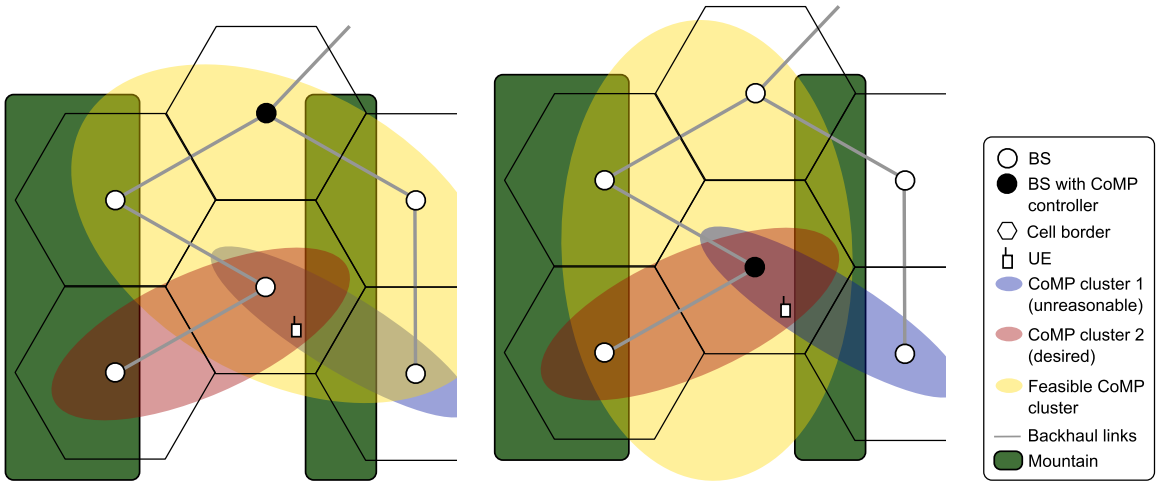


Figure 3.5: Not all BS clusters make sense. In this example, cluster 1 is meaningless as the mountain blocks signal propagation. Cluster 2, however, helps the UE.

Based on the desired clusters and the backhaul network conditions, the controller functions for the clusters, co-located at one of the BSs in the desired cluster, need to be

enabled in the network. The locations for the controller functions need to be selected such that the eventually feasible clusters (limited by the backhaul network capabilities) best match the desired clusters. This is illustrated in Figure 3.6. The illustration is based on the scenario in Figure 3.5 on the previous page. Instead of looking at the scenario from the side, it is now shown from the top to show the influence of the CoMP controller location.



(a) The upper BS hosts the CoMP controller function. Hence a large part of the shown network is able to participate in CoMP, as indicated by the large feasible cluster. The important desired cluster, however, is not feasible.

(b) Locating the CoMP controller function at the center BS shrinks the feasible cluster. The desired CoMP cluster, however, is feasible now, which it was not before. This is more beneficial than the solution shown on the left side.

Figure 3.6: Different CoMP controller locations in the network cause different feasible CoMP clusters, which have to be matched to the desired/reasonable clusters. The scenario is identical to Figure 3.5 on the previous page, just shown in a bird's eye view.

Now, the following problem needs to be solved for each *desired* cluster: One BS in the cluster has to be selected as controller such that capacity demands for all BSs in the cluster are fulfilled and the maximum latency between the controller and any BS in the cluster is not violated. This also requires to determine the routing paths between all BSs and the controller. I will call the set of BSs that is able to cooperate based on the given backhaul capabilities the *feasible cluster*. It is a subset of the *desired cluster*.

Definition (Feasible Cluster)

A *feasible cluster* is a set of BSs in which, according to the backhaul network capabilities, all cluster members *can* cooperate using a certain CoMP technique. The feasible cluster is specific to a chosen CoMP controller location, as it influences the data traveling through the network, and is also specific to a certain CoMP technique, as different techniques have different requirements on the backhaul network.

In case the backhaul network capabilities are not sufficient to support the desired CoMP technique, the *feasible* cluster does not contain all BSs of a *desired* cluster. Of course, a smaller feasible cluster also means a reduced gain from CoMP.

3.2.2 Optimal solution

This section extends the MILP described in Section 3.1.2 to additionally process desired clusters as input. The extended MILP takes the parameters in Table 3.4.

Table 3.4: Input scenario parameters

B	Set of BSs
L	Set of links, where $L \subseteq B \times B$
W_D	Set of BSs that cannot be controller ($W_D \subseteq B$)
$W_1 \dots W_n$	Desired clusters, where $W_i \subseteq B$, $W_i \cap W_j = \emptyset$, $\forall i = 1 \dots n$
W	Set of all BSs that are member of a cluster, calculated as $W = \bigcup W_i, \forall i = 1 \dots n$
$\overline{W}$	Set of all BSs that are *not* member of a cluster, calculated as $\overline{W} = B \setminus W$
b_costF_u ($t_c(c)$)	Fixed costs for setting up a controller at BS $u \in B$
b_costC_u ($t_p(c)$)	Costs per controlled BS for a controller at BS $u \in B$
b_cap_u ($b_{\text{req}}(v)$)	Required capacity (e.g., bandwidth/data rate) at BS $u \in B$
$\text{l\_cap}_{u,v}$ ($b_{\text{cap}}(u,v)$)	Capacity (e.g., bandwidth/data rate) of link $(u,v) \in L$
$\text{l\_t}_{u,v}$ ($l_{\text{cap}}(u,v)$)	Latency of link $(u,v) \in L$
$\text{l\_cost}_{u,v}$ ($t_l(u,v)$)	Costs of link $(u,v) \in L$
t_{proc} (l_{proc})	Processing time at controller
t_{max} (l_{max})	Maximum total round-trip time from BS to controller

Compared to the basic MILP parameters (Table 3.1 on page 19), the parameters W_D, W_i, W, and $\overline{W}$ have been added to cover the additional input of the desired clusters. Furthermore, cs_{min} and cs_{max} have been removed, as no general desired cluster sizes are given as input anymore but explicit desired clusters of BSs in terms of W_i.

The decision variables of the MILP are shown in Table 3.5. The meanings of the listed decision variables are identical to the basic MILP. Please refer to the description of Table 3.2 on page 19 for a detailed discussion.

Table 3.5: Decision variables

$\text{l\_act}_{u,v}$ ($l_{u,v}$)	Determines whether link $(u,v) \in L$ is active, $\text{l\_act}_{u,v} \in \{0,1\}$
b_act_u (c_v)	Determines whether a controller/processing node is colocated at BS $u \in B$, $\text{b\_act}_u \in \{0,1\}$
$\text{b\_ctrl}_{u,v}$ ($c_{u,v}$)	Determines whether a controller/processing node at BS $u \in B$ controls BS $v \in B$, $\text{b\_ctrl}_{u,v} \in \{0,1\}$
$\text{f}_{s,d,u,v}$	Determines whether a data flow from a controller/ processing node at $s \in B$ goes to $d \in B$ over link $(u,v) \in L$, $\text{f}_{s,d,u,v} \in \{0,1\}$
b_actCostF_u	Actual fixed controller/processor costs at $u \in B$, $\text{b\_actCostF}_u \geq 0$
b_actCostC_u	Actual controller/processor costs at $u \in B$ per each controlled BS, $\text{b\_actCostC}_u \geq 0$
$\text{l\_actCost}_{u,v}$	Actual costs for link $(u,v) \in L$, $\text{l\_actCost}_{u,v} \geq 0$

The individual costs need to be summed up to be able to minimize the total costs of the network configuration later on. This is done in Equation (3.18) and Equation (3.19), respectively.

$$\text{l\_cost}_{\text{total}} = \sum_{(u,v)\in L} \text{l\_actCost}_{u,v} \tag{3.18}$$

$$\text{b\_cost}_{\text{total}} = \sum_{u\in B} \text{b\_actCostF}_u + \text{b\_actCostC}_u \tag{3.19}$$

The goal is to use the following MILP to minimize the total costs while taking into account the constraints of the applied CoMP scheme (including the desired clusters), defined by the parameters in Table 3.1 on page 19. Hence, the resulting optimization goal is given in Equation (3.20).

$$\text{minimize } \text{b\_cost}_{\text{total}} + \text{l\_cost}_{\text{total}} \tag{3.20}$$

The first set of constraints determines how paths between controllers and BSs are established while adding BSs on the paths to the clusters, to ensure connected clusters. Each BS that is member of a cluster needs to be a destination of a routing path (Equation (3.21) and (3.22)) and every routing path has to have a source (Equation (3.23)), where the controller will be activated eventually. For every BS on a routing path, the

number of inbound and outbound routing paths has to be equal as ensured by Equation (3.24). The constraints in Equation (3.25) and (3.26) activate all links that are part of a routing path. Furthermore, every routing path has to contain exactly one controller at its source, which is ensured by constraints (3.27) and (3.28). Finally, constraints (3.29) and (3.30) set the variables $\text{b\_ctrl}_{s,d}$, which model the assignment of BSs to clusters. The MILP uses a big-M constant in some of the constraints, denoted as $\mathcal{M}$.

$$\text{s. t.} \sum_{s \in B,(u,d) \in L} f_{s,d,u,d} = 1, \forall d \in W \tag{3.21}$$

$$\text{s. t.} \sum_{s \in B,(u,d) \in L} f_{s,d,u,d} = 0, \forall d \in \overline{W} \tag{3.22}$$

$$\text{s. t.} \sum_{s \in B,(s,u) \in L} f_{s,d,s,u} = 1, \forall d \in W \tag{3.23}$$

$$\text{s. t.} \sum_{(v,u) \in L} f_{s,d,v,u} = \sum_{(u,w) \in L} f_{s,d,u,w}, \forall u \in B, (s,d) \in B \times W, u \neq s, u \neq d \tag{3.24}$$

$$\text{s. t. } \text{l\_act}_{u,v} \cdot \mathcal{M} \geq \sum_{(s,d) \in B \times W} f_{s,d,u,v}, \forall (u,v) \in L \tag{3.25}$$

$$\text{s. t. } \text{l\_act}_{u,v} \leq \sum_{(s,d) \in B \times W} f_{s,d,u,v}, \forall (u,v) \in L \tag{3.26}$$

$$\text{s. t. } \text{b\_act}_{s} \cdot \mathcal{M} \geq \sum_{d \in B,(s,v) \in L} f_{s,d,s,v}, \forall s \in B \tag{3.27}$$

$$\text{s. t. } \text{b\_act}_{s} \leq \sum_{d \in B,(s,v) \in L} f_{s,d,s,v}, \forall s \in B \tag{3.28}$$

$$\text{s. t. } \text{b\_ctrl}_{s,d} \cdot \mathcal{M} \geq \sum_{(u,v) \in L} f_{s,d,u,v}, \forall (s,d) \in B \times W \tag{3.29}$$

$$\text{s. t. } \text{b\_ctrl}_{s,d} \leq \sum_{(u,v) \in L} f_{s,d,u,v}, \forall (s,d) \in B \times W \tag{3.30}$$

The required link capacities and (round-trip) latencies between the controllers and the cluster members are ensured by the constraints in Equation (3.31) and (3.32), respectively.

$$\text{s. t.} \sum_{(s,d) \in B \times W} f_{s,d,u,v} \cdot \text{b\_cap}_{d} \leq \text{l\_cap}_{u,v}, \forall (u,v) \in L \tag{3.31}$$

$$\text{s. t.} \sum_{(u,v) \in L} f_{s,d,u,v} \cdot (\text{l\_t}_{u,v} + \text{l\_t}_{v,u}) \leq t_{\max} - t_{\text{proc}}, \forall (s,d) \in B \times W \tag{3.32}$$

Constraints in Equation (3.33) and (3.34) are responsible for calculating the controller costs at each BS in the network. Constraint (3.35) calculates the link costs. These three costs are summed up and eventually minimized, as shown in Equation (3.20).

$$\text{s. t. b\_actCostF}_c = \text{b\_act}_c \cdot \text{b\_costF}_c,\ \forall c \in B, \tag{3.33}$$

$$\text{s. t. b\_actCostC}_c = \sum_{d \in B,(c,v) \in L} \text{f}_{c,d,c,v} \cdot \text{b\_costC}_c, \forall c \in B, \tag{3.34}$$

$$\text{s. t. l\_actCost}_{u,v} = \text{l\_act}_{u,v} \cdot \text{l\_cost}_{u,v},\ \forall (u,v) \in L, \tag{3.35}$$

The constraint in Equation (3.36) ensures that no controller function is located at a BS that is not able to be a controller. The constraint in Equation (3.37) ensures that BSs that belong to one cluster are all assigned to the same controller. The constraint has to be repeated for all n desired clusters W_i with $i = 0 \dots n$.

$$\text{s. t. b\_act}_b = 0, \forall b \in W_D \tag{3.36}$$

$$\text{s. t. b\_ctrl}_{c,a} = \text{b\_ctrl}_{c,b}, \forall (a,b) \in W_i \times W_i, c \in B, \forall i = 0 \dots n \tag{3.37}$$

Furthermore, constraints in Equation (3.38) and (3.39) ensure that all routing paths, and thus all clusters, are disjoint.

$$\text{s. t. } \sum_{d \in W} f_{s,d,u,v} \leq \text{b\_ctrl}_{s,u} \cdot \mathcal{M}, \forall s \in B, (u,v) \in L \tag{3.38}$$

$$\text{s. t. b\_ctrl}_{a,a} = \text{b\_act}_a, \forall a \in B \tag{3.39}$$

We have shown that clustering without desired clusters as input is an NP-hard problem (Section 3.1.2). It maps to the constrained multicast routing problem, where multiple sources intend to send data to multiple destinations via a given constrained network while a certain cost metric should be minimized [51, 52]. Compared to that problem, the input is now reduced as not all BSs need to be assigned to a cluster but only those that are contained in one of the desired clusters, given as additional input parameter.

It has been shown that finding a cost-optimized and latency-constrained multicast routing tree from one source to a set of destinations is an NP-hard problem, too [53]. This problem has to be solved multiple times in a row (once for each desired clusters defined in the input parameters) while taking into account a second resource constraint (link capacity) to solve the overall problem described in this section. Hence this overall problem is NP-hard as well.

3.2.3 Heuristic approach

To avoid the long runtime of solving the problem optimally using the MILP, we also approach the extended clustering problem heuristically. For this, we developed an algorithm that solves the problem with a modified Breadth First Search (BFS). The core concept is to do a BFS from every BS in the network, which yields a BFS tree for each BS. These BFS trees are then checked in an iterative process whether they contain one

or multiple desired clusters and whether they obey the constraints on bandwidth and delay. If clusters are matched and no constraints are violated, the root BS of the BFS tree is a potential cluster controller.

The heuristic itself is not able to deal with intersecting clusters. This means that one BS cannot be part of multiple desired clusters given as input. This situation, however, can happen in reality where multiple UEs need CoMP and they have a common BS in their clusters. This limitation can be circumvented by two means. First, UEs that have similar wireless channel conditions can be grouped. This way, these UEs use one single cluster, which avoids multiple overlapping desired clusters. Of course, this is not always possible. In these cases, the heuristic can be executed multiple times in sequence to process all input clusters such that input clusters in each execution are disjoint.

We assume that desired clusters are selected such that removing even a few BSs results in a clearly degraded performance for the served UEs. This is valid for cluster sizes of up to about 5 BSs in typical urban scenarios [16]. Hence in case the backhaul network does not support all desired clusters, our goal is to fully implement fewer desired clusters completely instead of implementing all desired clusters as good as possible. E.g., instead of realizing only 3 BSs of two desired clusters containing 4 BSs, we prefer to fully realize all 4 BSs of one cluster and only two of the second cluster. This allows to prioritize UEs that really need CoMP such that they actually benefit from CoMP in the end.

Algorithm inputs The inputs for the heuristic are the desired clusters $W_i \in W$, given as sets of vertices, and the network itself as an annotated graph $G = (B, L)$, where each vertex corresponds to a node in the network, i.e., usually a BS or a router, and each edge to a link between two nodes. The annotations of a link $(u, v) \in L$ include the link capacity $\text{l\_cap}_{u,v}$ and the link latency $\text{l\_t}_{u,v}$. A node $b \in B$ is annotated with its required capacity b_cap_b. In addition to these graph annotations, the maximum tolerable round-trip delay $t_{\max}$ between two cooperating BSs is given.

Algorithm overview The overall algorithm consists of several steps that are executed to find the solution. An overview of these steps is given in the following to quickly explain the objective of each step:

a) Maximum-Path BFS
   ▷ *objective*: start modified BFS from all vertices
   ▷ *output*: BFS trees for all vertices

b) Intersect Clusters
   ▷ *objective*: intersect BFS trees and clusters to find matchings
   ▷ *output*: candidate BFS trees for all clusters

c) Back-Track BFS Trees
   ▷ *objective*: recheck constraints on candidate BFS trees
   ▷ *output*: reduced candidate BFS trees for all clusters

d) Intersect Clusters
   ▷ *objective*: recheck if BFS trees match clusters
   ▷ *output*: candidate BFS trees for all clusters

e) Find Best BFS Trees
   ▷ *objective*: compare candidate BFS trees
   ▷ *output*: one BFS tree for each feasible cluster

Maximum-path BFS (a) The heuristic performs a modified Breadth First Search (BFS) individually starting from each node $b \in B$ as root. Compared to a standard BFS, whenever a new tree edge $(u, v) \in L$ is discovered, the constraints for the complete path to the root node of the path are checked. Only if all constraints are fulfilled, the new vertex is added to the BFS tree. Furthermore, the predecessor annotation $p(v)$ is set to u. Additionally, when a new back edge $(u, w) \in L$ is discovered by the BFS, u is stored as an alternative predecessor in $p_{\text{alt}}(w)$.

The result is a set T of $|B|$ BFS trees that only consist of nodes that fulfill all CoMP requirements between the nodes of the tree and the corresponding root node. The check if all constraints are fulfilled is described in Algorithm 3.3. The parameters are the node v newly discovered by the BFS, its predecessor node u, and the BFS tree's root node s. Note that each tree has been checked individually and interrelations of multiple trees are not taken into account.

Algorithm 3.3 checkPathConstraints(u, v, s)

```
p(v) ← u // initialize predecessor variable
c ← 0 // initialize accumulated capacity variable
t ← 0 // initialize accumulated round-trip latency variable
valid ← true // initialize indicator for path validity
while v ≠ s do // follow path from v to s
   c ← c + b_cap_v // increase accumulated capacity for each new vertex on path
   t ← t + l_t_{u,v} + l_t_{v,u} // increase accumulated round-trip latency for each new vertex on path
   if c > l_cap_{u,v} then
      valid ← false // invalidate path if accumulated capacity excesses link capacity
   end if
   v ← u // load next vertex on path
   u ← p(u) // load next vertex on path
end while
if t > t_max then
   valid ← false // invalidate path if accumulated round-trip latency excesses maximum allowed latency
end if
return valid
```

Intersect clusters (b) This step of the algorithm checks for every BFS tree $T_i \in T$, as calculated in step *a)*, if it completely covers one or more desired clusters $W_i \in W$ or if it partially covers one or more of the desired clusters. The result is a list of completely matched clusters $W_i^{\text{match}} \subseteq W$ and a list of partially matched clusters $W_i^{\text{part}} \subseteq W$ for each BFS tree T_i.

Back-track BFS trees (c) The idea of this step is quite similar to the modified BFS in step *a)*. The constraint checking in step *a)* discarded vertices based only on the constraints on their own path to the tree's root node and did not include dependencies among different routing paths. In reality, however, all the desired clusters need to be operated in parallel, i.e., all requirements have to be fulfilled at once. Hence, the constraints need to be checked again for their fulfillment in parallel.

The reason why this is done in a separate step is that only with the result from the previous step *b)* it is possible to define a preference ordering, stipulating which nodes should be checked before others. Preferring nodes from completely matched clusters and after that checking nodes from partially matched clusters is beneficial. This step is implemented using a queue to store the nodes in the preference order. Details are given in Algorithm 3.4.

Algorithm 3.4 backTrackBFSTrees(T)

```
for all T_i ∈ T do
  Q ← ∅ // initialize vertex queue
  for all W_i ∈ W_i^match do // matched complete clusters
    enqueueVertices(Q, W_i) // enqueue all vertices from W_i
  end for
  for all W_i ∈ W_i^part do // matched partial clusters
    enqueueVertices(Q, W_i) // enqueue all vertices from W_i
  end for
  while Q ≠ ∅ do
    v ← dequeueVertices(Q) // get vertex from queue
    u ← p(v) // load predecessor
    s ← root(T_i) // get tree's root vertex
    if CheckPathConstraints(u, v, s) = false then
      remove(v, T_i) // remove v from tree, if constraints are violated
    else
      updateAnnotations(T_i) // update annotations (l_cap, l_t, b_cap) for path
      from v to s
    end if
  end while
end for
```

Whenever a node violates the constraints on its routing path, the node is removed from the BFS tree. If the constraints are fulfilled for all nodes on the path, annotations for all nodes on the path need to be updated: The links' free capacities $\text{l\_cap}_{u,v}$ have

to be decreased by the required capacities of the nodes on the path. The required capacity b_cap_v for each node v on the path has to be set to 0 because their capacity requirements have been fulfilled. The resulting trees only include vertices that properly meet the constraints for real routing paths.

Intersect clusters (d) After removing vertices in the previous step *c)* again, the information on matched clusters (W_i^{match} and W_i^{part}) is not valid anymore. Thus, step *b)* of the heuristic has to be repeated for the BFS trees reduced in the previous step. If a tree completely matches a cluster, the root of the tree is a valid candidate for controlling that cluster. Additionally, partially matched clusters can be used later to analyze why some clusters could not be matched.

Find best BFS trees (e) There might be cases where multiple possibilities exist to realize a certain desired cluster W_i, e.g., because multiple controller locations are possible or because multiple paths exists between BSs. In this case, there is more than one tree contained in W_i^{match} or W_i^{part}. To select a final realization for a certain desired cluster W_i, a total cost for each candidate BFS T_i in W_i^{match} or W_i^{part} is calculated using a cost function, like the resulting energy consumption or monetary costs for required hardware upgrades. After that, the heuristic determines the BFS tree which is the best candidate for each desired cluster. The output is exactly one BFS tree T_i for each cluster W_i, either matching the complete cluster or a partially matched cluster.

Time Complexity Step *a)* performs a BFS for each vertex $b \in B$ in the graph. This results in a runtime as shown in Equation 3.40.

$$\mathcal{O}\left(|B| \cdot (|L| + |B|)\right) \tag{3.40}$$

The BFS is modified to additionally check the CoMP constraints. These checks have a total complexity of $\mathcal{O}(|B|^2)$. Hence, the total time complexity for the first step *a)* is as given in Equation 3.41.

$$\mathcal{O}\left(|B| \cdot (|L| + |B|) + |B|^2\right) \tag{3.41}$$

In steps *b)* and *d)*, sets of vertices are matched for each combination of vertices and clusters. Assuming that the set matching itself is bound by $\mathcal{O}(|B|^2)$, the time complexity of the overall step can be summarized as shown in Equation 3.42.

$$\mathcal{O}\left(|B| \cdot |W| \cdot |B|^2\right) = \mathcal{O}\left(|W| \cdot |B|^3\right) \tag{3.42}$$

Step *c)* re-checks the constraints on the data paths. According to the time complexity already calculated for the constraint checking in step *a)* ($\mathcal{O}(|B|^2)$), the total time complexity for step *c)* is given in Equation 3.43.

$$\mathcal{O}\left(|B| \cdot |B|^2\right) = \mathcal{O}\left(|B|^3\right) \tag{3.43}$$

To calculate the total costs in step *e)*, in the worst case the algorithm has to visit each link $l \in L$ and node $b \in B$ for each remaining BFS tree. Hence, the complexity is bound by Equation 3.44.

$$\mathcal{O}\left(|B| \cdot (|B| + |E|)\right) \tag{3.44}$$

To sum up, as each step is executed sequentially, the worst case total time complexity is bound by $\mathcal{O}(|W| \cdot |B|^3)$, which is the runtime for steps *b)* and *d)*.

3.2.4 Evaluation

As shown in the previous section, the runtime of the heuristic algorithm is cubic in terms of the number of BSs in the input scenario. Hence, it is helpful to compare the heuristic's performance to the performance when solving the problem optimally (or within set bounds of optimality) using the MILP described in Section 3.2.2 in a practically relevant scenario. Furthermore, the MILP solver will always find solution within the given optimality bounds, whereas the heuristic algorithm usually does not. So it is also important to evaluate the heuristic's solution quality.

The evaluation uses the same BS distribution model as used in the evaluation of the previous basic clustering approach in Section 3.1.4. After that, two different ways of interconnecting the BS are considered: For a mesh topology, two BSs are connected by a link if their distance is less or equal to $1.25 \cdot s$, to produce a partially connected mesh network. We have chosen this value as it produces reasonable topologies. Smaller or larger values result in unrealistic (too sparse/dense) topologies. This topology model is identical to the one used in Section 3.2.4. As backhaul networks are often deployed as tree topologies, e.g., using PON technology, we also consider this kind of topology. We connect BSs in the same area according to a defined splitting factor of 9, i.e., 9 BSs are connected to a common node. This means that there are multiple tree structures. All trees' roots are located at a central site and are interconnected – a common practice for network operators.

The number of BSs to which a UE is connected is determined by the UE radius factor r using a disk of radius $r \cdot \bar{s}$ around each UE. For $r = 1$ the radius includes around 3 BSs and for $r = 1.5$ around 7 BSs.

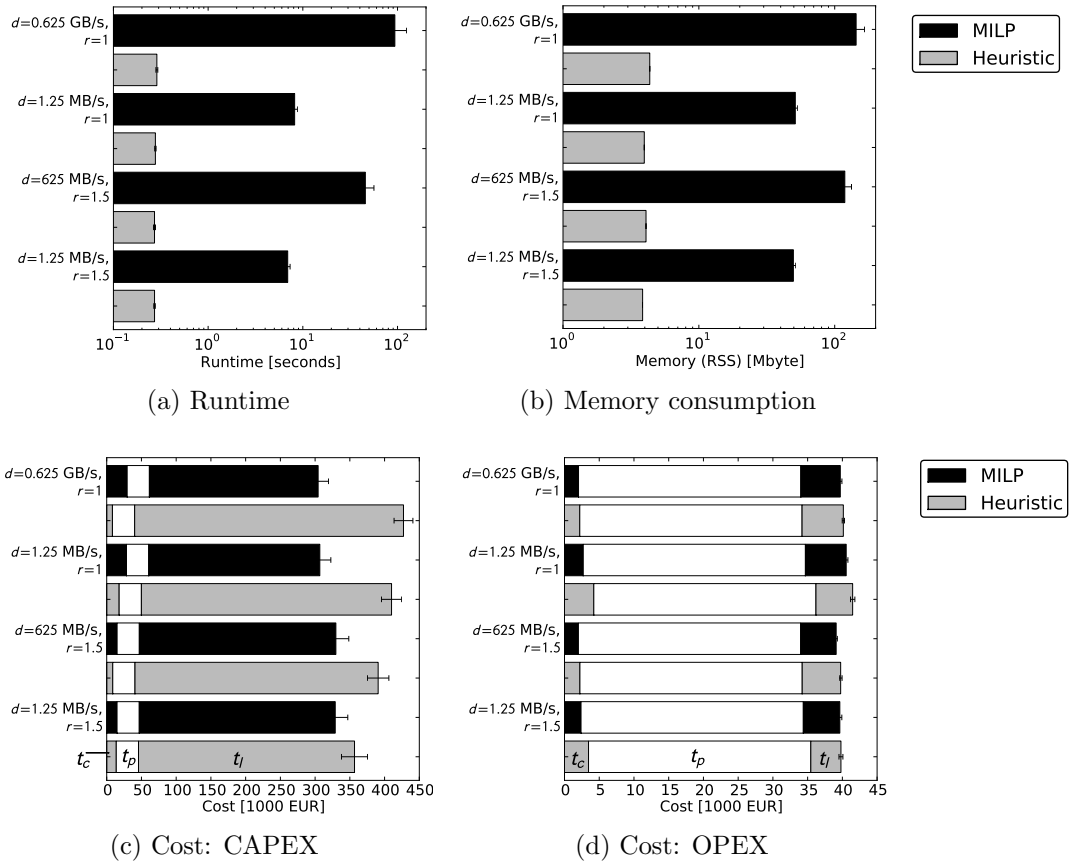
To evaluate the solution quality as well as the runtime and memory consumption of the heuristic, we used a mesh network topology with 16 BS. This number is limited by the 4 GB RAM of the evaluation machine, fully required by the MILP solver. The capacity for each link is 2.5 Gb/s and the capacity demand b_cap_b (was d) for each BS b is set to 1.25 Gb/s or 0.625 Gb/s. A capacity demand of 1.25 Gb/s implies that a maximum of two BSs can be connected to a controller over a single link, the same holds for a demand of 0.625 Gb/s and a maximum number of 4 BSs. The desired clusters are

generated by randomly placing them within the network such that they are disjoint to have equal and fair scenarios for the MILP and heuristic. The cluster radius is calculated by multiplying the mean inter-base station distance $\bar{s}$ with a factor r and clusters are placed as long as there is enough space left.

The solutions returned by the MILP solver and the heuristic need to be rated according a certain cost model. We use monetary costs for the proposed configuration, i.e., costs for the required links, CoMP controllers, etc. For this, we consider the following two models: one for CAPEX, i.e., we consider the application of the tools for network planning where all equipment has to be deployed from scratch. In this case, the CoMP controller costs are set to $t_c = 4$, the additional CoMP controller costs per controlled BS are $t_p = 1$, and link costs are $t_l = 5 \cdot s$. The CAPEX model refers to normalized costs for actually deploying the network, where costs for new links (e.g., optical fibers) are significantly higher than the costs for computing equipment for the CoMP controllers. The absolute costs that have been used for calculating these normalized costs are 17,000 \$ per kilometer fiber deployment [54], 13,600 \$ for a blade center that acts as CoMP controller (including equipment, like a rack and networking components), and 3,400 \$ per additional blade in the blade center that is required for each BS connected to the CoMP controller. These costs are real costs that have been estimated using publicly available prices. The second model reflects OPEX, i.e., the costs that occur for operating a network (mainly for consumed energy) and applying the heuristic online after the network has been deployed. The costs are set to $t_c = 1$, $t_p = 1$, and $t_l = 0.25$. The OPEX model uses normalized costs for operating deployed equipment, where computing equipment will consume more power than the links. The used values correspond to roughly 70 W power consumption per blade center and for each contained blade server [55] and 17.5 W per link, i.e., line card.

Figures 3.7a and 3.7b show the runtime and memory consumption for both the heuristic and the MILP solver. Clearly, the heuristic solves the problem much faster and consumes considerably less memory compared to solving the problem using linear optimization. Both metrics are plotted on a logarithmic scale.

Using the cost model for OPEX, there is also no significant difference in the total cost of the computed result, as depicted in Figure 3.7d on the next page. This means that the heuristic algorithm finds a solution that is nearly identical to the optimal one. For the CAPEX cost model in Figure 3.7c on the facing page, there are notable differences in the total costs for some of the evaluated parameter combinations. The reason is the increased per-link cost in the CAPEX model compared to the OPEX model. The BFS clustering heuristic strictly discovers routing paths with a BFS mechanism, while the optimizer can find arbitrary paths. Especially in the parameter combinations with lower capacity demand b_cap_b (was d) and a smaller cluster radius r, the heuristic tends to operate multiple clusters with a shared controller at the expense of using additional links, while the MILP solver is able to minimize the number of used links. The subbars in Figures 3.7c and 3.7d prove this: The leftmost subbar shows the controller costs t_c, the middle subbar the per-BS cost t_p and the rightmost subbar the link costs t_l. Obviously, t_p has to be equal for both the heuristic and the MILP solver, but the differences in t_c and t_l show how the heuristic and the optimizer work differently.



(a) Runtime (b) Memory consumption

(c) Cost: CAPEX (d) Cost: OPEX

Figure 3.7: Heuristic runtime, memory consumption, and solution quality for different cost models. The subbars from left to right in Figures 3.7c and 3.7d correspond to t_c, t_p, and t_l.

3.3 Summary

In this chapter, I addressed the problem of how to statically position controller functions in the network and how to connect BSs to the clusters to fulfill the backhaul network requirements of the desired CoMP techniques. In the basic version, only backhaul network properties and requirements on the backhaul network, resulting from the CoMP technique to be applied, are taken into account. In the extended approach, also desired BS clusters, which could be selected, e.g., based on wireless measurements, are taken into account in the process.

For both approaches, two different solutions are considered: the optimal one, calculated using MILPs, and the one calculated by heuristic algorithms. While the MILP returns the optimal solution, its long runtime and the high memory requirements turned out to make its application nearly impossible in real scenarios. Only small input scenarios are feasible. On the other hand, the heuristics produce solutions that are only

5 % worse than the MILP's output but have a runtime and memory consumption that is magnitudes lower than those of the MILP.

The proposed clustering methods and heuristics solve individual problems in the wireless domain, like high interference for certain UEs, by applying CoMP according to the capabilities of the limited backhaul network deployment.

Chapter 4

Dynamic clustering and its feasibility

Contents

Static clustering, discussed in Chapter 3, has the disadvantage that it does not fully exploit the potential of CoMP. Hence, dynamic clustering schemes have been proposed [56], where clusters for serving a certain UE are selected based on the wireless channel conditions between BSs and UEs to provide the best possible performance.

On the other hand, using a dynamic clustering scheme in a cellular mobile network is much more complicated compared to static clustering. As clusters are added and

removed based on the current wireless conditions of the UEs in the system, the clusters themselves, the controller functions, and therewith the additional load caused in the backhaul network frequently change over time. This prohibits an a priori planning of the CoMP clusters as their feasibility is unclear due to fluctuating load and latency properties in the backhaul network.

To find out the feasibility of such individual dynamic clusters, I first need an idea on how the number of BSs that participate in a Joint Transmission (JT) to an UE influences the UE's throughput. This evaluation has been done by Changsoon Choi and me by simulating the wireless part of JT without any backhaul network limitations (presented in Section 4.1). From this evaluation, I derive reasonable BS cluster sizes that are desirable from the wireless point of view. Thereafter, I check the feasibility of these cluster sizes from the backhaul network's point of view in a second simulation. Results will be presented in Section 4.2. Here, several backhaul network deployment scenarios, including different topologies and implementation technologies, are taken into account. Finally, Section 4.3 points out the consequences of the mismatches between the two types of clusters, like overhead resulting from infeasible clusters. Moreover, a solution is presented that eliminates all unnecessary overhead and hence better exploits available network resources.

4.1 Desired clusters from the wireless perspective

The effect of exploiting cooperative diversity has been studied in the past and generic analytical models have been developed [57, 58]. Increasing the cluster size in a JT CoMP system increases the UE throughput. The reason is that neighbor BSs that participate in the cluster do not cause interference anymore due to coordination. Hence, the more BSs in the cluster, the less interference at the jointly served UE. However, larger clusters require more backhaul capacity and result in higher signaling overhead. Therefore, it is important to get insights how much gain in terms of UE spectral efficiency can be expected from increasing the number of cooperating BSs. Therefore, we simulate JT CoMP in the evaluation scenarios given by 3rd Generation Partnership Project (3GPP) [59].

4.1.1 Wireless system model

The simulation assumptions and parameters for calculating the UEs' spectral efficiency depending on the number of cooperating BSs are shown in Table 4.1 on the next page.

During the simulation, BSs with hexagonal cells are generated and distributed with distances of 500 m between neighboring BSs. UEs are associated to the BS with the best SINR and are randomly positioned such that one UE is associated to each BS. This is achieved by randomly dropping UEs but skipping those that have the same best BS as an already existing UE. We limit the number of UEs per cell to one to get generic results that are independent of the scheduling mechanism, which would be required in case of multiple UEs per cell. On the other hand, we need at least one UE per cell to

Table 4.1: Wireless simulation parameters

Parameter	Value
Cell layout	61 hexagonal cells
Inter-BS distance	500 m
Frequency reuse factor	1
Number of sectors per BS	1
Number of UEs per BS	1
Maximum transmission power per BS	46 dBm
Antenna pattern for BSs and UEs	Omni-directional (0 dBi)
Bandwidth	10 MHz
Carrier frequency	2.0 GHz
Noise power spectral density	−174.0 dBm/Hz
Noise figure	8 dB
Numbers of antennas ($N_t \times N_r$)	2 (BS) × 2 (UE)
Minimum distance between BS and UE	35 m
Path-loss model	$128.1 + 37.6 \log(d)$, d in km
Penetration loss	0
Shadowing model	Log-normal shadow fading with 0 mean and 8 dB standard deviation
Shadowing correlation	0.5 (inter-BS) / 1.0 (inter-BS)
Small-scale fading model	Rayleigh flat fading
Fading correlation	0

simulate the inter-cell interference from neighbor cells, which should be eliminated by CoMP eventually.

To take into account large-scale fading, we generate the shadow fading map of 61 cells, i.e., 3 rings of cells surrounding the center cell under consideration. For small-scale fading, a Rayleigh flat-fading channel model is applied. We assume full knowledge of instantaneous CSI at the transmitter and synchronization among cooperating BSs.

The cluster of M BSs, which use CoMP to jointly serve a UE u (located in the center cell), is selected according to the SINRs between u and the BSs. As we would like to find out by this simulation how many BSs should reasonably cooperate, we calculate the spectral efficiency for cluster sizes from $M = 1$ (no cooperation) to $M = 16$ BSs. The BSs in the cluster are chosen from the 61 candidate BSs in descending SINR order. Hence, the clusters always contain those BSs that provide u with the best possible SINR.

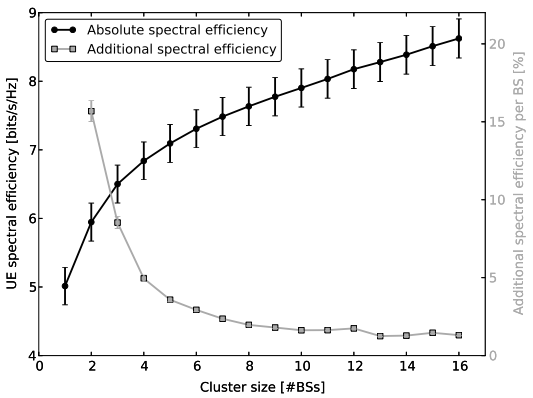
For the cooperative transmission, we use downlink MU-MIMO with equal power distribution in a single cluster [60], i.e., all cluster members send with the same transmission power. Such a cluster consists of the mentioned M BSs where each BS is equipped with N_t transmit antennas. UEs have N_r receive antennas. As only one UE is associated to each cell in this simulation, the number K of UEs per cluster is equal to M. The resulting cooperative BS antenna array, consisting of $M \cdot N_t$ antennas, is coordinated by calculating the precoding matrices using a Block Diagonalization (BD) method [60].

Finally, we calculate the spectral efficiency of the UE located in the center of the 61 cells [61, 35], which is the main metric in this simulation.

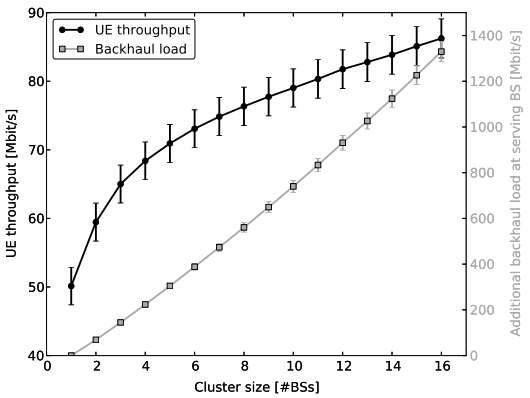
To get statistically relevant results, we repeated the simulation where each run uses different UE locations. It results in different wireless channel conditions and hence also in different wireless clusters. Confidence intervals have been calculated for a confidence level of 95 % and are shown in the plots.

4.1.2 Desired cluster size

The first simulation results are shown in Figure 4.1a. The plot contains two different kinds of information. First, the absolute spectral efficiency per UE averaged over all UEs (black line, left y-axis). And second, the additional relative spectral efficiency of adding an additional BS to the cooperating cluster (gray line, right y-axis). For example, the value for cluster size 3 gives the additional relative spectral efficiency of adding a third BS compared to a cluster of just 2 BSs. Both metrics are shown depending on the cooperating cluster size.



(a) Absolute spectral efficiency and relative gain in spectral efficiency per additional BS depending on cooperative cluster size.



(b) UE throughput and additional backhaul load at the serving BS depending on cooperative cluster size.

Figure 4.1: Cluster size influences UEs' spectral efficiency and BSs' backhaul load.

The curve of the absolute spectral efficiency shows the expected behavior that increasing the number of cooperating BSs leads to a higher average spectral efficiency per UE as interference from neighboring cells decreases. Furthermore, it shows that the additional gain of adding a BS to the cluster decreases when the cluster is already large.

A network operator might not want to accept the high additional overhead, and hence costs, for just a minor increase in performance. The gray curve in Figure 4.1a shows that in contrast to increasing the cluster size when it is small, where additional gains of up to 17 % can be seen when increasing the cluster by one BS, from cluster sizes of 7 and above the additional gain remains nearly constant at about 2 %. This behavior is important to notice because adding more BSs causes the additional overhead in the

backhaul and wireless domain even to increase. The additional gain visible to the UE, however, decreases the larger the cluster already is. This relation between gain and overhead is additionally illustrated in Figure 4.1b on the facing page. The plot shows the average UE throughput assuming 10 MHz bandwidth (Table 4.1 on page 45) and an estimation of the additional backhaul load caused by distributing UE data from the serving BS to the BSs in the cooperating cluster. This estimation is calculated by multiplying the UE throughput by the number of BSs in the cooperating cluster except the serving BS itself. As the throughput steadily increases when increasing the cluster size, the resulting backhaul load is not completely linear but increases a little bit faster. Additional signaling traffic is neglected.

Depending on the network operator's strategy, various cluster sizes are reasonable:

- Small clusters (∼3 BSs): In case the operator is interested in exploiting BS cooperation without causing too much overhead, small clusters are useful as the additional relative gain in spectral efficiency of just adding a few BSs is high. Adding up to 3 BSs to the cooperating cluster gives expected gains of at least 10 % per BS.
- Medium clusters (∼7 BSs): As the gain of additional BSs in the cluster flattens out at a cluster size of roughly 7 BSs, an operator might want to upgrade its network to support clusters only up to this size.
- Large clusters (∼16 BSs): Whenever a high service quality is the ultimate goal, clusters can be increased as long as the network infrastructure supports the additional overhead.

Based on these insights, I often focus on three different desired wireless cluster sizes of 3, 7, and 16 BSs for further evaluations in the remainder of this thesis.

4.2 Feasible clusters from the backhaul perspective

Now that we have information on reasonable desired wireless cluster sizes, the question arises if such clusters are feasible from the backhaul perspective. To get insights into how the implementation of the backhaul network influences the feasibility of the desired wireless BS clusters, I conducted several simulations. I investigate the backhaul network infrastructure from the topological point of view, trying to understand which topologies, like mesh or tree structures, best fulfill CoMP requirements. Furthermore, I look at the used technology, analyzing the different trade-offs offered by available and upcoming backhaul network solutions, like support for layer-2 switching where no Internet Protocol (IP) processing delay occurs or backhaul infrastructures that support single-copy multicast. For these different backhaul infrastructure implementations, I evaluate, for different traffic scenarios and backhaul connectivity levels, which BS clusters are actually feasible compared to the ones desirable from the wireless perspective.

The assumptions I made for the simulations and a description of how I evaluated the wireless cluster feasibility are given in Section 4.2.1. Simulation results and their discussion will be presented from Section 4.2.2 onwards.

4.2.1 Backhaul system model

The simulation consist of the following steps. First, BSs are distributed in the field. Thereafter, links are added between the BSs, i.e., the backhaul network is created. In a third step, desired wireless cooperation clusters are chosen for each BS. These clusters contain neighboring BSs for which we check whether the backhaul network permits cooperation or not, based on the given backhaul link properties. Details on each simulation step will be given in the following subsections.

Placing base stations For each simulation run, 100 BSs are distributed in a square grid of 10x10 BSs. The edges of this grid have a length of $\overline{s} = 500\,\mathrm{m}$, which corresponds to the average inter-BS distance. The positions of the BSs are randomized by shifting their horizontal and vertical position by two normally-distributed random variables with standard deviation $\overline{s}/5$ and zero mean.

Generating backhaul links After having placed all BSs in the field, links are generated to construct a backhaul network. I evaluate different types of backhaul topologies: a mesh-like and different tree-like topologies.

The *mesh* topology is generated by connecting two BSs A and B whenever the distance s_{AB} between them is smaller than a given bound $f_{\mathrm{dens}} \cdot \overline{s}$. The factor f_{dens} is a parameter that defines the link density of the backhaul network. I evaluate the range $0.6 \leq f_{\mathrm{dens}} \leq 1.4$, which, for the chosen BS location shifting with standard deviation $\overline{s}/5$, covers the full spectrum from an almost disconnected to a fully connected mesh topology. Examples are shown at the bottom of Figure 4.2 on page 51.

Generating *tree* topologies is done by first selecting root BSs for the trees. Thereafter, a tree is "grown" from each of these root BSs. Selecting a root BS R happens such that R is not yet assigned to an existing tree and such that R's distance to the already assigned BSs equals the expected cluster radius, i.e., the expected number of hops from the controller R to the farthest BSs in a cluster. This number of hops depends on the average backhaul network link capacity and delay and the requirements of the chosen CoMP technique. The exact calculation is identical to the concept in Algorithm 3.2 on page 23. The tree growing process thereafter is done in d iterations, where d is the desired maximum depth of the tree. In the first iteration, any BSs B_1 is added to the tree (connected to the root BS R) if $s_{RB_1} \leq f_{\mathrm{dens}} \cdot \overline{s}$. This step results in a star topology (tree with $d = 1$). Thereafter, if $d > 1$, the procedure is repeated for all BSs B_1 that have been added in the previous iteration. Links are created from B_1 to a new, *unassigned* BS B_2 if $s_{B_1B_2} \leq f_{\mathrm{dens}} \cdot \overline{s}$. Akin to the mesh topologies, the parameter f_{dens} controls the density of the backhaul network. The order in which BSs are selected while defining the tree structures is randomized. Together with the randomized BS positions, this avoids effects of equally constructed backhaul network trees in different simulation runs. Example topologies can be found at the bottom of Figure 4.3 on page 52.

Independently of the generated topology, all links have the same capacity and latency. Links are optical and have a latency of $s \cdot \frac{1.45}{c}$. Via the link capacity I change the level of overprovisioning in the simulations. To get generic results, I set the capacity of each

link in the topology to $f_{\text{lcap}} \cdot d_{\text{coop}}$, where d_{coop} is the average data rate generated by cooperatively served UEs in the cells. The factor f_{lcap} can be seen as normalized link capacity and is scaled from 1.25 to 10.

Base station cooperation and desired wireless clusters After generating the backhaul network, the scenario contains a field of BSs that are interconnected via a mesh or tree network. To find out how these backhaul networks influence the feasibility of wireless cooperation between the BSs, I need to select BS clusters in which cooperation is desirable from the wireless point of view. The evaluation in Section 4.1 has shown that depending on the network operator's strategy, reasonable desired cluster sizes are between 3 and 16 BSs. Other publications confirm these numbers [12, 62]. Hence, a desired cluster for a certain UE in our simulation consists of the n closest BSs, where $n \in \{3, 7, 16\}$ (Section 4.1.2).

Due to its advantages, as discussed in Section 2.2.1, I assume a semi-centralized implementation of JT within the clusters. UE data of a cooperatively served UE is simultaneously sent from all cluster members. As I focus on the downstream in our simulations, the UE data is forwarded by the serving BS to all its cluster members. For simplicity, the required capacity for the signaling traffic is neglected due to its small volume compared to the data traffic. Delay requirements are taken into account for both the data and signaling traffic.

Evaluating wireless cluster feasibility To find out how the backhaul network influences the feasibility of the desired wireless clusters, I check for each BS in the given input scenario which fraction of the corresponding desired wireless cluster is actually able to participate in the cooperation, considering the limitations of the backhaul network. Averaging the individual fractions for all BSs in the input scenario leads to the main metric, the *wireless cluster feasibility*. The order in which BSs are checked does not matter as only *one* deployed BS cluster is checked at a time, i.e., the clusters do not influence each other. Hence, the results are an upper bound as multiple clusters in parallel would require more resources and cause a lower wireless cluster feasibility.

The actual check whether a certain desired BS B can participate in the cluster around the serving BS S consists of two separate checks. First, the link capacities on the shortest path from S to B are checked. If adding B to the cluster without overloading any link on the path is possible, B can participate in the cooperative transmission from the capacity perspective. In this case, the load on the links is increased by the amount of the cooperative UE data rate.

Thereafter, the latency constraint from S to B is checked. We assume a maximum CSI validity duration of 1 ms, which corresponds to the duration of an LTE subframe. Furthermore, we reserve 0.5 ms for processing and transmitting at the BSs. The remaining portion can be used for the transport from a potential member BSs B to the serving BS S and back, i.e., in our simulations, the Round Trip Time (RTT) from X to B must remain below 0.5 ms. When determining the RTT between S and a potential cluster member B, we take into account the propagation delay in the fiber links and

IP processing and queuing delay at intermediate nodes. The link delays depend on the fiber lengths and are calculated as described in Section 4.2.1. The IP processing delay (including delay for queuing) is set to 0.1 ms per hop [63]. If this check succeeds as well, B is part of the feasible cluster.

4.2.2 Topology influence

The first backhaul network property that influences the wireless cluster feasibility is the way how the BSs are interconnected. I look at two different topology types: a mesh and a tree backhaul topology. We have calculated confidence intervals for all plots for a confidence level of 95 % and show them in the plots.

4.2.2.1 Mesh

The main parameter for the mesh network evaluation is the backhaul network density, controlled by the factor f_{dens} (Section 4.2.1). The factor is varied to result in a very sparsely ($f_{\text{dens}} = 0.6$) up to a very densely connected network ($f_{\text{dens}} = 1.4$). The second factor is the link capacity. It is changed by setting the link capacity factor f_{lcap} to 1.25 and 10. The resulting behavior is shown in Figure 4.2 on the next page.

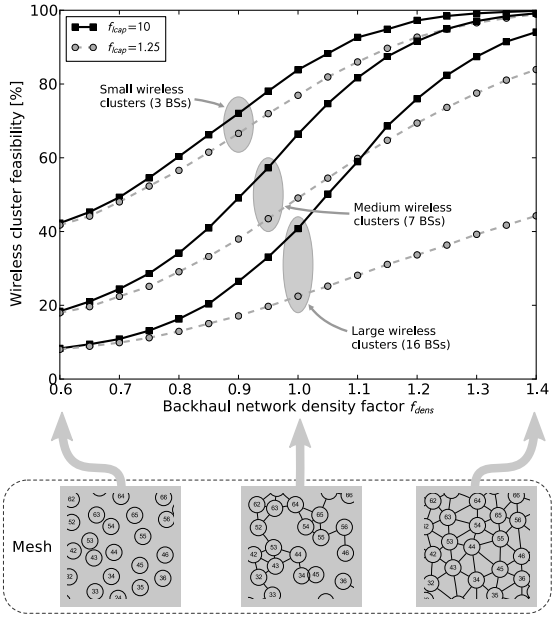
The plots show the expected overall behavior that a more densely connected backhaul network leads to a higher cluster feasibility. In Figure 4.2a on the facing page, the feasibility starts at approximately 10 %, 20 %, and 40 %, depending on the desired wireless cluster size, for $f_{\text{dens}} = 0.6$ (the serving BS is always member of the feasible cluster) and raises up to 100 % for small and medium desired wireless clusters and high link capacity ($f_{\text{lcap}} = 10$). Large clusters are not fully feasible even with a very high backhaul network link density and high link capacity.

For a medium network density ($f_{\text{dens}} \approx 1$), which would be reasonable for a real backhaul deployment, the feasibility is between 20 % and 80 %, depending on the desired wireless cluster size and the degree of link over-provisioning. Increasing the link capacity shows considerable gains only for medium and large desired clusters. In general, increasing the link capacities only improves the cluster feasibility when f_{dens} is higher than 0.8. Otherwise, connectivity between neighboring BSs is insufficient. The required detour increases the delay, which prevents cooperation.

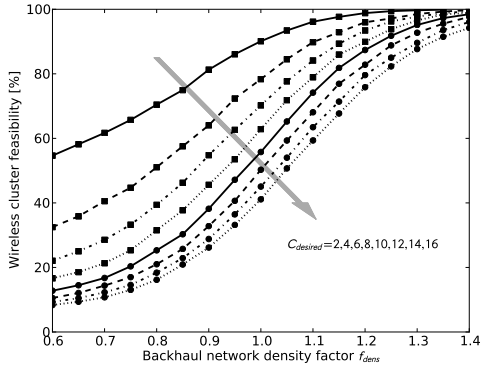
The results show that there is a clear trade-off between the network density, and hence the network costs, and the achieved wireless cluster feasibility.

4.2.2.2 Tree

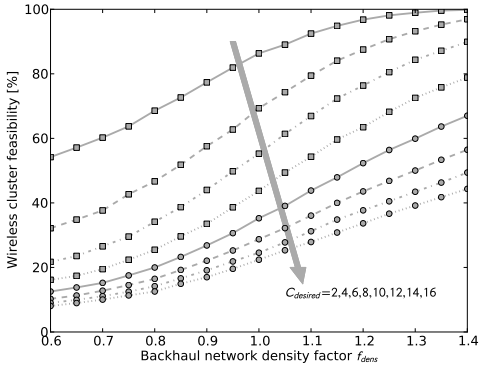
As an optical mesh network is hard to deploy for an operator due to high costs, I evaluated several tree topologies. The first topology I looked at is a tree with a maximum depth of one ($d = 1$), which corresponds to a star. The second tree topology permits a maximum depth of three ($d = 3$). Both can easily be implemented in a real network deployment, e.g., using PON technologies [44]. Studies on technologies that could support the demanding requirements of JP have been initiated, e.g., within 3GPP. They show



(a) Feasibility depending on network density f_{dens} and link capacity f_{lcap} for typical desired wireless cluster sizes. The three topology excerpts below the x-axis illustrate the topology for a density factor of 0.6, 1.0, and 1.4.



(b) Feasibility depending on f_{dens} and desired wireless cluster size C_{desired} from 2 to 16 BSs; $f_{\text{lcap}} = 10$.

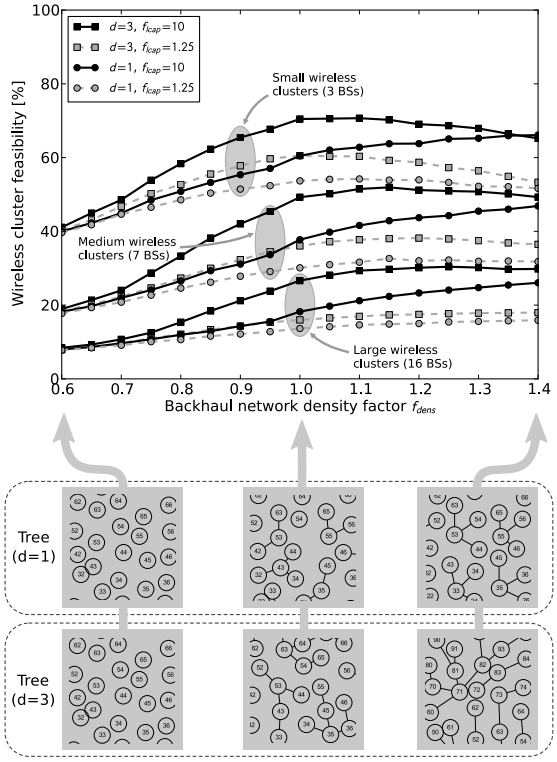


(c) Feasibility depending on f_{dens} and desired wireless cluster size C_{desired} from 2 to 16 BSs; $f_{\text{lcap}} = 1.25$.

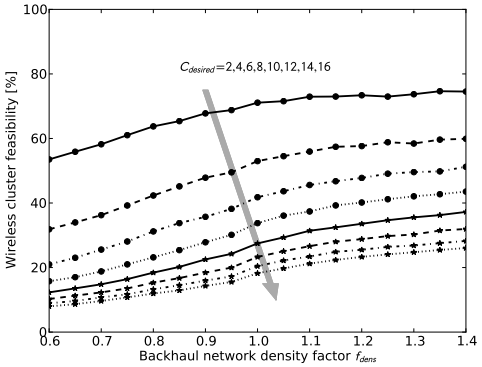
Figure 4.2: Wireless cluster feasibility in a mesh-like backhaul network.

that Passive Optical Networks (PONs) seem to be the most promising technology for the backhaul network. Figure 4.3 on the next page shows the wireless cluster feasibility for the two mentioned tree backhaul network topologies.

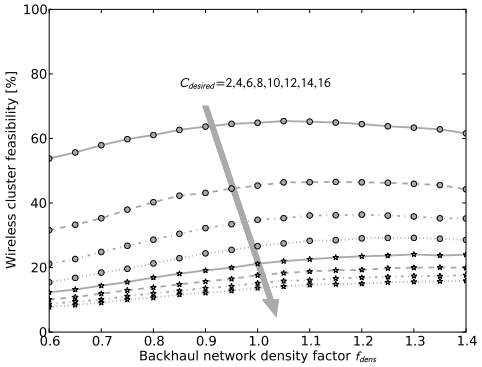
The simulation results show that, compared to the mesh topology, the wireless cluster feasibility in the tree topologies cannot be raised arbitrarily by increasing the link density. As a consequence, independently of the link capacity, the feasibility is limited to approximately 70 %, 50 %, and 30 % for small, medium, and large desired wireless clus-



(a) Feasibility depending on network density f_{dens}, max. tree depth d, and link capacity f_{lcap}. The excerpts illustrate the topology for a density factor of 0.6, 1.0, and 1.4.



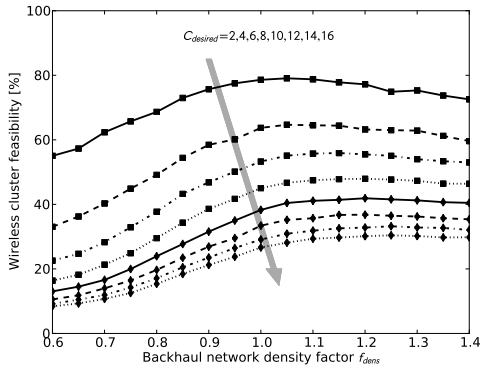
(b) Feasibility depending on f_{dens} and desired cluster sizes C_{desired} from 2 to 16 BSs; $f_{\text{lcap}} = 10$, $d = 1$.



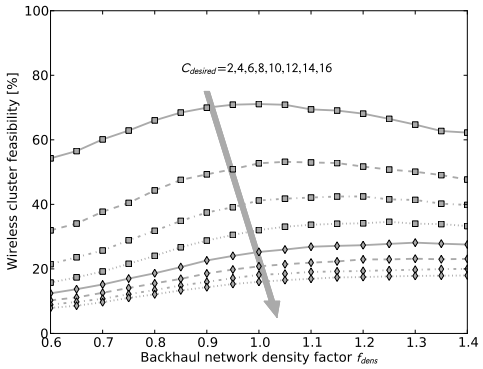
(c) Feasibility depending on f_{dens} and desired cluster sizes C_{desired} from 2 to 16 BSs; $f_{\text{lcap}} = 1.25$, $d = 1$.

Figure 4.3: Wireless cluster feasibility in tree backhaul networks (continued).

ters. The reason for this effect is the lower connectivity compared to the mesh topology. Exchanging data between adjacent BSs often requires a detour via the tree's root node. Although the reduced cluster feasibility is clearly a disadvantage, the reduced network connectivity, and hence reduced network costs, are the positive side of this trade-off.



(d) Feasibility depending on f_{dens} and desired cluster sizes C_{desired} from 2 to 16 BSs; $f_{\text{lcap}} = 10$, $d = 3$.



(e) Feasibility depending on f_{dens} and desired cluster sizes C_{desired} from 2 to 16 BSs; $f_{\text{lcap}} = 1.25$, $d = 3$.

Figure 4.3: Wireless cluster feasibility in tree backhaul networks (continued).

The plots also show that increasing the link capacity by a factor of 8 only marginally improves the cluster feasibility (approximately 10 % at $f_{\text{dens}} = 1$ for medium-sized desired wireless clusters). Here, the delay between the serving BS and potential cluster members is limiting. The link latency also prevents a better cluster feasibility when increasing the tree depth d from 1 to 3 for high network densities (both curves for $f_{\text{lcap}} = 10$ and medium-sized desired clusters end at approximately 50 %).

4.2.3 Influence of IP processing delay

As shown during the topology evaluation, the backhaul network latency plays a critical role in the feasibility of the overall wireless clusters. Independently of the available capacity on the links, the packet processing needed to inspect the IP packet header and making a forwarding decision introduces a delay that is proportional to the number of hops between the serving BS and the candidate cluster member BSs. Hence, I expect that reducing the amount of IP processing steps, which also cause queuing delay, improves cooperation feasibility.

To confirm this expectation, I evaluated the wireless cluster feasibility assuming a layer-2-switched network for each backhaul topology. In this setting, the IP processing occurs only at edge nodes while at intermediate nodes, data is switched at layer 2 between the BSs. Figure 4.4 on the following page illustrates the results of this simulation. The plots show the wireless cluster feasibility for a high backhaul link capacity ($f_{\text{lcap}} = 10$) and a desired wireless cluster size of 16 BSs.

As shown in Figure 4.4 on the next page, the mesh topology benefits most from layer-2 switching (marked as “No IP” in the figure’s legend). This occurs because the multi-hop connections between the BSs experience a lower end-to-end latency, which translates into an extended reach of the cluster. The wireless cluster will thus include those BSs that were not previously admitted due to high multi-hop latency.

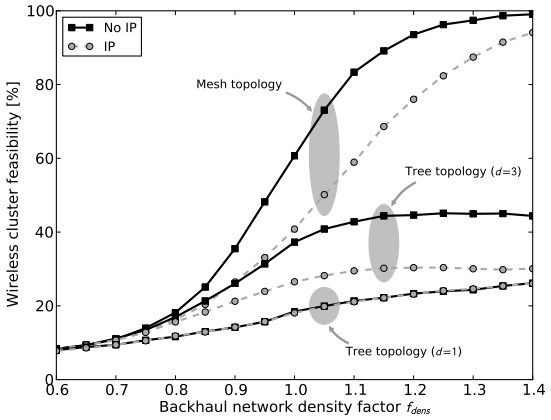


Figure 4.4: IP processing influence on wireless cluster feasibility for $f_{\text{lcap}} = 10$ and large desired wireless clusters with 16 BSs.

For the tree topologies, the feasibility gains depend on the maximum depth d of the trees. This maximum depth inherently introduces an upper limit on the possible number of hops between the cooperative BSs, thus determining an upper bound on the achievable wireless cluster feasibility improvements. In particular, the wireless cluster feasibility gains due to layer-2 switching are high whenever the serving BS is located at the leafs of a tree. In this case, the reduced latency is beneficial, since many hops are required to reach other BSs.

The tree topology with $d = 1$, which corresponds to a star topology, cannot exploit a layer-2-switched backhaul implementation. In this case, the number of hops is inherently limited to 2 and no feasibility gain can be achieved.

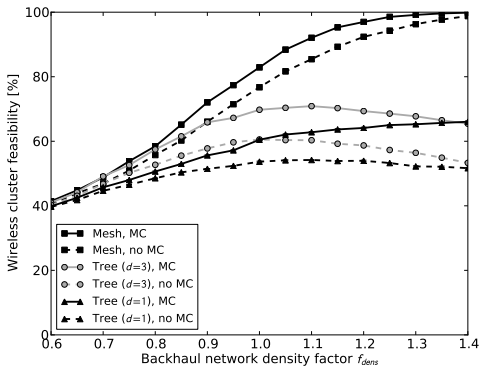
In general, for lower link capacities ($f_{\text{lcap}} < 10$), the gain of using layer-2 switching reduces as the capacity (rather than delay) becomes the bottleneck. The behavior is similar when decreasing the desired wireless cluster size as smaller clusters imply fewer intermediate hops between the BSs.

4.2.4 Influence of single-copy multicast capability

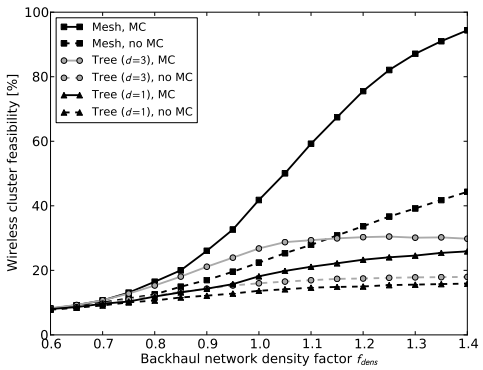
The second insight from the topology evaluations in Section 4.2.2 was that the backhaul link capacity plays a fundamental role for the BS cooperation feasibility as a lot of additional UE data has to be exchanged between the BSs. A possible solution for this is to use a backhaul network technology that supports single-copy multicast, i.e., packets can be copied *on their way* from the serving BS to the cluster members. The serving BS need not send data to all cluster members using multiple unicast flows, saving data rate in the backhaul network. A backhaul network technology that can support single-copy multicast are PONs, especially Time Division Multiplexing (TDM) PONs [44].

Simulation results that illustrate the advantage of using a multicast-enabled backhaul network are depicted in Figure 4.5 on the facing page. The plots are split up into four

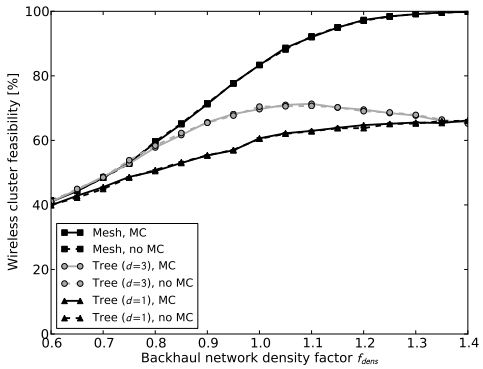
individual plots for small and large desired clusters sizes (3 and 16 BSs) and for low and high backhaul link capacities ($f_{\text{lcap}} = 1.25$ and 10).



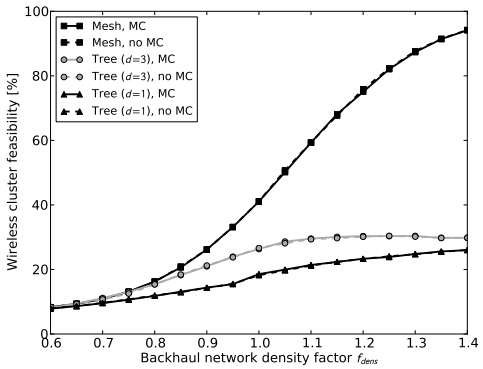
(a) Small desired clusters (3 BSs), low link capacity ($f_{\text{lcap}} = 1.25$).



(b) Large desired clusters (16 BSs), low link capacity ($f_{\text{lcap}} = 1.25$).



(c) Small desired clusters (3 BSs), high link capacity ($f_{\text{lcap}} = 10$).



(d) Large desired clusters (16 BSs), high link capacity ($f_{\text{lcap}} = 10$).

Figure 4.5: Wireless cluster feasibility depending on network density, link capacity, and desired cluster size with (MC) and without single-copy multicast capability (no MC).

The plots show that using single-copy multicast in the backhaul network leads to the highest gains in the mesh topology. The reason is that the probability of two or more independent unicast flows sharing the same link is high and can occur for any placement of the serving BS within the backhaul network. Then, savings in backhaul data rate become high, thus achieving very high wireless cluster feasibility gains.

In the tree topologies, having a multicast-enabled backhaul network is most beneficial in cases where a serving BS is located close to the leafs of the tree. In such cases, the links from the serving BS to the root of the tree are used for transporting multiple unicast flows containing the same data. Multicast capability compresses the unicast flows to one single flow, thus requiring much less data rate. For a serving BS close to the tree's root, multicast does not help that much as the fan out is already high.

In general, the plots show that multicasting reduces capacity problems. In scenarios with high link capacity, single-copy multicast does not lead to any improvement (Fig-

ures 4.5c and 4.5d on the previous page). When capacity is limited, and especially for large desired clusters, the gains are big (Figures 4.5a and 4.5b on the preceding page). Here, the achieved cluster feasibility with low-capacity links is identical to the case with 8 times higher link capacity.

4.2.5 Considerations for cooperation in the uplink

In the simulations above, I focused on JT in the *downlink*. For JP in the *uplink*, the situation changes slightly. Here, data that is sent by an UE is received by multiple BSs. These multiple copies may have errors that occurred during the wireless transmission. To correct these errors, and eventually get a correct copy of the sent data, the copies are sent by the receiving BSs to the controller, where they are combined. The resulting correct data is then forwarded to its destination.

For this procedure, the requirements on the backhaul network are different compared to the downlink cooperation. The latency only plays a minor role, as, compared to the simultaneous sending operation in the downlink, there is no strict synchronization required for combining multiple copies of the received data. On the other hand, capacity requirements for uplink joint processing are higher. The reason for this is that for combining multiple copies, these copies must be sent to the controller as IQ samples [64] or encoded soft-bits [65] to allow for more powerful combining techniques like Maximal Ratio Combining (MRC).

An IQ sample represents a quantized constellation point of a given subcarrier, received on a certain antenna. It is the output of the fourier transformation function of the Orthogonal Frequency-Division Multiplexing (OFDM) receiver chain. In contrast, a soft-bit represents just a single bit but includes additional information on the demodulation process, i.e., the quantized soft value of the coded bit. It is the output of the demodulation function of the OFDM receiver chain.

The amount of additionally required capacity in the backhaul network depends on the resolution that is used for quantizing IQ samples and soft-bits. For the soft-bit-based implementation, which has lower backhaul requirements than the IQ-sample-based solution [65], the additional load is still 5 times higher than the load for downlink cooperation where uncoded bits are exchanged (when using a typical resolution of 5 bits).

Hence, for the uplink cooperation case, the importance of the backhaul network latency on the cooperation feasibility is reduced. On the other hand, link capacity requirements increase by a factor of $\sim$5 at least.

4.3 Handling mismatch between desired and feasible clusters

The evaluations in Section 4.1 and 4.2 have shown that both the desired wireless clusters, determined by the wireless channel conditions, and the feasible clusters, limited by the backhaul network, are quite different. In some situations, e.g., when the backhaul network provides good connectivity, the wireless conditions limit the size of the cluster

eventually used for cooperation, as further increasing the cluster size does not lead to big additional gains. In other cases, e.g., when the backhaul network is weak, the desired wireless clusters are infeasible due to the missing backhaul capabilities. The cluster that can be used eventually for cooperation is the intersection of both the desired wireless and the feasible backhaul cluster.

These two kinds of limitations of the final cooperation cluster size obviously need to be taken into account jointly in a CoMP system. Otherwise, overhead in terms of, e.g., CSI collection for infeasible BSs or backhaul network monitoring for BSs that cannot support the cooperative transmission/reception, complicate CoMP deployment or make it even impossible.

In the following, I propose two components that address the two mentioned cluster limitations. *Wireless long-term pre-clustering* (Section 4.3.1) is a mechanism that estimates which BSs are beneficial to take part in the cooperative transmission/reception from the wireless channels' point of view. The *backhaul network pre-clustering* (Section 4.3.2) permits to predict the feasible cluster from the backhaul network point of view. Both components are then combined to an overall CoMP clustering system architecture in Section 4.3.3.

4.3.1 Wireless long-term pre-clustering

As discussed in Section 2.2.2, CoMP clustering can be categorized into *static* and *dynamic* schemes. Unlike static clustering, where the shape and size of the clusters are pre-defined when designing the cellular network and are fixed over time, dynamic clustering changes and optimizes the cluster shape over time according to the UEs' wireless channel conditions. Dynamically, for each UE that needs CoMP, those BSs form the cooperative cluster that give the best channel quality and hence performance.

Theoretically, the solutions for jointly optimizing clustering based on wireless and backhaul properties presented for the static clustering approach (Section 3.2) could be used for dynamic clustering, too. The problem is, however, that decisions on dynamic clusters need to be done on a time scale of milliseconds, which is not feasible with the presented approaches due to their complexity. Hence, a simpler and more efficient approach is required that also nicely integrates into existing cellular systems.

A good property to base the dynamic wireless clustering on are SINR measurements that are conducted periodically by UEs in LTE and LTE-A systems for hand-over purposes. For this, the UEs measure the SINR of the surrounding BSs. This is usually done based on Received Signal Quality Indicator (RSQI) measurements and is repeated once per LTE frame duration (10 ms). These measurements indicate the average wireless channel quality over this period. Such information comes for free, as it is anyway collected for hand-over decisions, and gives a good indication of the long-term wireless channel quality, compared to expensive CSI, which is only valid for a few milliseconds.

Based on the SINR measurements for a certain UE, the best cluster from the wireless perspective contains those BSs that have the highest impact on the UE, i.e., they have the highest SINR. Hence we form clusters by sorting the BSs in descending SINR order and limiting the number of BSs according to the required gain. This is a simplified version

of the procedure proposed in [66] and is summarized in Algorithm 4.1. Note that this procedure completely disregards limitations by the backhaul network and hence might lead to desired wireless clusters that are not fully feasible in the end. This problem will be addressed in the next section.

Algorithm 4.1 desiredCluster(u, candidateBSs, requiredAggSINR)

```
1: desiredBSs ← ∅ // initialize desired BS set with an empty set
2: aggSINR ← 0 // initialize aggregated SINR
3: BSsOrderedBySINR ← orderDescBySINR(candidateBSs, u) // Order all candidate
   BSs in descending order according to their SINR to UE s
4: while aggSINR < requiredAggSINR do
5:    currentBS ← pop(BSsOrderedBySINR) // get and remove BS with highest SINR
6:    aggSINR ← aggSINR + getSINR(currentBS, u) // add up SINR between UE u and
      the currently inspected BS
7:    append(desiredBSs, currentBS) // add currently inspected BS to desired cluster
8: end while
9: return desiredBSs
```

Algorithm 4.1 gets three inputs: the UE u for which the desired wireless cluster should be found, the set of candidate BSs, and the required aggregated SINR within the cluster. First, the set of desired BSs in the cluster is set to the empty set; likewise, the aggregated SINR is initialized with zero. After that, the candidate BSs are sorted according to their SINR to UE u. This information is available from the regular measurements reported by the UE for hand-over purposes. Now, BSs (ordered from high SINR to lower SINR to UE u) are added to the desired cluster until the required aggregated SINR is achieved. The resulting desired wireless cluster is then returned.

4.3.2 Backhaul network pre-clustering

The evaluations in Section 4.2 have shown that, assuming an optical backhaul network deployment with a realistic density and taking into account backhaul link capacity over-provisioning, there will still be situations where desired wireless clusters cannot be fully implemented due to the backhaul network's connectivity, capacity, and latency limitations. In such situations, choosing BS clusters for cooperation just based on the wireless channel conditions would lead to a degraded performance during the wireless transmission and would cause an unnecessary exchange of signaling and UE data, as the BSs in the cluster eventually cannot cooperate as expected.

To be able to take into account the backhaul conditions when deciding which BSs should cooperate, a mechanism is required that predicts which BSs *can* take part in an upcoming cooperative transmission/reception according to the current conditions in the backhaul network. Based on this prediction, BSs that, e.g., do not have enough backhaul capacity or a too high latency, can be excluded from the cluster of eventually cooperating BSs.

To achieve the goal of finding feasible clusters quickly and taking into account all relevant backhaul network information (capacity *and* latency), I developed the following

algorithm. After monitoring the backhaul network links for their load and latency, a graph representation of the backhaul network is created. This graph is then used to quickly calculate which BSs are able to cooperate whenever cooperating is required.

4.3.2.1 Pre-clustering algorithm

The goal of the following algorithm is to calculate the feasible CoMP cluster for a given serving BS. For this, the algorithm needs the current backhaul network status, i.e., link utilizations and latencies, and the backhaul requirements of the used CoMP technique as input.

The algorithm is based on a modified BFS. The reason why we have chosen a BFS approach is that this algorithm fully explores the close neighborhood of a node before inspecting nodes that are far away. This property is beneficial in the CoMP context as BSs, which are close to the serving BS, are likely to better support the cooperative transmission than those BSs that are far away. This is due to the higher interference these BSs cause at the UE.

Using a graph representation G of the backhaul network with BSs as vertices B and links as edges L, the algorithm FEASIBLECLUSTER starts at the desired serving BS s and iteratively extends the feasible cluster if the backhaul network permits this. A pseudo code version of this procedure is shown in Algorithm 4.2.

Algorithm 4.2 FEASIBLECLUSTER$(G(B, L), s)$

```
1: G_clust ← ∅ // output graph that will contain the feasible cluster
2: Q ← ∅ // queue for BSs that have to be checked
3: for all b ∈ B do
4:    state[b] ← UNKNOWN // initialize state of all BS
5: end for
6: state[s] ← MEMBER // initialize state of serving BS
7: ADDVERTEX(G_clust, s) // add serving BS to cluster
8: ENQUEUE(Q, s) // add serving BS to queue
9: while Q ≠ ∅ do
10:   u ← DEQUEUE(Q)
11:   for all v ∈ ADJACENTVERTICES(G, u) do
12:      if state[v] = UNKNOWN then
13:         if CLUSTEREXTENSIONPOSSIBLE(G_clust, u, v, s) then
14:            // the backhaul network properties permit to add BS v to the cluster
15:            state[v] ← MEMBER
16:            ADDVERTEX(G_clust, v)
17:            ADDEDGE(G_clust, u, v)
18:            ENQUEUE(Q, v)
19:         end if
20:      end if
21:   end for
22: end while
23: return G_clust
```

The algorithm feasibleCluster(G, s) gets the graph G, which represents the backhaul network consisting of the BSs B and backhaul links L, and the vertex s, which represents the serving BS of the UE that needs cooperation, as input. Note that only a small subset of the overall network needs to be provided as input in G. It is sufficient to provide the network part that contains the candidate BSs (typically up to 10 BSs), determined by the wireless long-term pre-clustering step (Section 4.3.1). The output will be a graph G_{clust} that represents the feasible cluster of BSs, which are able to participate in the cooperative transmission/reception from the backhaul network perspective. The graph G_{clust} is always a subgraph of G.

In the first step, the output graph G_{clust} and the queue Q are initialized. Furthermore, the state of all BSs in the input graph G is set to UNKNOWN. This means that it has not been checked yet if they can participate in the cooperation or not. As the serving BS is always member of the feasible cluster (the backhaul network is not involved), its state is set to MEMBER and it is added to the output graph G_{clust}. After that, it is added to queue Q to start the feasibility check from this BS in the next steps.

Now, while there are BSs in Q, the BS u that spent the longest time in the queue is removed. This BS u is always member of the feasible cluster. The backhaul network to each neighbor v of u that is still in the state UNKNOWN is now checked. If v and the link $u \rightarrow v$ fulfill the requirements for participating in the cooperation, the status of v is set to MEMBER, v and the link $u \rightarrow v$ are added to the output graph, and v is enqueued to Q. The method clusterExtensionPossible($G_{\text{clust}}, u, v, s$) checks whether a new BS v can be added to the current feasible cluster, represented by G_{clust}, using the link from BS u to v while fulfilling all requirements. This checks for, e.g., link loads and the latency between the serving BS s and v. The detailed pseudo-code is shown in Algorithm 4.3 on the facing page. The feasibility check in feasibleCluster is repeated until the queue Q is empty, which means that no additional BS can be added to the feasible cluster without violating the backhaul network requirements.

The function clusterExtensionPossible checks if BS v can be added to an existing feasible cluster, given by G_{clust}, with CoMP controller s. The new BS v is connected via the link (u, v), where u is already a member of the feasible cluster. After initializing the used variables, the algorithm walks from the new BS v up to the tree's root node s, which is the CoMP controller of the existing feasible cluster. On this way, free link capacities are checked if they are sufficient to transport the additional load from the controller s to the new BS v (downlink) as well as in the uplink direction and occurring round-trip delays are summed up for each traversed link. Based on the accumulated delay, the fulfillment of the latency constraints are checked when s has been reached.

4.3.2.2 Algorithm complexity and stability

The proposed clustering algorithm's space and time complexities are similar to those of the conventional BFS algorithm. The space complexity is $\mathcal{O}(|V|)$, where V is the set of BSs in the input graph G.

The time complexity is $\mathcal{O}(|V| + |E|)$, with E denoting the number of edges in G. This corresponds to the number of links in the backhaul network. As efficient parallel

Algorithm 4.3 clusterExtensionPossible($G_{\text{clust}}(B, L), u, v, s$)

```
predecessor[v] ← u // set predecessor for new node
t_totalRT ← 0 // initialize accumulated round-trip latency
req_cap_down ← b_cap_down[v] // get required downlink (s to v) capacity
req_cap_up ← b_cap_up[v] // get required uplink (v to s) capacity
constraints_fulfilled ← true // initialize return value
while v ≠ s do // follow path from v up to s
    t_totalRT ← t_totalRT + l_t[u, v] + l_t[v, u] // increase accumulated round-trip delay
    if req_cap_down > l_cap_free[u, v] or req_cap_up > l_cap_free[v, u] then
        // link capacity is exceeded
        constraints_fulfilled ← false
    end if
    // step up one link towards s in the tree
    v ← u
    u ← predecessor[u]
end while
if t_totalRT > t_max then
    // accumulated round-trip delay exceeds allowed threshold
    constraints_fulfilled ← false
end if
return constraints_fulfilled
```

implementations of BFS algorithms are available [67], which scale nearly linear with the number of available processors, a fast implementation of our algorithm is easily feasible. This will be confirmed by the measurements in the following section.

I propose to implement the adjacentVertices function to return the BSs in descending order of the SINR to UE under consideration. This way, BSs are prioritized that provide higher gains and available backhaul resources are spent for such BSs first. This provides high performance gains under current backhaul conditions. Furthermore, this gives a deterministic ordering in which the BSs are checked.

In an LTE or LTE-A system, the smallest scheduling interval is a sub-frame, which lasts 1 ms. This is also the smallest time interval in which CoMP decisions can be made. Hence, in the worst case (depending on wireless channel and backhaul network stability), desired CoMP clusters change for each sub-frame. Consequently, the cluster feasibility has to be re-evaluated every millisecond, too, and the results of the clustering algorithm need to be available before the sub-frame ends. Due to the low time complexity and efficient implementations on current hardware (cf. Section 3.1.4, where multiple executions in a row of a similar algorithm haven been shown to complete clearly faster than 1 ms), the proposed heuristic permits to evaluate desired clusters up to this worst-cast time scale.

4.3.2.3 Algorithm performance and output optimality

This section discusses two important aspects of the backhaul network pre-clustering algorithm: its actual runtime on current hardware and the quality of its output compared to the optimal solution.

Runtime and memory consumption Compared to solving the problem using mathematical optimization, the proposed algorithm has the advantage that it has a much shorter runtime and much lower memory requirements. For input scenarios consisting of 50 BSs, it finishes the clustering below 1 ms on current hardware (single 3.33 GHz CPU), compared to roughly 20 hours when finding the solution via an MILP (using an optimality gap of 10 %, i.e., the returned solution of the MILP is at most 10 % worse than the optimal solution). The memory consumption is reduced from approximately 1 GB to 40 MB when using the proposed heuristic. These performance improvements in runtime and memory consumption enable to use the backhaul pre-clustering online, i.e., while the mobile network is operating, for dynamic CoMP clustering systems.

Output quality compared to optimal solution Although the algorithm is fast enough to be used in real systems, the quality of its output compared to the optimal solution has to be confirmed. To get an idea of the cluster quality, we compared the output clusters returned by the proposed heuristic and the (nearly) optimal ones from the MILP. The MILP optimizes the number of BSs controlled by a given serving BS without violating the capacity and latency constraints. For details on the MILP please refer to Section 3.2.2. Note that for these evaluations, the optimality gap of the MILP solver has been set to 10 %. Lower values, i.e., better solutions, were not possible due to the hardware limitations of the machine running the MILP solver.

For the tree topologies, the heuristic always found the same clusters as the MILP throughout all parameter combinations that have been evaluated in Section 4.2. When applying the algorithm in mesh topologies, the output clusters are smaller than the optimal ones. Details on the cluster feasibility ratio, i.e., the ratio between the cluster size of the algorithm and the optimal one, are given in Figure 4.6 on the next page. The evaluation for $f_{\text{lcap}} = 1.25$ has been omitted due to the long runtime required to solve the MILP. Instead, an intermediate link capacity of $f_{\text{lcap}} = 2.5$ is included. Furthermore, small clusters consisting of 3 BSs are omitted in the plot as here both the MILP and the heuristic algorithm always found identical solutions.

The plot shows that while for medium wireless cluster sizes (consisting of 7 BSs) and high link capacities ($f_{\text{lcap}} = 10$) the ratio is nearly 100 %, the ratio decreases by up to 20 % for large wireless clusters and smaller link capacities.

As, due to cost reasons, real backhaul networks will probably be implemented based on tree topologies or sparse to medium-dense mesh topologies, the algorithm is well-suited for backhaul network pre-clustering in real networks.

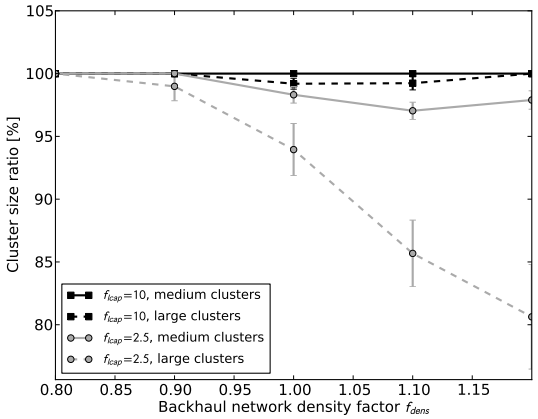


Figure 4.6: Ratio of cluster sizes returned by the backhaul network pre-clustering algorithm (Algorithm 4.2 on page 59) compared to the optimal cluster size returned by the MILP solver.

4.3.3 System architecture

With the two presented techniques of wireless long-term pre-clustering and backhaul network pre-clustering, both the expected desired clusters as well as the feasible clusters can be predicted. In a real-world CoMP system, these two components need to be combined in a beneficial way such that overhead is avoided and CoMP transmission/ reception can be exploited best. To achieve this, Luca Scalia, Changsoon Choi, and I propose a system architecture as illustrated in Figure 4.7.

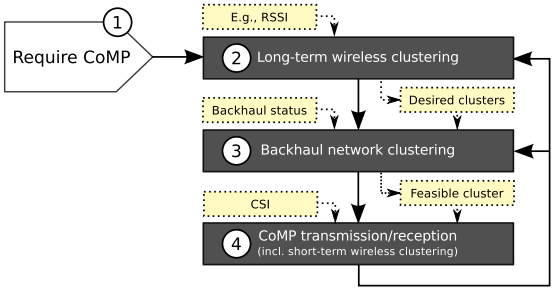


Figure 4.7: System architecture combining wireless and backhaul network clustering to reduce overhead. Solid arrows indicate the application flow, dotted arrows stand for information flow.

After detecting that CoMP is required in Block (1), e.g., because the requested service quality of a UE cannot be fulfilled, the desired wireless clusters are determined. This is done in (2) and requires information about the wireless channel conditions, like Channel Quality Indicator (CQI) or Received Signal Strength Indicator (RSSI) measurements, as described in Section 4.3.1. There are several methods for deciding on the desired clusters [66]. This step is done first in our architecture as long-term wireless channel quality measurements from each UE are available "for free" at the BSs. Such

measurements are collected anyway in cellular systems, like LTE or LTE-Advanced, to make hand-over decisions. Furthermore, overhead for measuring and collecting CQI or RSSI values are much lower compared to CSI.

Note that Block (2) can either return one single desired cluster for the UE or a set of clusters. In case multiple clusters are returned, they should be ordered by their priority, i.e., the expected gain from each of the desired clusters, to ease further processing.

The resulting desired wireless clusters are used together with the backhaul network's current status, like link loads and latency, to calculate which part of the desired clusters is feasible. In case there are multiple desired clusters for a UE, the clusters with higher priority are checked first until a feasible cluster has been found. This happens in (3) and is implemented using the backhaul network pre-clustering algorithm presented in Section 4.3.2. We do this step at this point because desired clusters are available as input for free. This allows to limit the amount of backhaul network parameters that have to be monitored to only those BSs that are contained in the desired cluster. This reduces overhead for calculating the feasible clusters.

Finally, the short-term wireless clustering, sometimes referred to as "finding the active set of BSs", and the actual cooperative transmission/reception is done in Block (4). As the feasible clusters are known from step (3), CSI required for deciding which BSs eventually cooperate and required for the cooperative transmission/reception itself, is only collected for the BSs that really *can* cooperate. This way, collecting and processing CSI, which is the most challenging overhead as it requires a lot of wireless and backhaul resources, is reduced to a minimum.

This overall procedure has to be repeated whenever the desired wireless cluster, i.e., the long-term wireless channel conditions, or the cluster feasibility, i.e., the backhaul network load and latency, change more than a defined threshold. In these cases, the corresponding clustering steps have to be done again, as indicated by the arrows on the right side of Figure 4.7 on the previous page.

4.3.4 Advantages

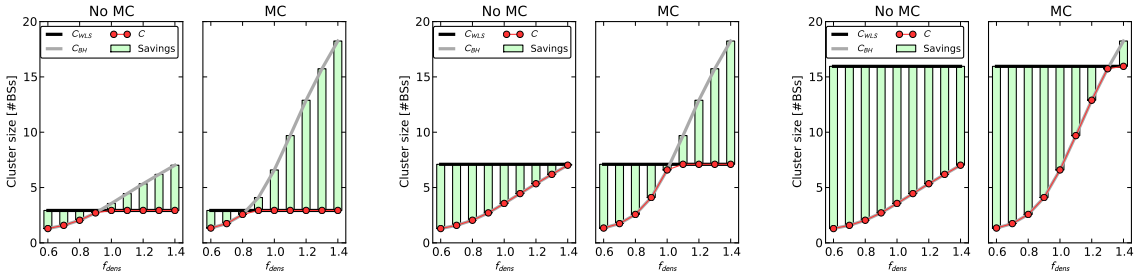
The CoMP system architecture introduced in Section 4.3.3 contains three different clustering steps: first long-term wireless pre-clustering, then backhaul network pre-clustering, and finally the short-term wireless clustering. Each of these steps takes into account different parameters which limit the BSs that can take part in the cooperative transmission/reception. Depending on the scenario in which CoMP is applied, the two first pre-clustering steps might be partially superfluous as the parameters that are taken into account sometimes do not limit the cluster feasibility. For example, in a scenario with a highly over-provisioned backhaul network, the backhaul network clustering step will not further decrease the size of the desired cluster determined by the long-term wireless clustering in the first step. It does, however, cause overhead to execute this step, which might not be desired in such cases.

To see the importance of the clustering steps and to see in which scenarios they are required and in which not, I evaluated the cluster sizes for each step individually. The results for the mesh network are shown in Figure 4.8 on the facing page.

The plots in Figure 4.8 show two kinds of information. First, absolute cluster sizes in terms of number of BSs they contain. There is one line for the long-term wireless cluster size C_{WLS} (using 3, 7, and 16 BSs in the corresponding subfigures) and another line for the average feasible cluster size C_{BH} as determined by the backhaul network pre-clustering. For both cluster sizes it is shown how they depend on the backhaul link density f_{dens} and the availability of single-copy multicast capability (*No MC*, *MC*). Furthermore, according to our clustering mechanism, the final cluster size C is the minimum of C_{WLS} and C_{BH}. A line for the cluster size C is also included in the plots for clarity. The second kind of information in the plots is the resulting cluster size reduction S, i.e., overhead savings when using our proposed clustering mechanism. It can be calculated according to Equation 4.1. It is the difference between the two cluster sizes C_{WLS} and C_{BH}.

$$S = \max(C_{\mathrm{WLS}}, C_{\mathrm{BH}}) - \min(C_{\mathrm{WLS}}, C_{\mathrm{BH}}) \tag{4.1}$$

The savings S are illustrated in the plots via the bars between C_{WLS} and C_{BH}. Savings are large when the desired wireless clusters differ greatly from the feasible ones.



(a) Small wireless clusters (b) Medium wireless clusters (c) Large wireless clusters

Figure 4.8: Cluster sizes from long-term wireless and backhaul network pre-clustering and resulting reduced overhead when using the proposed system in a mesh network scenario. Numbers on the x-axis indicate the backhaul network link density factor f_{dens}.

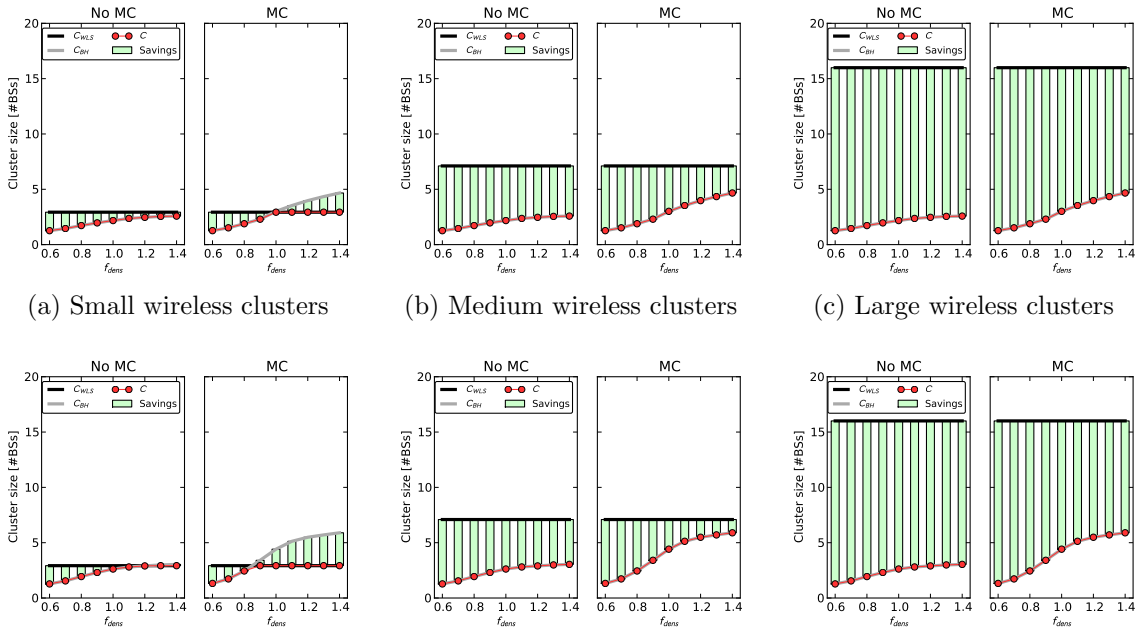
For a sparse backhaul network deployment ($f_{\mathrm{dens}} \approx 0.75$), the backhaul limits the final cluster size. This means that the long-term wireless clustering limits the scope of the backhaul clustering only (which is just done within the desired long-term wireless cluster). The final CoMP cluster size in not influenced. However, as the long-term wireless clustering is "for free", this is not problematic. On the contrary, it simplifies the backhaul clustering step. This step is definitely required.

In medium-dense backhaul networks ($f_{\mathrm{dens}} \approx 1.0$), a similar behavior as for the sparse backhaul scenario is visible. However, for small desired wireless clusters, the final cluster size is now limited by the wireless parameters, not the backhaul network. This is the case especially for the multicast-enabled backhaul network, where feasible clusters from the backhaul network perspective has nearly twice the size of the desired wireless cluster.

The dense backhaul network ($f_{\text{dens}} \approx 1.25$) causes the feasible clusters from the backhaul perspective to get even larger. Hence, the backhaul network does not limit the final cluster size for small desired wireless clusters. For the multicast-enabled backhaul scenario, even the medium clusters are feasible. In these cases, the backhaul clustering step is shortened as it stops when the desired wireless cluster has been found to be feasible and no more BSs have to be checked.

Overall, the long-term wireless pre-clustering is in particular beneficial in scenarios where the backhaul network capabilities are no bottleneck or only small clusters are desired. The long-term wireless pre-clustering clearly reduces the amount of CSI that has to be collected and processed. On the other hand, in scenarios with a rather limited backhaul network the backhaul network pre-clustering plays the dominant role as it determines the final cluster size and therewith the CSI overhead reduction. If a network is only operated in one of these two regimes, it is possible to omit one of the two pre-clustering steps as described. Nevertheless, the long-term wireless pre-clustering comes "for free". And as it simplifies the backhaul network pre-clustering, which is likely to be required anyways due to the fluctuating backhaul properties in real network deployments, these evaluations confirm the benefits of the overall system architecture.

I did a similar evaluation for the tree backhaul topologies with maximum depth $d = 1$ and $d = 3$. The resulting plots are shown in Figure 4.9.



(a) Small wireless clusters (b) Medium wireless clusters (c) Large wireless clusters

(d) Small wireless clusters (e) Medium wireless clusters (f) Large wireless clusters

Figure 4.9: Cluster sizes from long-term wireless and backhaul network pre-clustering and reduced overhead when using the proposed system architecture in a tree network scenario (in top row $d = 1$, bottom $d = 3$). Numbers on the x-axis indicate the backhaul network link density factor f_{dens}.

The behavior for the two tree backhaul network topologies is qualitatively comparable to the mesh backhaul network scenario in Figure 4.8 on page 65. Due to the lower connectivity, however, the backhaul network limitations play a more important role than the limitation by the long-term wireless pre-clustering. Except for the small desired wireless clusters in the dense scenarios ($f_{\text{dens}} \geq 1.0$), the final cluster is always determined by the backhaul network pre-clustering step.

This shows that in backhaul-limited scenarios backhaul network pre-clustering is essential. But instead of omitting the long-term wireless pre-clustering in such scenarios, it is even more important to base the backhaul pre-clustering on the result of the long-term wireless pre-clustering. The reason is that the limited backhaul resources should be exploited for making BSs cooperate that really contribute to a better service quality of the UE. Hence, checking those BSs in the desired cluster that have better wireless channel conditions first for their backhaul capabilities is beneficial to avoid wasting resources for adding BS to the cluster that have worse wireless channel conditions.

4.3.5 Interaction with hand-over process

CoMP reduces interference in cellular networks, especially where a frequency reuse factor of 1 is applied. In such networks, high interference is in particular present for cell-edge users, i.e., such users are likely to use CoMP to reduce interference. At the same time, when the interference power exceeds the received signal power, a UE needs to change its serving cell to another, neighboring BS that provides the highest signal power. This is achieved through the hand-over procedure. After the hand-over procedure, the UE is still located at the cell edge, just on the other side of the cell border, and still needs CoMP to combat the high interference from the neighboring cells, including the old serving cell from which it has been handed over. Hence, it is very likely to have a hand-over process for mobile UEs while CoMP is applied. Therefore, it is important to consider the interrelation of the hand-over process and our proposed CoMP mechanisms.

During hand-over, UE traffic is forwarded from the old serving cell to the new cell. As soon as the hand-over process is completed, the UE data is re-routed to go directly to the new serving cell. If CoMP is applied during the hand-over process, the desired wireless cluster for the UE probably does not change because most of the interfering cells are the same. On the other hand, the CoMP cluster that is feasible from the backhaul network perspective can change completely. This especially occurs for SU-MIMO JT and DCS as the traffic in the backhaul network changes due to the UE data re-routing and the moving of the CoMP controller function to the new serving cell. For this reason, triggering the hand-over procedure for an UE that is served cooperatively should be linked to an immediate backhaul network pre-clustering step at the new serving cell to check whether CoMP is still feasible without impairing other UEs' transfers. This step could even be done proactively when a hand-over procedure is expected soon.

4.4 Cluster prioritization

Results in Section 4.2 have shown that not all desired clusters are feasible in future cellular networks due to the backhaul network limitations. One question arises from this observation: Given a certain amount of backhaul resources, like capacity, which clusters should be realized and which not. In particular, is it better to implement many small clusters or should one implement a few large clusters.

To get insights on how to spend available backhaul capacity to possible JT CoMP clusters, I use the simulation outcome of Section 4.1, which gives the expected downlink cell throughput depending on the number of cooperating BSs $r_{\text{WLS, BS, down}}(c)$ (can be read from Figure 4.1b on page 46). Based on these numbers, I can calculate metrics, like total system throughput, for a certain clustering of a set of BSs.

The considered scenario consists of 16 BSs that would like to cooperate to jointly serve all their UEs. The backhaul network provides a total capacity C_{BH}, which only supports a limited amount of cooperation. This capacity can be used to realize different cluster combinations, as long as it is not exceeded.

The total wireless throughput of a cluster containing c BSs can be calculated as shown in Equation (4.2). It is the individual cell throughput (at cluster size c) times the number of BSs c participating in the cooperation. Note that although it might look like $r_{\text{WLS, cluster, down}}(c)$ is linear in c, this is not the case as $r_{\text{WLS, BS, down}}(c)$ neither constant nor linear in c (cf. Figure 4.1b on page 46).

$$r_{\text{WLS, cluster, down}}(c) = c \cdot r_{\text{WLS, BS, down}}(c) \tag{4.2}$$

The corresponding required backhaul capacity consists of the required uplink and downlink data rates occurring at each BS in the cluster. The downlink backhaul data rate at each BS $r_{\text{BH, BS, down}}$ can be calculated according to Equation (4.3); it is identical to the total UE throughput in the cluster. The uplink backhaul rate is less as the data for the UE in its own cell is already available and has to be forwarded to $c-1$ other BSs (Equation (4.4)). This forwarding is necessary as MU-MIMO requires the data of all jointly served UEs at all cooperating BSs.

$$r_{\text{BH, BS, down}}(c) = r_{\text{WLS, cluster, down}}(c) = c \cdot r_{\text{WLS}}(c) \tag{4.3}$$
$$r_{\text{BH, BS, up}}(c) = (c-1) \cdot r_{\text{WLS}}(c) \tag{4.4}$$

In total, a cluster of size c causes backhaul load as calculated in Equation (4.6).

$$r_{\text{BH, BS, total}}(c) = r_{\text{BH, BS, down}}(c) + r_{\text{BH, BS, up}}(c) \tag{4.5}$$
$$r_{\text{BH, cluster, total}}(c) = c \cdot r_{\text{BH, BS, total}}(c) \tag{4.6}$$

Based on this, I can now calculate the achieved wireless downlink system throughput $r_{\text{WLS, cluster, down}}(c)$ and the corresponding required backhaul load $r_{\text{BH, cluster, total}}(c)$ for a

cluster of size c. In a scenario consisting of multiple clusters, the rates for individual clusters can be summed up to get the total system data rates.

From the equations and the behavior in Figure 4.1b on page 46, it is visible that larger clusters lead to higher wireless system throughput but at the same time to much higher required backhaul capacity. To find out whether it is better to deploy many small or spend all available backhaul capacity for implementing a few large clusters, I defined a maximum allowed cluster size $c_{\max}$, with $1 \leq c_{\max} \leq 16$. Then, in the given scenario of 16 BS, I deployed as large clusters as possible (ensuring that $c \leq c_{\max}$) while varying C_{BH}. This gives an overview on which total wireless system throughput can be achieved depending on the maximum allowed cluster size $c_{\max}$. Results are plotted in Figure 4.10.

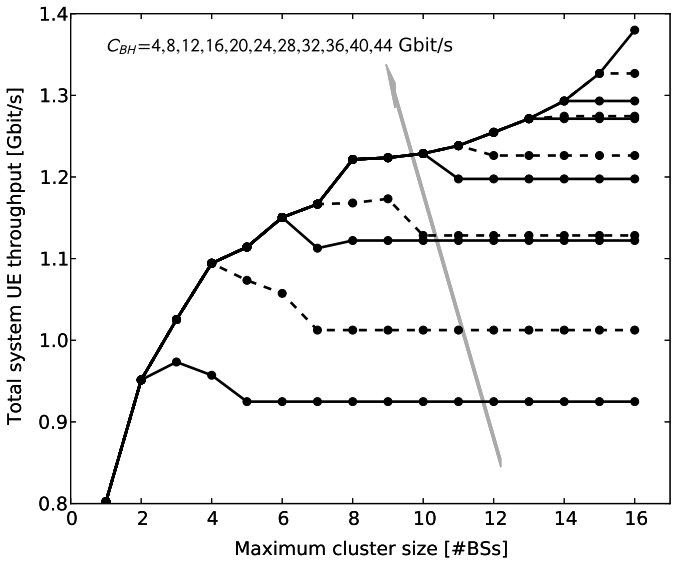


Figure 4.10: Wireless system throughput depending on maximum allowed cluster size $c_{\max}$ and total available backhaul capacity C_{BH}. The total required backhaul capacity for implementing one single cluster with all 16 BSs is $\approx 43\,\mathrm{Gbit/s}$. Without cooperation, the required capacity is $\approx 0.8\,\mathrm{Gbit/s}$.

The plot shows two important behaviors. First, preferring large clusters in favor of small clusters does not lead to a high total system throughput in general. Especially in scenarios with very limited backhaul capacity, the throughput is higher when not implementing the largest possible clusters. An example is the case where $C_{\mathrm{BH}} = 8\,\mathrm{Gbit/s}$. This assumption allows clusters up to the size of 7 BSs, but the system throughput is maximized when using clusters not larger than 4 BSs.

The second important insight is that if the backhaul network provides more than approximately 50 % of the required capacity for full cooperation ($\approx 25.5\,\mathrm{Gbit/s}$ in Figure 4.10), the curves become monotonic, i.e., larger clusters always lead to better system throughput on the wireless side.

Figure 4.11 on the next page gives an impression on the total backhaul network load versus the achieved wireless system throughput. The parameters are identical to

Figure 4.10 on the preceding page, i.e., available backhaul capacity C_{BH} is spent for clusters of different size, depending on $c_{\max}$.

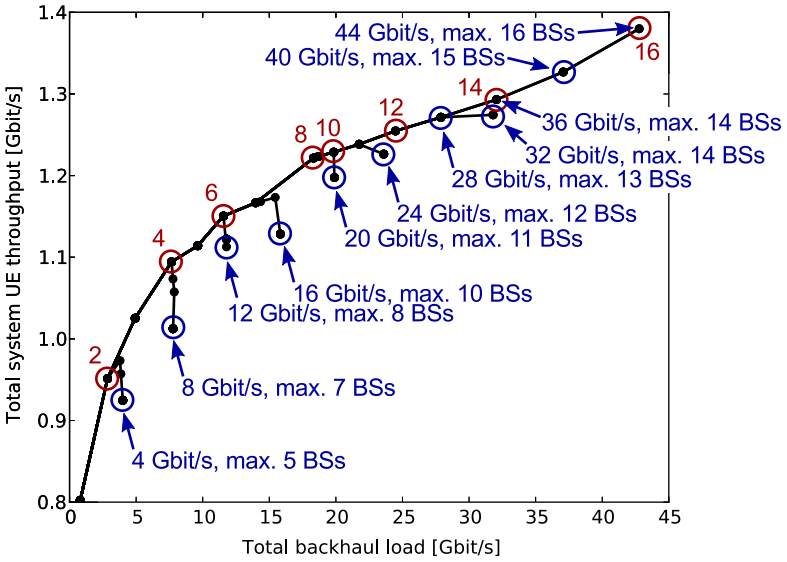


Figure 4.11: Wireless system throughput versus occurring total backhaul load for different total backhaul network capacities $C_{\mathrm{BH}} \in \{4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44\}$ Gbit/s.

The plot shows the interdependence between wireless system throughput and corresponding backhaul load. The topmost line shows the case when the backhaul network does not limit CoMP ($C_{\mathrm{BH}} = 44\,\mathrm{Gbit/s}$). The red circles and numbers from 2 to 16 denote the corresponding cluster size c for the marked dot. The blue labels give the maximum feasible cluster size for the given backhaul capacity C_{BH}. The additionally dots marked in blue show the corresponding system throughput and backhaul load.

Figure 4.11 confirms that implementing larger clusters, and hence spending more backhaul capacity, does not always pay off. Higher throughput can be achieved at slightly lower backhaul load for $C_{\mathrm{BH}} \in \{4, 8, 12, 16, 20, 24\}$ Gbit/s.

The plot also clearly points out that the ratio between required data rate in the backhaul network and achieved throughput data rate on the wireless side is not linear. While the ratio is roughly 5 for small clusters, it increases to approximately 30 for large clusters with 16 BSs.

4.5 Summary

This chapter has investigated dynamic clustering in CoMP scenarios, i.e., the BS clusters that jointly serve a UE change over time. As clusters can be formed individually for (groups of) UEs, the expected performance gain is higher than for static clusters.

Three major contributions have been presented in this chapter: (1) an evaluation of the reasonable cluster sizes from the wireless perspective, (2) an evaluation of the feasibility of the desired BS clusters under various realistic backhaul network scenarios,

and (3) a tool that addresses the discovered mismatch between the clusters *desired* from the wireless perspective and those that are *feasible* according to the backhaul conditions.

In the first contribution, I have evaluated how big reasonable clusters for JT CoMP have to be from the wireless point of view. This simulation has shown that the gain in UE spectral efficiency per additional BS decreases from nearly 20 % down to about 2 % when clusters consist of more than 7 BSs. This renders cluster sizes between 2 and 7 BSs as reasonable assumption.

Based on these results, I have evaluated the feasibility of such desired clusters for different optical backhaul network scenarios. Simulations have shown that for realistic future backhaul network deployments, the desired clusters are often infeasible due to lacking backhaul capacity and too high latency. On the other hand, in case a mobile network operator only wants small clusters (2-3 BSs) and provides a well-dimensioned backhaul network architecture, the feasible clusters from the backhaul perspective can be much larger than the desired ones. These differences between desired and feasible cluster sizes lead to significant overhead on the backhaul and wireless network, e.g., for gathering and processing CSI and for exchanging UE data between BSs that will not take part in the CoMP transmission/reception eventually.

This mismatch is addressed by a combined wireless/backhaul CoMP clustering architecture. It takes into account both the wireless channel conditions and the current backhaul status when deciding which BSs take part in CoMP. This approach reduces overhead by up to 85 % in terms of reduced CSI gathering as well as signaling and UE data exchange.

Finally, I have shown in a generic evaluation that the overall wireless network throughput is maximized if a limited pool of backhaul network resources is used to deploying few large clusters instead of many small ones. This gives a general guideline for deciding how to spend available backhaul resources. In a real deployment, decisions need to be made case by case using the proposed clustering techniques, due to the possibly diverse character of the backhaul links.

The insights gained in this chapter show that properties of the backhaul network play an important role when applying BS cooperation, especially JT, in future mobile access networks. Even future high-capacity optical backhaul networks cannot guarantee that cooperation techniques can be applied ubiquitously due to remaining capacity bottlenecks and latency shortcomings. Hence, mechanisms and algorithms for deciding where, when, and how to cooperate must not only rely on information from the wireless side but follow a cross-layer approach by also incorporating the backhaul network's status.

Chapter 5

Improving CoMP cluster feasibility

Contents

The evaluations in Chapter 4 have shown that capacity and latency requirements of CoMP, and especially JP, cannot be fulfilled even with future optical backhaul network technology without excessive over-provisioning. As over-provisioning is not a desired solution for network operators due to its high cost, Luca Scalia, Changsoon Choi, and I developed several approaches to improve CoMP cluster feasibility without the need for over-provisioning.

The developed approaches can be divided as follows: technology-independent techniques (Section 5.1), which are suitable for any backhaul network technology, and PON-specific techniques (Section 5.2), which can be applied to improve future PON systems to better support CoMP (for an introduction to PONs please refer to Appendix A). Furthermore, note that the generic techniques can be applied in PON scenarios, too, which is the reason why some evaluations will be made assuming a PON-based backhaul network. Their application, however, is not limited to PONs.

5.1 Technology-independent techniques

The technology-independent techniques can be applied to any kind of backhaul network to improve CoMP cluster feasibility. This is particularly useful to better support future

wireless technologies in scenarios where the backhaul network infrastructure has not been upgraded yet.

The first approach dynamically moves the CoMP controller function within the network to optimally exploit the available backhaul resources. It is presented in Section 5.1.1. The second approach exploits flexibilities of certain backhaul network technologies to adapt the backhaul network's properties to current wireless requirements. Details are given in Section 5.1.2.

5.1.1 CoMP controller reassignment

Luca Scalia and I propose a mechanism that addresses both the backhaul latency and capacity problem in a CoMP-enabled network. Instead of having a fixed location in the network for a cluster's CoMP controller, we move the controller function dynamically. Moving the controller can be achieved, e.g., by assigning a UE to a different BS in the cooperative cluster by triggering a hand-over. The BS to which the controller function is moved (i) provides a low latency to all BSs in the desired cluster, and (ii) has sufficient backhaul link capacities to accommodate the additional traffic load generated by JP. The major difference to traditional network-initiated hand-over procedures is that it is not only done based on wireless aspects. The novelty is to jointly optimize the wireless and wireline part of the network by triggering a hand-over.

Due to its generality, this approach is applicable to most wireless access systems, like LTE or LTE-A. Furthermore, it enables to balance load in backhaul architectures which cannot exploit inherent load balancing or statistical multiplexing (e.g., WDM PONs).

5.1.1.1 Idea

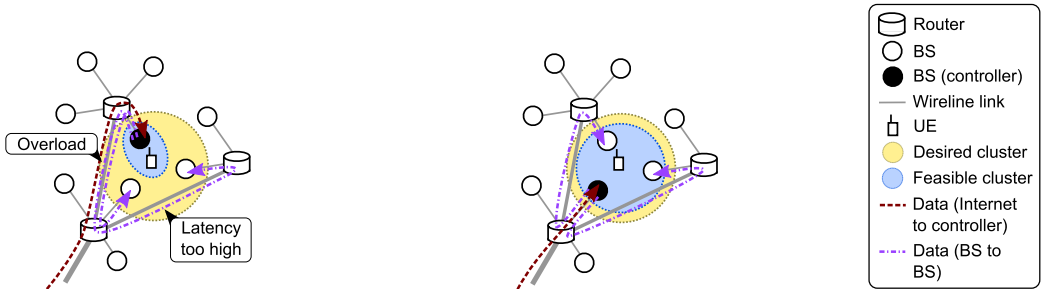
In most cellular wireless systems, each UE is connected to a single serving BS that provides the best channel quality/service among all the available BSs the UE can receive. In a semi-distributed JP-capable system, which turned out to be a beneficial implementation strategy for CoMP systems (Section 2.2.1), the serving BS usually not only manages the data streams coming from/directed to the UE, but also acts as a controller for the cooperative JP cluster. The controller distributes the UE data streams to all cooperating BSs and computes precoding information based on collected CSI. Furthermore, the concept of the serving BS becomes more loose. Basically, each cooperating BS can act as CoMP controller from the served UE's point of view since the wireless transmission is done simultaneously by all BSs and not only by the serving one.

Changing the CoMP controller location for an UE that is served cooperatively is not visible to the UE itself as the cooperating BS cluster does not change. Within the core and backhaul network, however, the load distribution can change significantly. This is due to UE and signaling data streams having to go via different paths.

In our mechanism, we exploit knowledge about the current core and backhaul network status, like link load and latency. Instead of fixing the CoMP controller location for a certain UE based on the wireless channel properties, as it is done for the serving BS today, we position the controller within a cooperative cluster such that the UE is provided the

best cluster, e.g., the one with the most feasible cooperating BSs. We will call this mechanism the *CoMP Controller Reassignment (CCR)* mechanism in the following. As we will confirm later, this CCR can significantly influence JP feasibility.

Figure 5.1 illustrates the CCR mechanism. When the controller function is located at the serving BS of the UE, just a very small JP cluster is feasible due to the backhaul network limitations. Moving the function to another BS allows more BSs to cooperate and hence to provide a better service quality. The figure only shows the data distribution phase; CSI signaling is omitted.



(a) Closest BS acts as controller, i.e., no CCR. The feasible cluster is small.

(b) Best suited BS acts as CoMP controller. The feasible cluster is large.

Figure 5.1: Benefits of CCR mechanism in an example scenario.

The detailed steps of the CCR procedure from detecting the necessity of CoMP until the activation of CoMP in the end are summarized in Figure 5.2.

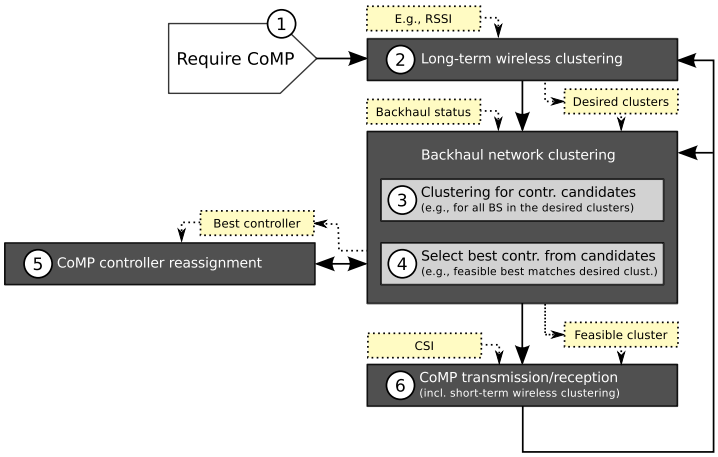


Figure 5.2: Illustration of CoMP Controller Reassignment (CCR) mechanism.

After the necessity for CoMP to serve a certain UE has been detected in Block (1), the *desired* BS cluster is determined in Block (2) based on wireless channel conditions. We propose to do this at the current serving BS of the UE as this BS knows about the long-term wireless channels' qualities between the UE and the neighboring BSs. This

information is collected anyway for hand-over decisions and hence can be used to limit the scope of the procedure's next step.

The clustering is done for all candidate CoMP controllers in Block (3). As the wireless transmission itself is not influenced by the CoMP controller location in the network, basically any BS is a candidate for hosting the controller function. To limit the scope of controller candidates, it makes sense to start with BSs that are members of the *desired* cluster. For each candidate controller, the *feasible* cluster is calculated based on the current backhaul and core network status using the mechanisms introduced in Section 4.3.2. The most beneficial BS for hosting the controller, i.e., the one whose corresponding *feasible* cluster best matches the *desired* cluster, is selected in Block (4). As soon as one candidate controller fully supports the desired cluster, Block (3) can terminate the search prematurely.

After finding the best CoMP controller location, Block (5) performs the actual CoMP controller reassignment. Depending on the capabilities of the deployed BSs, there are multiple possibilities to achieve this. In case the CoMP controller can be positioned arbitrarily, i.e., it is not bound to the serving BS of a UE, this feature can be used to move the controller to the determined BS. Otherwise, the conventional hand-over procedure can be exploited to change the serving BSs and hence move the CoMP controller and the UE data and signaling load within the backhaul network. This is possible if CoMP is applied in downlink *and* uplink direction. Otherwise, the candidate BSs in Block (3) need to be limited to those that are able to still provide the required service quality in the direction that does not use CoMP.

Finally, the cooperative transmission happens in Block (6) based on the configuration determined in the previous steps.

5.1.1.2 System model and evaluation procedure

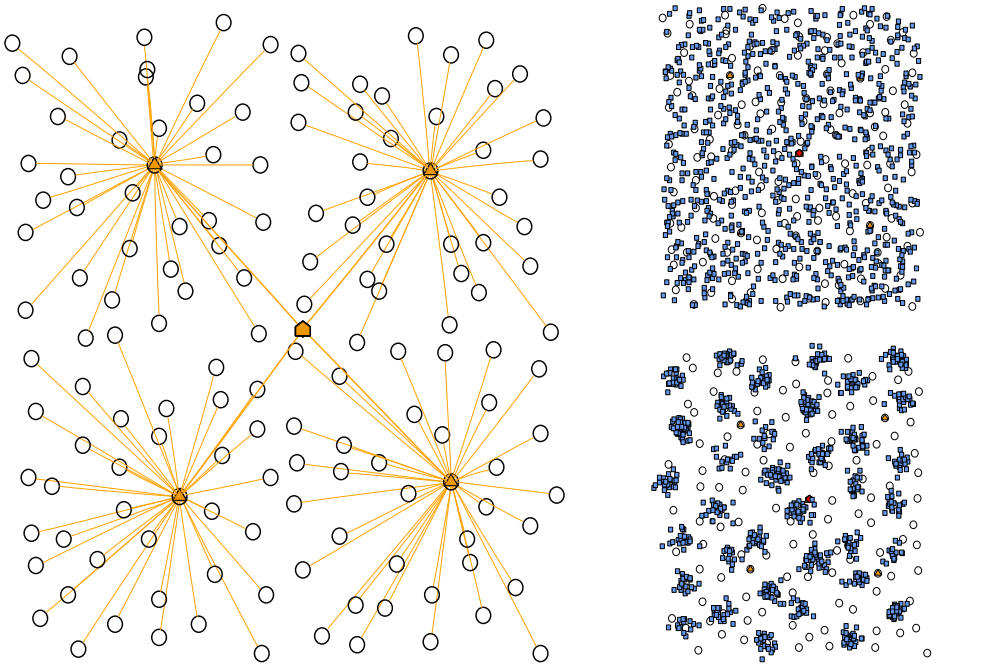
This section provides system settings, assumptions, and procedures for the simulative evaluation of the CCR mechanism. The goal is to evaluate the capability to improve CoMP cluster feasibility by CCR without upgrading backhaul network hardware, like increasing available capacity.

Base station distribution and backhaul topology We distribute 144 BSs in a square field of 12x12 BSs. The edge lengths of this grid, i.e., the inter-BS distances, are $\overline{s} = 500\,\mathrm{m}$, which corresponds to an urban scenario setting. In every simulation run, the position of each BS is shifted horizontally and vertically by two normally-distributed random variables with standard deviation $\overline{s}/5$ and zero mean.

Once deployed on the grid, the BSs are connected through a backhaul network. We consider two different PON technologies for the backhaul network implementation: Time Division Multiplexing (TDM) and Wavelength Division Multiplexing (WDM) PONs, which both define a backhaul tree topology with different link properties.

Depending on the PON splitting factor $\mathrm{sf}_{\mathrm{PON}}$, the field of BSs is divided into multiple PONs such that each BS is assigned to exactly one PON; each BS is colocated with an Optical Network Unit (ONU). A value of $\mathrm{sf}_{\mathrm{PON}} = 36$, which leads to 4 PONs to cover

the BS field, is typical for a WDM PON setup. For the TDM PON setup, a smaller splitting factor is required to provide enough capacity to each of the BSs. We choose a splitting factor of $\mathrm{sf}_{\mathrm{PON}} = 9$, which results in 16 PONs. A snapshot of a possible BS arrangement and correspondent topology is shown in Figure 5.3a.



(a) WDM PON backhaul topology with $\mathrm{sf}_{\mathrm{PON}} = 36$, consisting of 144 BSs.

(b) Uniform and crowded UE distribution for $n_{\mathrm{UE,cell}} = 2$; BSs have 3 sectors/cells.

Figure 5.3: Examples for a PON backhaul topology and UE distribution types. Circles are BSs with ONUs; triangles represent AWGs. A house symbol marks the colocated OLTs. Lines represent fiber connections. UEs are drawn as rectangles.

PON configuration In WDM PONs, each ONU is connected to the Optical Line Terminal (OLT) via two dedicated wavelengths (one for uplink, one for downlink). We assume that the wavelength capacity is 2.5 GBit/s both in the uplink and in the downlink [44]. This capacity is shared by the three sectors of each BS, which results in roughly 830 MBit/s per sector. This high data rate makes the WDM PON suitable for implementing the backhaul network for LTE-Advanced (1 GBit/s wireless downlink and 500 MBit/s wireless uplink per cell [68]). The downside of the fixed capacity allocation in WDM PONs is that capacity cannot be shifted to ONUs where it is currently required. Hence, there is no multiplexing gain.

In TDM PONs, the overall fiber capacity, usually one single wavelength, is shared among all ONUs. Typical shared capacities are 10 GBit/s in the downlink and 2.5 GBit/s in the uplink [44]. With these settings and a splitting factor of 9, each BS gets 1.1 GBit/s

in the downlink and 280 MBit/s in the uplink direction on average (respectively 370 MBit/s and 93 MBit/s per sector). These data rates make TDM PONs a candidate backhaul technology for LTE (325 MBit/s wireless downlink and 85 MBit/s wireless uplink per cell [69]), but complicates its use for LTE-Advanced. Due to the flexible capacity allocation, TDM PONs can easily adapt to load inequalities among ONUs. The optical link latency is determined by the light propagation delay in the fiber.

The OLTs of the PONs are all colocated at the center of the BS field ("house" symbol in Figure 5.3a on the previous page). All these OLTs are connected to a single IP router, which is also colocated with them, to enable inter-PON data exchange for JP.

Besides the IP processing at the OLT router, IP processing also occurs at the ONUs/ BSs. We assume an IP processing delay of 0.1 ms [63].

User placement and traffic generation The traffic load is generated by UEs placed in the BS field. Each UE is assigned a certain uplink and downlink data rate.

For the UE positioning we have considered two different approaches. In the first one, all UEs are distributed uniformly in the field. We evaluate our CCR mechanism in this rather unrealistic scenario as this is a very bad situation for the mechanism because there is not much spare capacity at neighboring BSs. In the second scenario, a subset of the BSs, called *hubs*, is selected around which UEs are concentrated. This is done for each UE by uniformly selecting one of the hubs and choosing the UE's position from two normally-distributed random variables with the hub's coordinates as mean. The standard deviation is set to 250 m. Changing this parameter influences the gains of the CCR mechanism. Larger values cause a more equal UE distribution, similar to the first scenario, resulting in smaller gains. Smaller values cause the UEs to be concentrated at hubs even more. Hence, shifting load using the CCR technique leads to even higher gains. Examples for both scenarios are given in Figure 5.3b on the preceding page.

The average number of active users is varied from 2 to 20 UEs per sector, i.e., 6 to 60 UEs per BS, assuming 3 sectors per BS. This covers the full spectrum from low load up to dense urban scenarios.

The UE data rate r_{UE} is varied from 10 MBit/s to 100 MBit/s. This data rate is split into an uplink and downlink component. Recent measurements of smartphone traffic have shown that the ratio between uplink and downlink traffic varies between 1:1 and 1:10, with an average of 1:6 [70]. The formerly different traffic requirements in the uplink and downlink direction diminish due to more content sharing by mobile users, e.g., via social applications. As this trend is expected to continue in the future, we assume a ratio between uplink and downlink of 1:3, i.e., $r_{\mathrm{UE}} = 100$ MBit/s results in 25 MBit/s upstream and 75 MBit/s downstream data rate.

For each generated UE we also decide whether it needs JP or not. The probability that JP is required is set to 50% to reflect the high level of interference, which is expected in future cellular networks due to the higher cell density.

Base station cooperation and wireless clustering We want to find out the influence of the CCR mechanism on the feasibility of JP. For this, we first need to select the

BS clusters in which cooperation is *desirable* from the wireless point of view. We reuse the results of Section 4.1 and use small, medium, and large desired wireless clusters consisting of 3, 7, and 16 BSs.

We focus on the downlink cooperation in our simulations. As we assume a semi-distributed implementation of BS cooperation within the clusters [3], the BS hosting the CoMP controller function needs to forward the data for a UE that needs JP to all BSs in the cluster of this UE.

Wireless cluster feasibility evaluation We first check the wireless cluster feasibility with the closest (serving) BS S acting as controller for each UE that needs JP. This is the reference feasibility without any CCR mechanism. Thereafter, this step is repeated for the $n_{\text{candidate}}$ closest BSs to the UE under consideration to simulate the CCR mechanism. This way, we get a feasible cluster for each candidate controller BS, which is always smaller than or equal to the desired wireless cluster. The controller whose feasible cluster covers most of the desired cluster is selected as active controller.

In this simulation, we look at the overall system with all UEs being active in parallel. Hence, the order in which UEs are checked influences the feasibility of the individual clusters (UEs that are checked first have more remaining backhaul capacity than UEs that are considered later). Therefore, the order in which UE clusters are tested for feasibility is randomized in each iteration of the simulation.

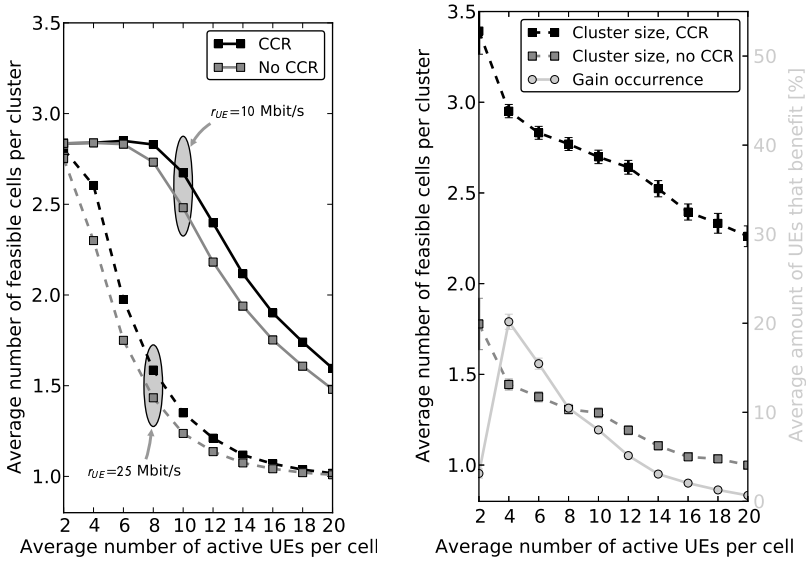
Checking the feasibility of the desired cluster for a certain controller candidate BS C is done by checking the backhaul network's remaining capacity and latency between C and all BSs in the desired cluster. This procedure has been described in Section 4.3, in particular in Algorithm 4.2 on page 59. If the remaining link capacities are high enough to transport the UE downlink data and the latency is low enough to permit timely CSI and data exchange, the BS can participate in the cooperation.

5.1.1.3 Simulation Results

In the simulations, I evaluated two different PON technologies: TDM PONs and WDM PONs. All following plots contain confidence intervals for a confidence level of 95 %.

TDM PON I have first evaluated the cluster feasibility in a TDM PON backhaul network deployment assuming a uniform UE distribution in the network. This means that also the load is equally distributed among the BSs in the scenario. The feasible cluster sizes with and without the CCR mechanism are compared in Figure 5.4 on the following page.

Figure 5.4a on the next page shows the number of cells per cluster, averaged over all UEs in the system that require JP. I show the influence of the average overall number of active UEs per cell (this includes those UEs that require JP and those that do not) and the UEs' data rate r_{UE}. While at low load the desired cluster size (≈ 3.4) can be reached, the reachable size decreases when the load increases. The plot shows a moderate gain when using the CCR mechanism; the improved cluster size is $\approx 10\,\%$ higher. This effect occurs although the load is distributed uniformly in the network. Gains only occur in the



(a) All UEs' average

(b) Benefiting UE average; $r_{\mathrm{UE}} = 25\,\mathrm{MBit/s}$.

Figure 5.4: Feasible cluster sizes with and without CCR in TDM PON scenario assuming uniform load distribution and small desired wireless clusters (3 BSs).

phase when the network is just getting saturated, e.g., in the range from 4 to 14 UEs per cell for $r_{\mathrm{UE}} = 25\,\mathrm{MBit/s}$. Here, the backhaul links of some BSs are overloaded already while some neighbor BSs still have free capacity. Furthermore, as UEs are distributed randomly, they are not fully uniformly distributed in each simulation run. Hence, the slight load variance increases the benefit of the CCR mechanism.

The asymmetry of both the link capacities in the TDM PON and the UE data rates (the downlink rate is higher than the uplink rate) also increases the benefit of CCR. The reason is that a CoMP controller has to forward the high-rate downlink data stream of a UE requiring JP via its low-rate uplink backhaul link to other BSs in the cluster.

In Figure 5.4b, only those UEs are taken into account that actually benefit from the CCR mechanism. The plot shows that up to 25 % of all UEs benefit from CCR and that CCR increases their feasible cluster size by 100 % to 150 %. This shows that even in the difficult TDM PON scenario, CCR is able to provide big gains to a few UEs.

The advantage of CCR further increases when the wireless side desires larger clusters. Figures 5.5 and 5.6 show the influence of CCR for medium-sized and large desired wireless clusters containing 7 BSs and 16 BSs, respectively. The UE data rate r_{UE} has been set to 25 MBit/s.

The plot for medium desired clusters (Figure 5.5 on the next page) shows that now up to 50 % of all UEs benefit from CCR. Therefore, the average gain for all UEs also raises up to $\approx 50\,\%$, while the gain for those UEs that benefit is between 100 % and 200 %. For large desired clusters (Figure 5.6 on the facing page), the qualitative behavior is identical. The gains of CCR, however, are even higher. This behavior comes from the fact that

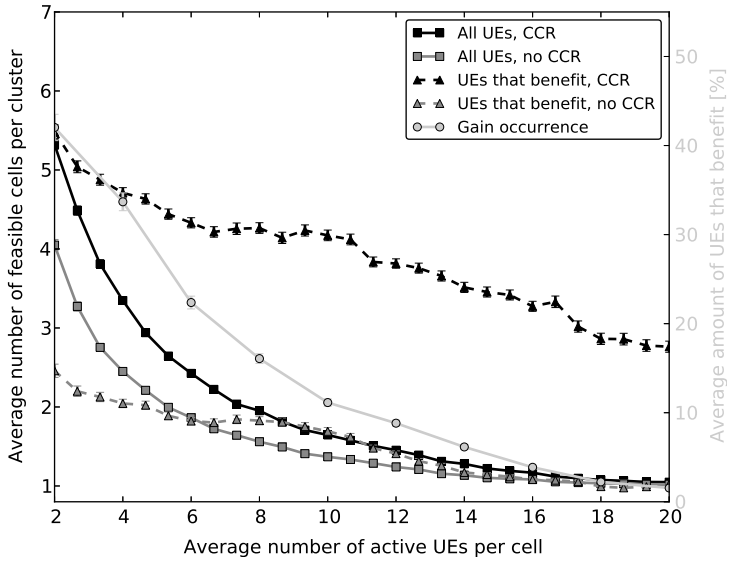


Figure 5.5: Feasible cluster sizes in TDM PON assuming uniform load distribution and medium desired wireless clusters of 7 BSs; $r_{\mathrm{UE}} = 25\,\mathrm{MBit/s}$.

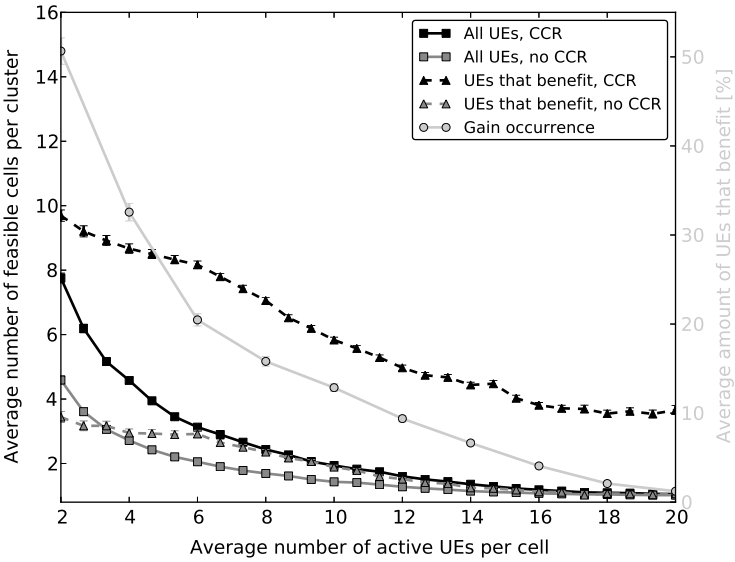
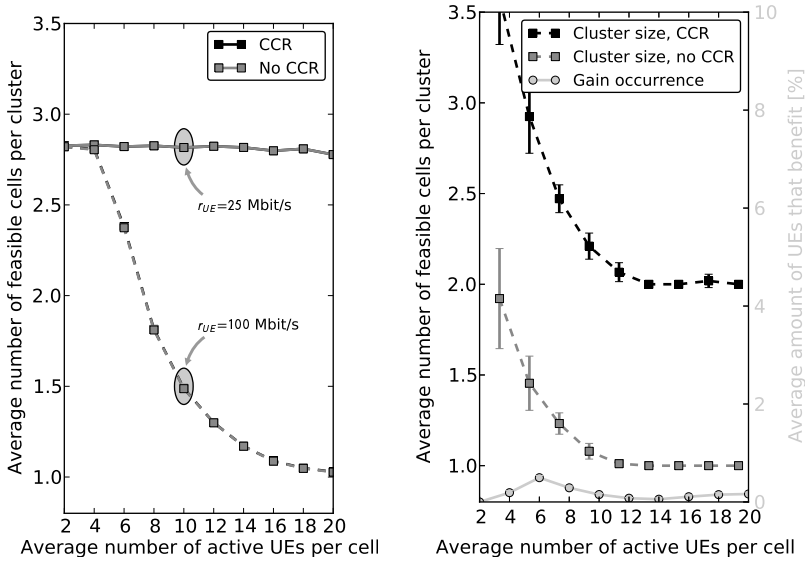


Figure 5.6: Feasible cluster sizes in TDM PON assuming uniform load distribution and medium desired wireless clusters of 16 BSs; $r_{\mathrm{UE}} = 25\,\mathrm{MBit/s}$.

the larger clusters have higher requirements towards the backhaul network. Hence, the probability to fail at the closest BS, and to improve the situation with another serving BS, is higher.

WDM PON I also evaluated cluster feasibility in a WDM PON backhaul network deployment, which is more suitable for an LTE-Advanced system due to the higher

capacity it provides. Therefore, I increased the UE data rate to 100 MBit/s. The resulting feasible cluster sizes for a uniform UE distribution with and without the CCR mechanism are compared in Figure 5.7.



(a) All UEs average (b) Benefiting UEs average

Figure 5.7: Feasible cluster sizes in WDM PON assuming uniform load distribution and small desired wireless clusters (3 BSs).

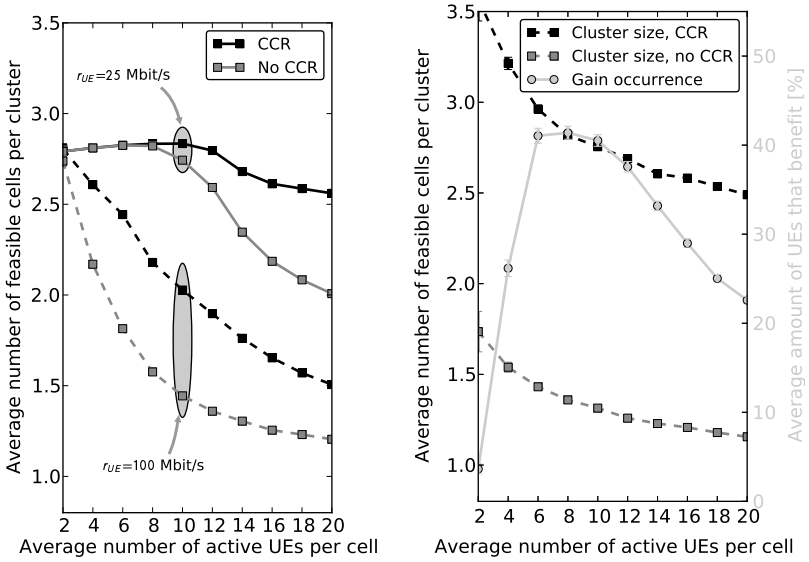
The plots reveal that there is nearly no gain when applying the CCR mechanism in the WDM PON assuming a uniform load distribution in the network. The reason is that compared to TDM PON, where uplink and downlink capacities are asymmetric, the bottleneck in the uplink is mitigated.

One big advantage of the TDM PON is that, due to the shared medium, unused capacity can be dynamically shifted to ONUs that require it. This is usually done at the OLT by running a Dynamic Bandwidth Allocation (DBA) algorithm. This way, TDM PONs can inherently cope with non-equal load distributions. WDM PONs do not have this feature. Each ONU has a fixed capacity that cannot be shared with other ONUs. Hence, an unequal load distribution in the network is problematic for a WDM-PON-based backhaul network as soon as the inequalities cause overload at one ONU.

The CCR mechanism provides a solution to this problem. Instead of shifting capacity from one ONU to another, as in the TDM PON, now load is shifted from one ONU to another, less loaded one. The effect is similar. Furthermore, as just the CoMP controller is moved within the network, no drawbacks on the wireless side are introduced as the wireless transmission is not touched, i.e., the desired wireless clusters do not change.

Figure 5.8 on the facing page shows the benefits of CCR when the load in the system is not distributed equally but according to the scheme described in Section 5.1.1.2.

The plot in Figure 5.8a on the next page shows that now the average feasible cluster size even of all UEs is improved by up to $\approx 50\,\%$, e.g., for 10 UEs per cell and $r_{\mathrm{UE}} =$



(a) All UEs average (b) Benefiting UEs average

Figure 5.8: Feasible cluster sizes in WDM PON assuming non-uniform load distribution and small desired wireless clusters (3 BSs).

100 MBit/s. For those UEs that benefit (up to 45 %), the gain is between 100 % and 120 % (Figure 5.8b).

These results show that the CCR mechanism is able to solve the unequal load distribution problem in fixed-capacity backhaul technologies like WDM PONs. Note that non-uniform UE distribution is just one source for unequal load among different BSs. Another example are the diverse traffic patterns that UEs can have.

Figure 5.9 on the next page shows the equivalent evaluation in the WDM PON scenario that has been shown for the TDM PON in Figure 5.5 on page 81: the desired wireless cluster size is increased to 7 BSs), the UE data rate r_{UE} is 100 MBit/s, and the UEs are distributed uniformly.

The observed behavior is similar to the TDM PON. The overall gains, however, are clearly visible throughout the whole load range. This difference comes from the removed bottleneck in the PON's uplink capacity.

5.1.2 Network reconfiguration

The CoMP Controller Reassignment (CCR) approach (Section 5.1.1) changes the UE handling within the cellular network to improve CoMP cluster feasibility. The backhaul network itself remains untouched. This is an advantage, as no interaction with the backhaul network's control plane is required, but also does not fully exploit all network capabilities. This drawback is addressed in this section. Martin Dräxler, Luca Scalia, and I developed an approach that better exploits the backhaul network infrastructure by adapting it to current requirements resulting from wireless channel conditions.

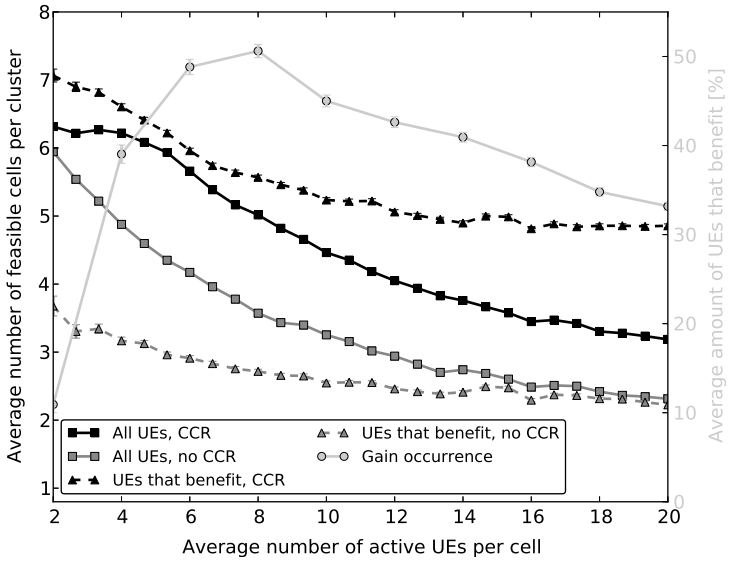


Figure 5.9: Feasible cluster sizes in WDM PON assuming non-uniform load distribution and medium desired wireless clusters (7 BSs); $r_{\mathrm{UE}} = 100\,\mathrm{MBit/s}$.

The proposed mechanism exploits two things. First, the information about *why* a desired cluster could not be established, e.g., due to missing capacity on a certain backhaul link. Such information is gained by slightly extending the backhaul network pre-clustering algorithm presented in Section 4.3.2. Second, the mechanism exploits flexibilities of backhaul networks, for example, adding or shifting capacity to links where it is needed. Example technologies that are likely to be used in future backhaul networks are given in the following:

- Passive Optical Networks (PONs) provide high capacity at low cost and are therefore well suited to connect BSs to the metro or core network. Depending on the actual PON type, capacity can be dynamically added or shifted within the network by modifying the DBA mechanism in case of a TDM PON or by activating additional wavelengths in case of a WDM PON [44].

- Microwave (MW) links are used to create a mesh interconnection between neighbor BSs. They can be activated on demand to deal with load peaks or to dynamically support CoMP. Furthermore, rate adaption can be exploited to change the available capacity on the links [71].

- Similar to MW links, Free-Space Optics (FSO) can be used to interconnect neighbor BSs. They provide even higher capacity and can be activated on demand in addition to low-capacity wireline backhauling links [72].

- In networks that provide virtualization, additional capacity can be dynamically added to slices that require it [73].

Such flexibilities of the backhaul network technologies can be exploited to adapt the backhaul network according to the wireless requirements. We call this adaptation *backhaul network reconfiguration*. For example, if a desired wireless cluster is not feasible due to missing capacity in a WDM-PON-based backhaul network, this lack of capacity can be eliminated by activating an additional wavelength on the link that does not have enough capacity. This way, instead of accepting and dealing with backhaul network limitations, we are now able to actually improve CoMP performance.

The approach presented in the following is very generic, i.e., it is not limited to a specific wireless network system, like LTE or WiMAX, and is also independent of the actual backhaul network technology. Hence the ideas can be applied in a variety of scenarios where CoMP techniques shall be used.

5.1.2.1 Detecting reconfiguration opportunities

The basic requirement to enable backhaul network reconfiguration is to detect bottlenecks that prevent CoMP for desired wireless clusters. Based on the backhaul network pre-clustering algorithm, which has been presented in Section 4.3.2, such bottlenecks can be detected on the fly during the pre-clustering step. To achieve this, I extended Algorithm 4.3 on page 61 to not only report if a feasible cluster *can* be extended but also return the *reasons* if it is not possible. Hence, the output contains a list of bottlenecks in the backhaul network that are preventing a cluster extension. The resulting function clusterExtensionPossibleReport is shown in Algorithm 5.1 on the following page.

Compared to the original version of the algorithm, the extended one adds features to remember and return why adding an additional BS v to the cluster is not possible. For this, two additional set variables are added that store links that do not have enough capacity (bottleneck_capacity) and (sub-)paths from the new BS v to the controller s where the latency requirements cannot be fulfilled (bottleneck_latency). Whenever one of these constraints is violated, a tuple containing the location (link or path) and the amount of missing resources (additionally required capacity or delay reduction) is added to the corresponding set variable. Finally, these sets are returned.

5.1.2.2 System architecture

Based on the detected bottlenecks, the required changes need to be applied to the actual backhaul network. This requires a close interaction between the CoMP procedure, especially the clustering, and the backhaul network's control plane. An overview of the system component interaction is shown in Figure 5.10 on page 87.

After detecting that CoMP is required in (1), the desired wireless clusters are determined. This is done in (2) and requires information about the wireless channel conditions, like CQI or RSSI measurements. This mechanism of wireless long-term pre-clustering has been described in Section 4.3.1. The resulting desired cluster is used together with the backhaul network's current status in the backhaul network pre-clustering step to check the feasibility of the desired cluster. This happens in (3) and is done by the heuristic presented in Section 4.3.2 using the extended function clusterExtension-

Algorithm 5.1 clusterExtensionPossibleReport($G_{\text{clust}}, u, v, s$)

```
1: predecessor[v] ← u // set predecessor for new node
2: v_orig ← v // remember new node
3: t_totalRT ← 0 // initialize accumulated round-trip latency
4: req_cap_down ← b_cap_down[v] // get required downlink (s to v) capacity
5: req_cap_up ← b_cap_up[v] // get required uplink (v to s) capacity
6: constraints_fulfilled ← true // initialize return value
7: bottleneck_capacity ← ∅ // initialize additionally required link capacities
8: bottleneck_latency ← ∅ // initialize required latency reductions
9: while v ≠ s do // follow path from v up to s
10:   t_totalRT ← t_totalRT + l_t[u, v] + l_t[v, u] // increase accumulated round-trip delay
11:   if t_totalRT > t_max then
12:     // accumulated round-trip delay exceeds allowed threshold
13:     constraints_fulfilled ← false
14:     missing_latency ← t_totalRT − t_max // calculate missing link capacity
15:     ADD(bottleneck_latency, ((u, v_orig), missing_latency)) // store required latency reduction
16:   end if
17:   if req_cap_down > l_cap_free[u, v] then
18:     // downlink capacity is exceeded
19:     constraints_fulfilled ← false
20:     missing_capacity ← req_cap_down − l_cap_free[u, v] // calculate missing link capacity
21:     ADD(bottleneck_capacity, ((u, v), missing_capacity)) // store required link capacity
22:   end if
23:   if req_cap_up > l_cap_free[v, u] then
24:     // uplink capacity is exceeded
25:     constraints_fulfilled ← false
26:     missing_capacity ← req_cap_up − l_cap_free[v, u] // calculate missing link capacity
27:     ADD(bottleneck_capacity, ((v, u), missing_capacity)) // store required link capacity
28:   end if
29:   // step up one link towards s in the tree
30:   v ← u
31:   u ← predecessor[u]
32: end while
33: return (constraints_fulfilled, bottleneck_capacity, bottleneck_latency)
34: // return if feasible cluster can be extended and reasons in case it is not possible
```

PossibleReport from Algorithm 5.1 instead of the function clusterExtensionPossible. Now a threshold is used to decide whether the feasible cluster is enough (then the CoMP transmission/reception is conducted immediately in (5) within the found feasible cluster) or whether the feasible cluster is too small. In this case, the backhaul network reconfiguration is triggered to improve the feasible cluster. How to choose the order in which multiple desired clusters are selected for implementation depends on the strategy of the network operator. It is similar to the discussion in Section 4.4.

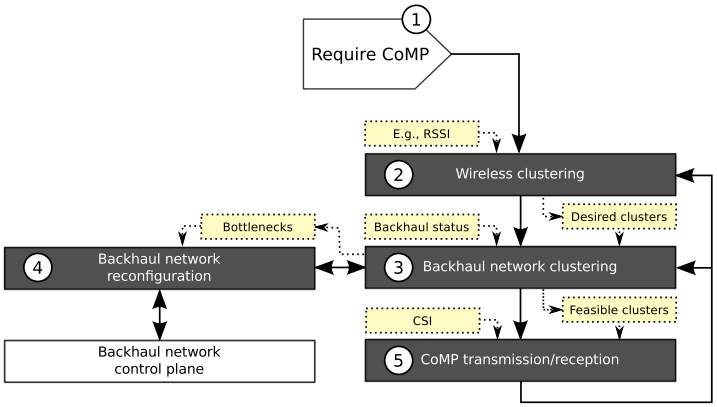


Figure 5.10: Overall clustering/reconfiguration system architecture

Block (4) needs to decide which actions to take in the backhaul network to improve cluster feasibility. This decision is made based on two kinds of information. First, the output of the backhaul network clustering heuristic in Block (3), which contains the reasons why certain BSs cannot be in the feasible cluster. And second, information about the backhaul network itself, e.g., which parameters can be changed for which links. This information is provided by the network's control plane. Now, these two things are matched by checking for each of the shortcomings whether one of the available backhaul parameters is able to combat the problem. If this is the case, the backhaul network is reconfigured via its control plane to eliminate the bottlenecks that prevent desired wireless clusters. This mechanism can also be used to reduce the backhaul network's capabilities again when they are not needed, e.g., to save energy.

After a reconfiguration step, the backhaul network clustering step in (3) has to be executed again to check whether feasibility has improved. If necessary, this loop can be executed multiple times until the desired wireless cluster becomes feasible or a termination condition, like a maximum number of performed reconfigurations steps, is reached. Furthermore, it is possible to detect that no further reconfiguration is possible anymore. This information is available directly in Block (4) when interacting with the backhaul network control plane.

This overall procedure has to be repeated whenever the desired wireless cluster, i.e., the long-term wireless channel conditions, or the cluster feasibility, i.e., the backhaul load and latency, change more than a defined threshold. The repetition is indicated by the arrows on the right side of Figure 5.10.

5.1.2.3 Evaluation

To get an idea of the impact of applying backhaul network reconfiguration, we have conducted several simulations. The simulation model is similar to the one used in the previous cluster feasibility evaluations in Section 4.2. 36 BSs are distributed in an urban scenario with a mean inter-BS distance of $\bar{s} = 500\,\mathrm{m}$. For each BSs, we vary its load

demand in Gbit/s via a parameter d. For the backhaul network, we evaluate two different types: a mesh network with $f_{\text{dens}} = 1.0$, which results in a network with medium link density, and a tree network. For the tree network, we assume a WDM PON deployment with a splitting factor of 9, i.e., each root of the tree (OLT of the PON) is connected to 9 BS. All OLTs are co-located at a central site and are directly interconnected.

For the desired cluster size, we reuse the same values as in previous evaluations: small clusters consisting of 3 BSs, medium-sized clusters consisting of 7 BSs, and large clusters with 16 BSs.

We simulate backhaul network reconfiguration in the mesh topology. For this, we define a fraction that determines which part of the reported backhaul network shortcomings are eliminated eventually by upgrading the backhaul network. This means that the bottlenecks reported by the backhaul network pre-clustering step (Block (3) in Figure 5.10 on the preceding page) are not all eliminated by Block (4) in Figure 5.10 on the previous page. This limitation reflects reality, where not all link properties can be changed arbitrarily as desired.

Figure 5.11 shows the results for cluster feasibility, which is the same metric that we used earlier in Section 4.2. It is the fraction of the number of BSs that actually *can* participate in the cooperation compared to the number of BSs that are contained in the *desired* wireless cluster, averaged over all simulation runs. BSs have 3 sectors and each of these sectors contains 8, 16, or 24 active UEs each requesting a data rate of 50 Mbit/s, which covers the spectrum from a rural to dense urban environment.

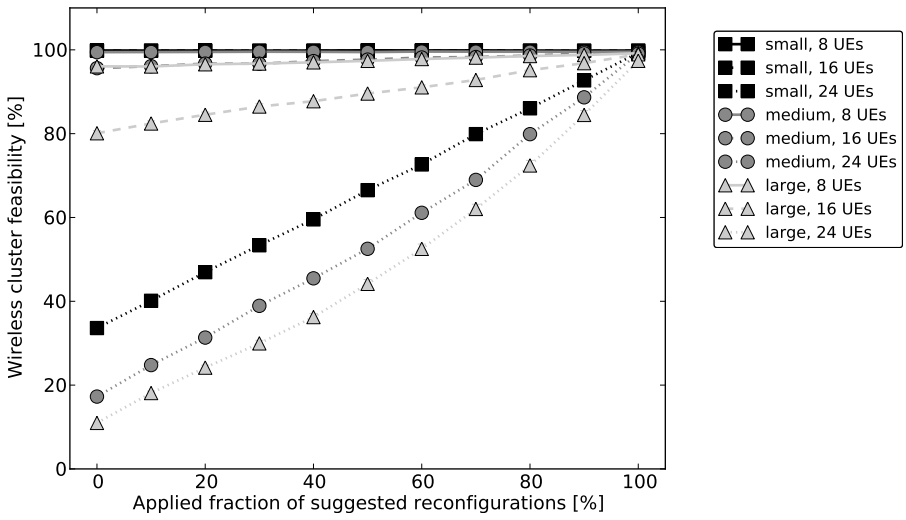


Figure 5.11: Reconfiguration influences cluster feasibility in a mesh backhaul network

The plot shows that increasing the fraction of implemented reconfiguration recommendations nearly linearly increases the wireless cluster feasibility up to 100 % when applying all requested changes to the backhaul network. This shows that exploiting the knowledge gained during the backhaul network clustering step clearly improves CoMP feasibility by adapting the backhaul network to the wireless requirements where possible.

Compared to mesh networks, which provide point-to-point links between BSs, in WDM PONs, fine-grained reconfiguration changes are not possible. The only degree of freedom is the assignment of additional wavelengths to individual ONUs, which increases the capacity between the BS connected to the ONU and the OLT.

Figure 5.12 shows the results for a simulation assuming a backhaul network based on WDM PONs. All BSs are connected via four WDM PONs such that each BS is connected to the OLT via two optical wavelengths with 2.5 Gbit/s capacity each. For the reconfiguration, each BS can be assigned two additional wavelengths on demand. The parameters are r_{UE} for the traffic demand for each active UE in Mbit/s, the desired wireless clusters size, and the number of active UEs per BS.

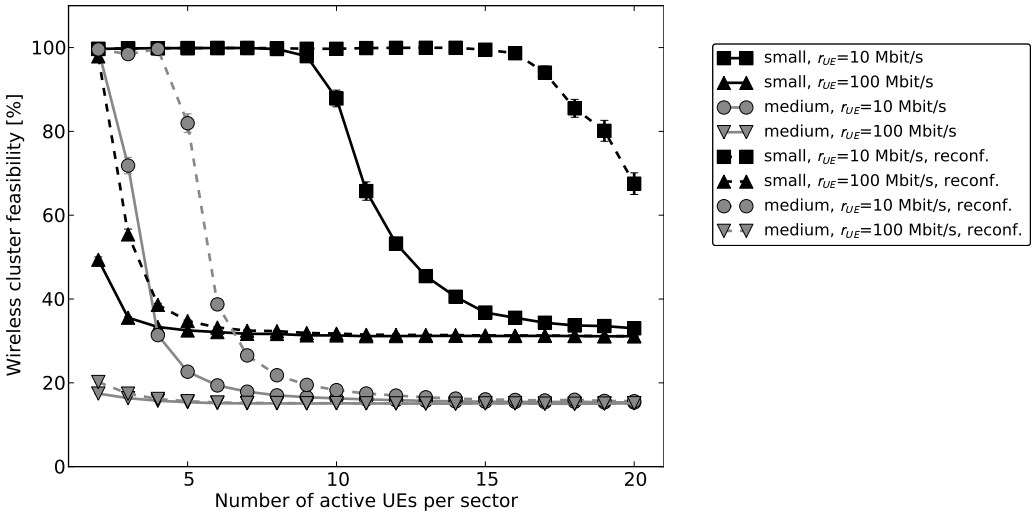


Figure 5.12: Reconfiguration influences cluster feasibility in a tree backhaul network

The solid lines show the feasibility before any reconfiguration steps, using the same metric as before. The dashed lines indicate how the feasibility improves when using the reconfiguration mechanism, which activates additional wavelengths for overloaded BSs whenever required. The results indicate that backhaul network reconfiguration significantly increases the wireless cluster feasibility, especially in high load scenarios.

5.2 PON-specific techniques

The technology-independent techniques, presented in Section 5.1, are able to improve CoMP cluster feasibility in arbitrary scenarios. This advantage is a drawback at the same time as such generic approaches are not able to exploit certain technology-specific properties. Exploiting such properties can lead to even further improvements.

In this section, I focus on Passive Optical Networks (PONs). PONs are optical networks that provide high capacity at reasonable invest and have only low maintenance

costs as equipment deployed in the field is passive. For an overview of different types of PONs and their properties please refer to Appendix A.

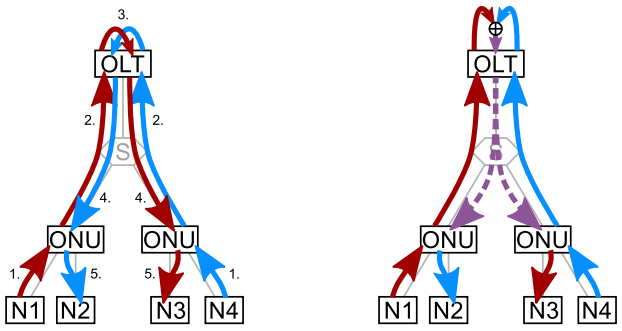
I first present an approach for reducing load by applying network coding techniques in Section 5.2.1. Thereafter, Section 5.2.2 introduces an extension for PONs that permits to directly exchange data between nodes in the same PON to reduce latency and to provide additional capacity for PON-internal data exchange.

5.2.1 Network coding

In this section, I propose an extension to the standard Ethernet PON (EPON) architecture (Section A.1), which has been developed together with Konstantin Miller and Hagen Woesner. This extension uses linear Network Coding (NC) [74] to increase downlink throughput by up to 50 % without changing PON hardware. The approach leverages PON-internal traffic, i.e., traffic whose source and destination are both located within the same PON. Details on the extension are provided in Section 5.2.1.1. Deployment scenarios and applications where a significant amount of PON-internal traffic is present, like audio/video conferencing, Peer-to-Peer (P2P) file dissemination, and, the most important one for this thesis, wireless access networks with cooperating BSs are discussed in Section 5.2.1.2. Section 5.2.1.3 evaluates the benefits of NC in EPONs.

5.2.1.1 Network Coding extension for Ethernet Passive Optical Networks

The conventional as well as the new NC-enhanced data exchange within a PON is illustrated in Figure 5.13.



(a) No NC active. (b) NC is activated.

Figure 5.13: PON-internal data flow with and without NC. N1 to N4 are nodes behind an ONU, e.g., BSs. The small numbers indicate the order of the various transmissions.

Concept Assume any node connected to ONU_1 wants to send a packet p_1 to a node connected to ONU_2. At the same time, a packet p_2 is to be sent in the opposite direction (from a node connected to ONU_2 to a node connected to ONU_1), not necessarily between

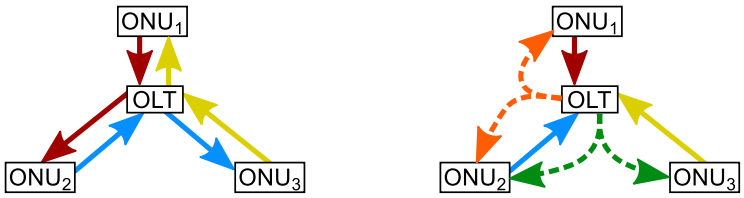
the same two nodes. Without NC, two packet transmissions are required in the PON upstream and two in the downstream (Figure 5.13a on the preceding page).

This symmetric traffic pattern occurs inherently when using JP CoMP techniques. UE data and signaling have to be exchanged between cooperating BSs when choosing the distributed and semi-centralized implementation strategy (Section 2.2.1).

We modify the OLT behavior such that instead of forwarding p_1 to ONU_2 immediately, it waits until the reception of p_2. If p_2 arrives with a maximum delay of $d_{\max}$, both packets are linearly combined to a new packet $p := p_1 \oplus p_2$, where $\oplus$ is the bit-wise exclusive or (XOR) operation. The new packet p is then sent to the multicast group consisting of ONU_1 and ONU_2. Upon reception, ONU_1 decodes p_2 by applying the XOR operation once again: $p_2 = p \oplus p_1$; ONU_2 decodes by $p_1 = p \oplus p_2$. If p_2 does not arrive at the OLT within $d_{\max}$, p_1 is forwarded downstream uncoded. The resulting data flow with activated NC is illustrated in Figure 5.13b on the facing page.

Since multicast in TDM PONs exploits the broadcast nature of the underlying medium, only one time slot is required per downstream multicast operation. Hence, downstream throughput increases by 50 % by using the time slots freed by NC. A similar approach has been studied in the context of wireless networks [75].

The presented mechanism can be extended to scenarios with no direct communication between any ONU pair. Instead, any cyclic traffic is sufficient as indicated in Figure 5.14.



(a) No NC active. (b) NC is activated.

Figure 5.14: Cyclic data flow among three ONUs with and without NC. The splitter node is omitted for clarity.

Assume that, e.g., ONU_1 sends a packet p_1 to ONU_2, which sends a packet p_2 to ONU_3. ONU_3, in turn, sends a packet p_3 to ONU_1 again. These 3 transmissions can occur in an arbitrary order. In such a case, NC can be applied in the following way. The OLT multicasts packet $p' := p_1 \oplus p_3$ to ONU_1 and ONU_2, and packet $p'' := p_2 \oplus p_3$ to ONU_2 and ONU_3. Upon reception, ONU_1 decodes by applying $p_3 = p' \oplus p_1$, ONU_2 decodes by $p_1 = p' \oplus p'' \oplus p_2$, and ONU_3 decodes by $p_2 = p'' \oplus p_3$. Instead of 3 downstream transmissions, the OLT only requires 2 transmissions, achieving still a gain of 33 %. In general, the NC gain for any cyclic traffic is $\frac{1}{n}$, where n is the number of nodes forming the cycle. The aforementioned bidirectional traffic is then a special case of cyclic traffic with $n = 2$.

Implementation at the OLT We propose the following scheme to apply NC to traffic cycles with lengths up to $c_{\max}$. The scheme runs at the OLT and requires a buffer

matrix $\{b_{ij}\}$. Each buffer b_{ij} stores packets from ONU_i to ONU_j together with their arrival times.

1. Whenever the OLT receives a packet p_i from ONU_i to ONU_j, it constructs a directed graph $G = (V, E)$, where each vertex $v \in V$ corresponds to an ONU and edges ij correspond to non-empty buffers b_{ij}. This graph is analyzed to find cycles of maximum length $c_{\max}$ that contain edge ij. This can be done very efficiently, e.g., by using Dijkstra's shortest path algorithm [76].

   a) If no cycle is found, the received packet is stored in buffer b_{ij} together with its arrival time. If the buffer was empty, a timer with the packet's maximum buffering time $d_{\max}$ is started.

   b) If such a cycle is found, the OLT performs the corresponding coding operation(s) and transmits the coded packet(s). For each packet removed from a buffer, the corresponding timer has to be restarted with a duration equal to the remaining maximum buffering time of the oldest packet in that buffer, or canceled if there is no remaining packet.

2. Whenever a timeout occurs for buffer b_{ij}, the oldest packet in this buffer is sent downstream uncoded. The timer is restarted with a duration equal to the remaining buffering time of the now oldest packet in the buffer. If no packets remain in the buffer, the timer is canceled.

The presented scheme considerably simplifies if we consider only cycles of maximum length 2, that is, if only bidirectional traffic is coded. This is reasonable if the CoMP data exchange is implemented pairwise, which is usually the case for JP CoMP. In this case, the graph construction and the search for cycles reduces to one single test if buffer b_{ji} is empty. In Section 5.2.1.3, we will present an analysis suggesting that the additional gain achieved by considering cycles with length greater than 2 is relatively small.

A simulative study of NC in PONs without additional buffering at the OLT is performed in [77]. Coding is performed among packets that are queued at the OLT due to congestion. We go one step further and not only encode packets that are buffered incidentally but intentionally buffer packets at the OLT.

Implementation at ONUs To decode incoming packets, an ONU has to keep a copy of each outgoing upstream packet whose destination is located within the same PON. Thus, like the OLT, ONU_i maintains a buffer b_{ij} for each destination ONU_j with $j \neq i$. Whenever ONU_i sends a packet p_i to ONU_j, a copy is stored in b_{ij}.

Now, consider the case where coding is performed over traffic cycles of size 2 (bidirectional traffic). In this case, one of the following can happen. (1) The OLT forwards p_i to ONU_j uncoded, (2) the OLT encodes p_i with another packet p_j going from ONU_j to ONU_i and multicasts the resulting packet $p_i \oplus p_j$ to both ONUs, or (3) p_i is dropped, e.g., due to congestion. We handle all three cases without the need for additional packet identifiers by exploiting once again the broadcast property of the PON's downstream

link. In the first case (no NC), the sending ONU_i overhears the uncoded downstream transmission of its own packet p_i and discards the locally stored copy. In the second case (NC), ONU_i receives the encoded packet $p_i \oplus p_j$. It is then decoded using the stored packet copy, which is removed from the buffer afterwards. In the third case (packet loss), it must be ensured that copies of packets that are lost are eventually removed from the buffer. To achieve this, in the first and second case, ONU_i not only removes the stored copy of p_i but also copies of packets that were sent to ONU_j prior to p_i. Those are exactly the lost packets.

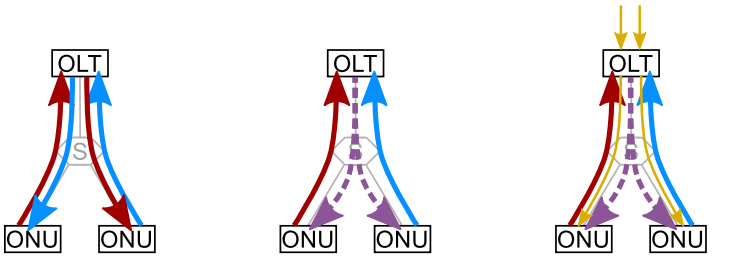
The proposed scheme requires that a source ONU is able to determine the destination ONU of each packet it sends upstream. Therefore, each ONU maintains a list of IP address $\leftrightarrow$ Logical Link ID (LLID) mappings, similar to an Address Resolution Protocol (ARP) cache. These mappings are created/updated whenever an ONU overhears a downstream frame that contains an IP datagram.

If coding shall be done for traffic cycles that contain more than 2 ONUs, additional mechanisms are required to inform the ONUs about how to decode encoded packets. An example for such a signaling mechanism and its evaluation can be found in [78].

5.2.1.2 Application scenarios

This section discusses several application scenarios that benefit from the proposed NC scheme in PONs.

General strategies In general, NC helps in scenarios where the PON downstream is a bottleneck. In this case, NC reduces downstream traffic whenever the traffic is cyclic and PON-internal, like interactive voice or video communication. This is illustrated for bidirectional traffic in Figure 5.15.



(a) No NC. Up- and downlink are saturated.

(b) NC active. Downlink utilization is reduced by 50 %.

(c) Free downstream capacity is used for additional traffic.

Figure 5.15: Exploiting NC gain when traffic is PON-internal and bidirectional.

Additionally, some applications allow to influence the traffic patterns to generate cyclic and PON-internal traffic when the source of data does not matter, like for P2P file transfers or information-centric networking [79]. Instead of downloading a data chunk from an external peer, it is fetched from a peer within the same PON. After replacing

external with internal traffic, the situation is identical to Figure 5.15a on the previous page. NC then allows to free downstream capacity that can be used, e.g., for additional data chunk downloads from peers that are located outside the PON.

Cooperative cellular networks The most relevant application in the context of this thesis is a cooperative cellular network, i.e., a network consisting of BSs using CoMP. In this case, data is exchanged between the cooperating BSs, which results in bidirectional traffic between ONUs.

For the different types of CoMP, introduced in Section 2.1, the NC approach can be applied as follows:

- Joint Processing (JP): In case of Joint Transmission (JT), it depends whether cooperating BSs use SU-MIMO or MU-MIMO. If SU-MIMO is used, the NC approach is only beneficial if two BSs *both* support each other by jointly serving at least *two* UEs that are *not* associated to just one of the BSs. Otherwise, no bidirectional traffic is created as only one BS forwards UE data to the other BS. The same is true for DCS techniques.

  In the most beneficial CoMP scheme, MU-MIMO JP, BSs serve a group of UEs using the same physical resources. Hence, the UE data and the CSI is exchanged among the cooperating BSs, which results in naturally symmetric traffic between the BSs that also has nearly identical data rates and packet sizes. This is a good base for applying the proposed NC scheme.

  For JP in the uplink, the situation is similar to the DCS and SU-MIMO case. Bidirectional traffic is generated only if two BSs both support each other.

- Joint Beam-Forming (JB): As no UE data is shared among cooperating BSs, NC can be applied only to exchanged CSI and signaling. The volume of this data is lower compared to UE data, which results in lower overall gains compared to JP.

- Joint Scheduling (JS): Here, the amount of exchanged data is even lower compared to JP. No CSI but only scheduling information is exchanged. Hence, expected gains from NC are rather low.

The highest gain from applying network coding is expected in scenarios where the JP CoMP scheme is applied, especially with MU-MIMO JT. This fact is caused by the high capacity requirements, which are hard to fulfill even in PONs. So, whenever two UEs are served cooperatively by two BSs and these BSs are connected to the same PON, network coding can be applied to save downstream capacity.

5.2.1.3 Evaluation

This section evaluates the gain of NC in different scenarios. It contains both analytical and simulative results.

Expected gain for bidirectional traffic To get a first idea about possible NC gains and requirements that have to be fulfilled in order to gain, we evaluate the scenario for bidirectional traffic analytically. For this, we assume that each traffic flow's rate R_{ij} per coding period $d_{\max}$ from ONU_i to ONU_j is a random variable with a normal distribution, $N(\mu, \sigma^2)$; all R_{ij} are identically distributed and independent both among ONU pairs and across successive timeslots. To normalize the variability of these flows and to ease the presentation of results later on, we will not use the standard deviation σ but the coefficient of variation $c_v = \frac{\sigma}{\mu}$, hence $R_{ij} \sim N\left(\mu, (\mu c_v)^2\right)$.

The amount of codeable traffic between ONU_i and ONU_j is the minimum of R_{ij} and R_{ji}. This is also the absolute gain G_{abs} (two packets from OLT to ONUs are replaced by one network-coded packet); its expected value can be calculated according to Equation (5.1). The expected value of the minimum of two independent, normally distributed random variables is solved in [80].

$$\begin{aligned} G_{\text{abs}} &= E\left[\min(R_{ij}, R_{ji})\right] = \mu - \frac{\mu c_v}{\sqrt{\pi}} = \\ &= \mu\left(1 - \frac{c_v}{\sqrt{\pi}}\right) \end{aligned} \tag{5.1}$$

Dividing G_{abs} by the total traffic produced by two ONUs whose traffic is encoded (both produce a total data rate of 2μ) gives the relative gain G_{rel}. It only depends on c_v and is shown in Equation (5.2).

$$G_{\text{rel}} = \frac{G_{\text{abs}}}{2\mu} = \frac{1}{2}\left(1 - \frac{c_v}{\sqrt{\pi}}\right) \tag{5.2}$$

The gain becomes larger the less variable the traffic flows are. But traffic can be smoothed by buffering it for a longer time. Hence, we consider time as additional dimension to relate the maximum buffering time at the OLT $d_{\max}$ to c_v. The effect of doubling $d_{\max}$ can be seen as combining the traffic of two adjacent time slots of length $d_{\max}$. Under the assumption that traffic in subsequent time slots of length $d_{\max}$ are i.i.d. and normally distributed as well, the random variable $\tilde{R}_{ij}$, which represents the traffic during both time slots, is the sum of $R_{ij} + R_{ij} \sim N\left(2\mu, 2\sigma^2\right)$. Equation (5.3) shows the traffic's new coefficient of variance $\tilde{c}_v$.

$$\tilde{c}_v = \frac{\sqrt{2\sigma^2}}{2\mu} = \frac{\sqrt{2}\sigma}{2\mu} = \frac{1}{\sqrt{2}}\frac{\sigma}{\mu} = \frac{1}{\sqrt{2}}c_v \tag{5.3}$$

If for a given $d_{\max}$ the resulting traffic has a variance of c_v, buffering for $2^l d_{\max}$ leads to a gain of $G_{\text{rel}}(l)$ (Equation (5.4)).

$$G_{\text{rel}}(l) = \frac{1}{2}\left(1 - \frac{\tilde{c}_v(l)}{\sqrt{\pi}}\right) = \frac{1}{2}\left(1 - \frac{1}{\sqrt{2}^l}\frac{c_v}{\sqrt{\pi}}\right) \tag{5.4}$$

This equation allows to calculate the required $d_{\max}$ (in terms of factor exponent l) to achieve the desired NC gain for traffic with coefficient of variation c_v. Figure 5.16 shows the relation between c_v, l, and the resulting G_{rel}. To achieve a large gain, the traffic has to be smooth (small c_v) or larger buffers and long delays are required (large l).

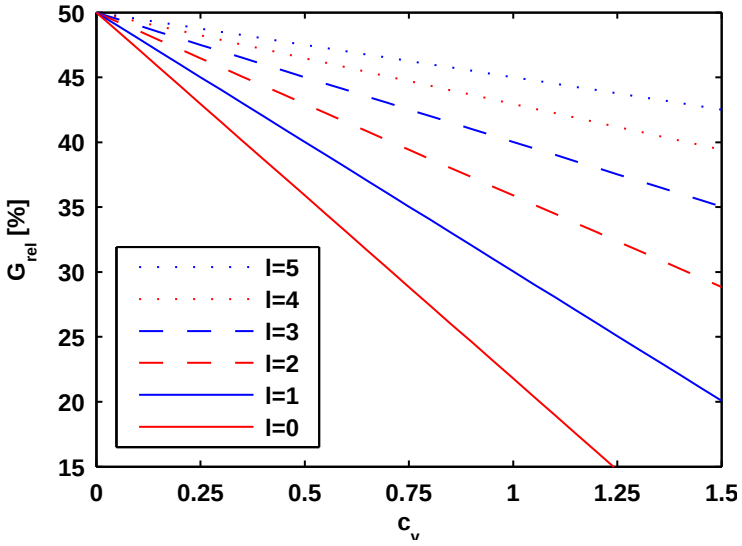


Figure 5.16: Relative NC gain G_{rel} depending on c_v for different factor exponents l.

Expected gain for arbitrary traffic cycles As mentioned in Section 5.2.1.1, NC can be applied for any cyclic PON-internal traffic. The gain, however, decreases as the cycle size increases, and detecting larger traffic cycles is more difficult. This leads to the question whether spending effort for detecting large traffic cycles is beneficial at all.

To answer this question, we simulated the abstract traffic model from the previous section in a PON consisting of 8 ONUs. We analyzed the traffic to detect cyclic flows with up to $c_{\max}$ involved ONUs and finally calculated the mean gain when applying NC for the detected cycles. The maximum cycle size $c_{\max}$ is varied from 2 (bidirectional flows) to 5. Coding opportunities are detected in a greedy way to reduce complexity as described in Section 5.2.1.1: Smaller cycles (higher gains) are searched for first. Thereafter, remaining traffic is analyzed for larger cycles.

According to the traffic model, each ONU pair $\text{ONU}_i \rightarrow \text{ONU}_j$ is assigned a certain data rate r_{ij}, with $R_{ij} \sim N(6, \sigma)$. Different application/traffic types with higher and lower symmetry property are simulated by varying σ. The resulting gains and analytic values for $c_{\max} = 2$ are shown in Figure 5.17 on the next page.

The plot shows the expected behavior that smaller values for σ (i.e., more balanced/ symmetric traffic) lead to higher gains. Furthermore, increasing $c_{\max}$ only shows additional gains for high σ values. These gains, however, are only on the order of a few percents. Hence, in following evaluations, we only consider bidirectional traffic.

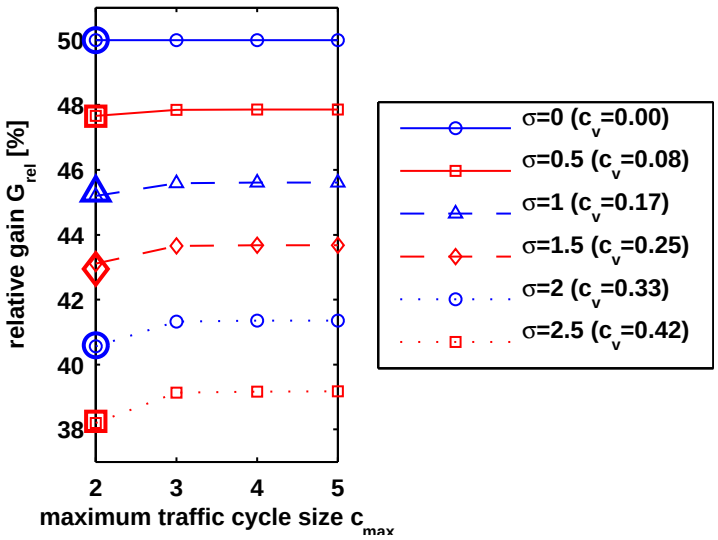


Figure 5.17: Expected NC gain depending on $c_{\max}$ and σ. Besides the simulation results, analytic values according to Equation (5.2) are plotted (large markers) for $c_{\max} = 2$ to validate the model.

Performance for real-world traffic We implemented the proposed NC extension for bidirectional traffic using the discrete event-based simulation framework OMNeT++.

In the evaluated setting, clients are connected to ONUs via intermediate routers, while the OLT is connected to the Internet via a gateway router. ONU routers forward all upstream traffic to the gateway router, which either forwards it to the Internet or, in case of internal traffic, to the router of the destination ONU. As a result, the PON operates as a distributed switch, i.e., it is transparent to the clients. All links (optical and non-optical) operate at a data rate of 1 Gbit/s.

For the scheduling of upstream transmissions, we used the *Interleaved Polling with Adaptive Cycle Time (IPACT)* [81] Dynamic Bandwidth Allocation (DBA) algorithm, which assigns timeslots to ONUs in a round-robin manner. We have chosen this algorithm as it is very efficient in terms of capacity utilization.

In order to avoid joint encoding of very large with very small packets, e.g., data and acknowledgment packets, as this would lead to a low NC gain, we only feed packets into the encoder that are larger than 100 bytes. This procedure has the advantage that it is very easy to implement. On the other hand, it clearly wastes some coding opportunities. To overcome this drawback, flow synchronization techniques [78] can be applied before feeding traffic into the encoder and joining packets together.

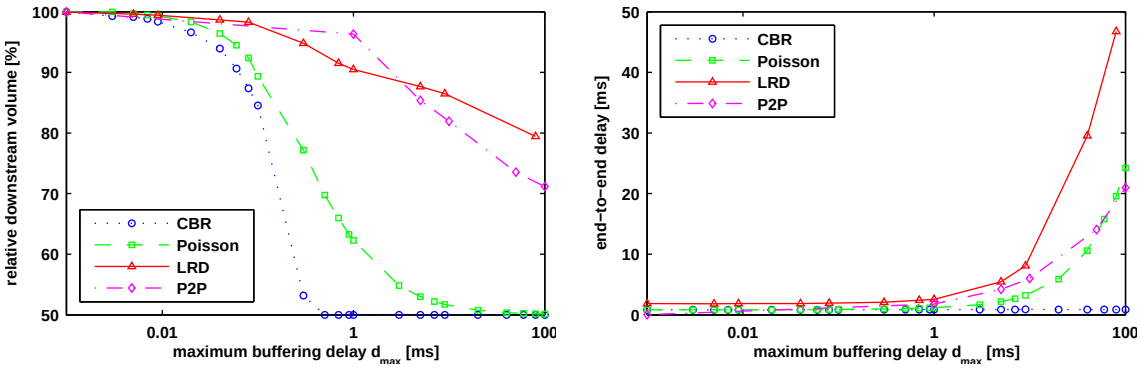
Obviously, $d_{\max}$ is a critical parameter for the performance. If $d_{\max}$ is too short, fewer coded packets are generated and the NC gain decreases. If $d_{\max}$ is too high, real-time applications suffer from delay and buffers need to be larger. Further, a high value of $d_{\max}$ increases the RTT variance and thus decreases TCP throughput.

We simulated two different traffic scenarios. In the first one, two ONUs exchange traffic with a rate of 5 Mbit/s and a packet size of 200 bytes (resulting in a mean inter-

arrival time of 320 μs). With these parameters, we simulated three different types of streams: Constant Bit Rate (CBR) and Poisson streams, and streams with Long-Range Dependent (LRD) inter-arrival times. CBR traffic is generated by sending packets with identical inter-packet times of 320 μs. The Poisson stream is produced by using exponentially distributed inter-packet times with a mean of 320 μs. Finally, a long-range dependent traffic model is used to generate bursty traffic, which is an important property of packet-switched data, e.g., in the Internet [82]. Note that higher data rates result in similar results but increase the simulation running time a lot. This is the reason why we have chosen these rather low data rates.

Some of today's popular P2P protocols, like BitTorrent, have built-in mechanisms that inherently generate bidirectional traffic, e.g., "tit-for-tat" policies. Furthermore, in order to reduce inter-provider costs and to increase performance, many research activities focus on controlling P2P transfers to exploit locality on provider level by introducing a so called P2P oracle [83, 84]. That is the reason why we have additionally investigated a second traffic scenario, where users participate in a 760 Mbyte BitTorrent-like P2P download that is subdivided into 152 chunks of 5120 kbyte. Each user is allowed to retrieve up to 10 chunks in parallel and each time a peer intends to start a new chunk download, an oracle is contacted to provide a suitable PON-internal source peer.

Results of both traffic scenarios are shown in Figure 5.18. Figure 5.18a illustrates the total amount of data sent downstream by the OLT, normalized to the amount of data that is sent when NC is disabled. Figure 5.18b shows the average end-to-end delay experienced by PON-internal packets.



(a) Relative downstream volume

(b) End-to-end delay

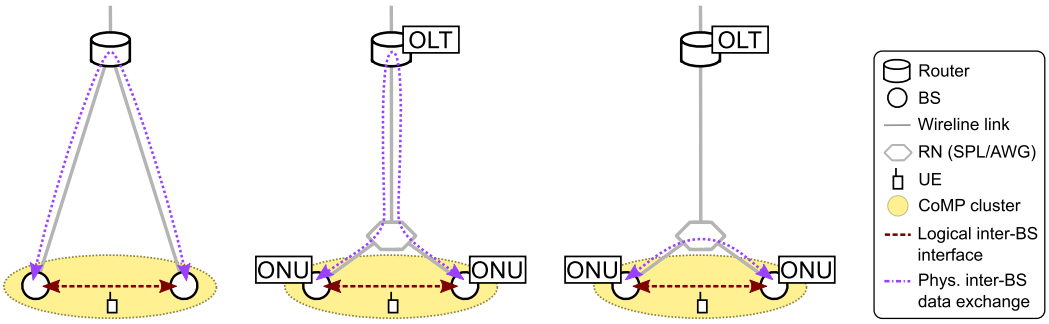
Figure 5.18: Impact of NC on the relative downstream volume and end-to-end delay for CBR, Poisson, and LRD streams, as well as for TCP-based P2P downloads.

The simulations show that NC reduces the mean downstream utilization by up to 50 %, or, in other words, increases downstream throughput by 50 %, if enough coding opportunities are present. In the case of two CBR streams, a value of $d_{\max} = 1$ ms is enough to achieve the maximum possible gain without sacrificing delay. For the P2P download, the mean downstream utilization is reduced by up to 30 %. The smaller gain compared to the CBR scenario is due to the fact that a certain fraction of data has to

be retrieved from outside of the PON, where NC cannot be applied. Further, due to the burstiness of TCP traffic, packet inter-arrival times can temporarily exceed $d_{\max}$, even if the latter is set to a high value.

5.2.2 Inter-ONU data exchange

CoMP, and especially the JP CoMP scheme, requires to exchange UE data and/or signaling information like CSI via the backhaul network that interconnects the BSs. A problem arises from the fact that neighbor BSs are usually not physically connected but have only a logical interface, which transports data via the access network, e.g., a PON. In conventional backhaul access networks, as shown in Figure 5.19, inter-BS data exchange is realized via a central switching node to minimize hardware costs.



(a) Generic scenario (b) PON scenario (c) Ext. PON scenario

Figure 5.19: Logical inter-BS interface and physical inter-BS data exchange in a generic, a conventional PON-based, and the proposed extended PON-based backhaul network.

This way of implementing inter-BS data exchange (Figure 5.19a and 5.19b) has been no issue for conventional cellular networks because the latency requirements for the inter-BS interface is on the order of 20 ms maximum with a typical average of 10 ms [85]. This limit origins from data forwarding for UE hand-over and control plane support for radio resource management. Capacity-wise, the BSs need to exchange just the UE data for UEs that are handed over to another BS. This traffic amount is estimated at 4 % of the total cell traffic on average, going up to 10 % for cells serving highly mobile users [86]. For CoMP, however, such delay between neighbor BSs is too high and the available capacity is too low to transport the additional high amount of UE and signaling data between cooperating BSs. Consequently, CoMP requirements get violated.

One step towards solving these issues is to use WDM PON technologies in the backhaul access network. This increases the capacity available on the links from the central node (OLT) to the BSs (ONUs), and hence between the BSs, but the delay problem for inter-BS data exchange still persists.

One way to provide *physical* inter-BS links to reduce the latency is to use microwave point-to-point connections between the BSs. Such a mesh solution, however, requires

a lot of additional hardware and additional spectrum licenses for the used microwave bands. Moreover, the limited capacity in microwave bands make it more challenging to offer more than 1 Gbit/s, which is required for LTE-A and beyond. Using higher frequencies, such as millimeter waves, is an alternative, but introduces even higher costs than microwave. No matter which wireless technique is used for point-to-point inter-BS links, the link quality is always worse compared to fiber-optic links due to its susceptibility to environmental influences like weather.

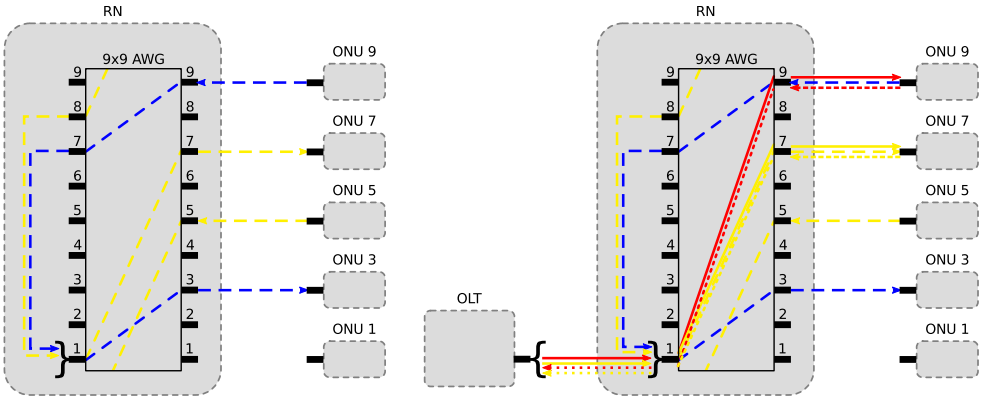
In the following section, Changsoon Choi and I propose an extension to the standard WDM PON architecture that enables data exchange between ONUs, i.e., BSs, without going via the OLT, which is far away (Figure 5.19c on the previous page). This way, the delay between ONUs in a WDM PON is reduced and, in addition, further capacity is provided for inter-ONU data exchange. The approach does not require additional fiber and is compatible with conventional WDM PONs, which allows a cost-effective implementation. These benefits are helpful to support CoMP and the, anyways expected, smaller cell sizes in future cellular networks, which ensue much more UE hand-overs.

5.2.2.1 WDM PON architecture with pure optical inter-ONU links

From the performance point of view, dedicated fiber-optic links forming a mesh between neighbor BSs would be optimal. This approach is of course not practical. That is why recently the realization of physical inter-ONU links, which skip the detour via the OLT, has been proposed for TDM PONs [87]. The idea is to use additional optical splitters at the Remote Node (RN) of a TDM PON for distributing optical inter-ONU signals without going up to the OLT. This promises lower latency for inter-ONU traffic as packets do not pass through the OLT but directly go to the destination ONUs via the RN. Furthermore, propagation time in the fiber is reduced. The drawback of the proposed approach is that it is based on broadcasting, which is not sufficient for supporting dynamic clustering approaches for CoMP. Furthermore, as the sent optical signal is split up to multiple destination ONUs, which even do not need the signal, the optical SNR at the destination ONUs is reduced, which limits the achievable data rate. This is especially problematic as the capacity of TDM PONs will not be enough to support JP CoMP for systems like LTE-A and beyond. This is where WDM PONs come into play. They have enough capacity thanks to individual wavelengths for individual ONUs. In the following, we propose a WDM PON architecture that supports the direct data exchange between ONUs via the RN to have the advantages of reduced latency and additional PON-internal capacity in WDM PONs. The direct transmission is possible from any ONU to any other ONU in the same PON, not limited to pre-defined groups.

To achieve the proposed features, two changes are required: at the ONUs and at the RN. We first describe the changes to the RN. An overview showing the differences between the conventional WDM PON architecture and the extended RN for the described direct inter-ONU traffic is given in Figure 5.20 on the facing page.

The inter-ONU data exchange is done in a separate optical waveband, different from the used bands for downlink and uplink transmissions (indicated by the different line styles for the optical signals in Figure 5.20 on the next page). This has two advantages:



(a) Overview of optical signal flow for direct inter-ONU data exchange. The OLT is not involved, only the RN. ONU 9 sends to ONU 3 and ONU 5 to ONU 7.

(b) Full optical signal flow, including uplink, downlink, and inter-ONU data. In addition to the inter-ONU data exchange, analogue to the left figure, ONUs 7 and 9 also perform uplink and receive downlink transmissions to/from the OLT in parallel.

Figure 5.20: Modified RN architecture permits to exchange data between ONUs without going over the OLT. Different line styles indicate different optical wavebands, different colors stand for individual wavelengths within the waveband.

First, the inter-ONU transfers are isolated from other transmissions (as it is done for up- and downlink already), which guarantees downward compatibility, and second, inter-ONU capacity is not taken from the up- or downlink capacity.

To achieve the direct inter-ONU transfers, we exchange the conventional $1 \times N$ Array Waveguide Grating (AWG) by an $N \times N$ AWG ($N = 9$ in the example shown in Figure 5.20). Furthermore, the $N-1$ new ports on the left side, where also the feed fiber from the OLT is connected to port 1, are connected to a passive optical coupler and fed again into port 1, in addition to the optical signals coming from the OLT. This means that optical signals arriving at the left ports 2 to N of the AWG (the inter-ONU signals) are combined and sent down to the ONU again. In case inter-ONU broadcasting should be supported, an additional Fiber Bragg Grating (FBG) is required to avoid inter-ONU signals arriving at the main port 1 to be forwarded to the other ports.

The new inter-ONU waveband is chosen such that the port diversion properties of the AWG are equal to the one in the downlink waveband (L-band). This means that if $\lambda_{x,\text{down}}$ (wavelength x in the downlink waveband) and $\lambda_{x,\text{inter}}$ (wavelength x in the inter-ONU waveband) enter the AWG at the same port, they also leave the AWG at the same port on the other side. Now, if an ONU_x wants to send directly to ONU_y, it sends on $\lambda_{y,\text{inter}}$. The optical signal arrives at the RN and is diverted within the AWG to one of the uplink ports 2 to N. Then, it is re-fed into the main port 1 such that the signal reaches the desired ONU. This is illustrated in Figure 5.20a, where ONU_9 sends to ONU_3 and ONU_5 to ONU_7. Note that AWGs have cyclic properties, which is exploited in the transmission from ONU_5 to ONU_7.

Thanks to the individual wavebands for uplink (C-band), downlink (L-band), and inter-ONU transfers, all these kind of transfers can happen at the same time without interfering. This is shown in Figure 5.20b on the preceding page where ONU_9 and ONU_7 have concurrent up- and downlink transmissions to the ongoing inter-ONU transmissions.

Due to the short fiber transmission distance for the optical inter-ONU signals, the attenuation is much lower compared to the links between the ONUs and the OLT. This allows to omit optical amplifiers, like required for the up- and downlink transmission. This, in turn, allows to use any waveband for the inter-ONU data exchange whose wavelength separation to the used wavelength band in the downlink corresponds to the Free Spectral Range (FSR) of the AWG, as the unavailability of amplifiers does not have to be taken into account.

ONUs in WDM PONs usually use a Colorless Laser (CL), which is a laser whose output wavelength can be tuned, as an optical source. This way, the ONU can be assigned arbitrary wavelengths within the used waveband for the uplink transmission. For receiving downlink signals, a wideband Photo Detector (PD) is used. To allow an ONU to simultaneously send and receive upstream and downstream data as well as inter-ONU data, the ONU has to be extended. An additional CL and PD is added for the inter-ONU data exchange. Furthermore, traffic leaving the ONU to the OLT or to another ONU has to be filtered and fed into the corresponding transmitter. This setup is depicted in Figure 5.21.

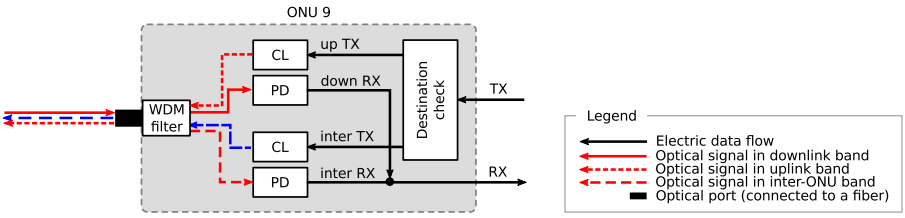


Figure 5.21: Extended ONU architecture including an additional PD and CL for inter-ONU traffic. The optical signals correspond to the signals of ONU_9 in Figure 5.20b on the previous page.

Theoretically, it is also possible to use only one CL and PD for both the uplink and inter-ONU transmissions. This, however, requires a low CL wavelength tuning time to avoid loosing capacity while switching from the uplink to the inter-ONU wavelength and back. In addition, such a scheme requires a Medium Access Control (MAC) protocol to avoid collisions between inter-ONU and downlink signals at the ONUs.

In addition to the inter-ONU traffic, the additional waveband can be used to accommodate additional downlink traffic to the ONUs in case the inter-ONU data exchange is not required at the moment. Optical signals sent by the OLT in the inter-ONU waveband reach the ONUs as well. This permits a dynamic on-demand use of the capacity of the inter-ONU waveband.

The proposed inter-ONU extension for WDM PONs provides high-capacity and low-latency point-to-point links between ONUs of the same PON. For point-to-multipoint

transmission between ONUs, the wavelength tuning time of the CL has to be minimized because it needs to change every time the destination ONU changes. Furthermore, optical inter-ONU broadcasting to all ONUs is also possible if a broadspectrum optical source, like a Super-Luminescent Light Emitting Diode (SLED), are used at the ONUs instead of a CL. Such optical sources contain all wavelengths in a given waveband, therefore optical inter-ONU signals are distributed to all ONUs by the AWG in the RN (Figure 5.20 on page 101). An SLED is much cheaper than a CL, which even permits to deploy both kinds of optical transmitters in one ONU. To avoid optical collisions for such inter-ONU transmissions, a multiple access scheme can be used, like Time Division Multiple Access (TDMA) or Sub Carrier Multiple Access (SCMA).

The proposed architecture is fully compliant with the conventional WDM PON, i.e., conventional ONUs and the OLT can be operated in parallel to the extended ones that support direct inter-ONU traffic.

5.2.2.2 Evaluation

To get an idea of the influence of the proposed WDM PON extension for direct inter-ONU transfers on the feasibility of JP CoMP, we did several simulations. Assumptions and results will be discussed in the following subsections.

Assumptions and system model The used system model is similar to the one already used in the evaluations in previous sections. For details, please refer to Section 4.2.1. Differences are summarized in the following.

We generate a square grid of 2000 BSs with hexagonal cells and an inter-BS distance of 500 m. Each of these BSs is attached to one ONU of a WDM PON, using a splitting factor of 40, i.e., 40 ONUs (and hence BSs) are attached to one OLT via an RN. The BSs/ONUs are assigned to OLTs based on their proximity and the RN is positioned such that the fiber lengths between the ONUs and the RN are minimized in each PON. The RN is connected to an OLT. All OLTs are colocated in one location, which is a common practice of network operators to reduce operation costs. The location of the so-called *OLT hotel* is determined by the maximum possible transmission distance within the WDM PON. It depends on the actual WDM PON technology [44], thus we leave it as a parameter $l_{\text{PON}} \in \{30, 40, 50\}$ km. The fiber propagation delay for a fiber of length s is again calculated as $s \cdot \frac{1.45}{c}$. IP processing at the OLT takes 0.1 ms. In the conventional PON architecture, if two communicating ONUs belong to the same PON, IP processing delay occurs once at the OLT. If the ONUs belong to different PONs, IPs processing occurs at both PONs' OLTs in the OLT hotel. Propagation delay between the co-located OLTs is neglected due to the short distance that has to be covered within the hotel.

The second simulation parameter is the WDM PON link capacity. We define f_{lcap}, the normalized link capacity between the OLT and each ONU, as the ratio between the PON link capacity and the average data rate required for CoMP at each BS, akin to the definition in Section 4.2.1. For simplification, we consider 1 Gbit/s for UE data and 100 Mbit/s for CSI exchange required for MU-MIMO processing at each BS, which corresponds to an LTE-A system. The link capacity is varied between 1 Gbit/s, 2.5 Gbit/s,

and 5 Gbit/s, which are typical values for WDM PON systems. Downlink and uplink capacities are identical.

The third parameter is the wavelength tuning time of the CL used in the ONUs. It influences the actual link capacity and link delay, as the link cannot be used while tuning the wavelength. A low tuning time is beneficial but notably increases the costs for each ONU. We evaluate the range from 0 to 0.4 ms. If not stated otherwise, we set the tuning time to 0.1 ms.

The last parameter is the use of the proposed direct inter-ONU data exchange extension. Without this extension, all data exchange between ONUs, and hence BSs, have to happen via the OLT hotel. With the extension, data exchange between ONUs attached to the same OLT can be done directly via the RN without going to the OLT itself.

UEs are uniformly distributed in the network such that one UE is associated to each BS. In each simulation run, one UE is considered to check its CoMP cluster feasibility. The wireless clustering is done based on a large-scale fading channel. For this, we generated the shadow fading map of 61 BSs where the UE under consideration is located in the cell of the BS located in the center of the 61 BSs [61, 60]. A log-normal shadow fading model with a standard deviation of 8 dB is used. The shadowing correlation for the serving BS is 1 and for other BSs 0.5. The path-loss model used in the simulation is $\mathrm{PL} = 128.1 + 37.6 \cdot \log(L)$, where L is distance in kilometers.

In alignment to the previous evaluations, we again use the metric of *wireless cluster feasibility*. It is the fraction of the number of BSs that are desired from the wireless point of view to the number of BSs that actually *can* participate in the cooperation, considering the limitations of the backhaul network. Averaging this fraction for all UEs in the input scenario leads to the main metric. To determine the desired wireless clusters and the feasible clusters, we use the system as introduced in Section 4.3. It includes wireless long-term pre-clustering, which finds the desired clusters, and backhaul network pre-clustering for checking the backhaul network's capability.

Results In a first step, we evaluate how the wireless cluster feasibility depends on the desired cluster size, the maximum reach of the WDM PON l_{PON}, and the normalized link capacity f_{lcap}. The results are plotted in Figure 5.22 on the next page.

The plot shows that increasing the desired cluster size decreases the cluster feasibility due to the increased capacity requirements as more BSs need to join the clusters. It also shows that the cluster feasibility degrades when increasing the maximum PON reach l_{PON}, which is caused by the increased propagation time within the fiber. In the FTTx context, there have been considerable efforts to increase the PON reach (and hence the coverage) for lowering CAPEX and OPEX. The results given in Figure 5.22 on the facing page, however, imply that this approach introduces a problem to use such long-reach WDM PONs for future backhaul networks that exploit CoMP techniques.

Increasing the link capacity in the WDM PON is one way to increase the CoMP cluster feasibility. This is shown in Figure 5.22 on the next page where the cluster feasibility is improved for $l_{\mathrm{PON}} = 30$ km and $l_{\mathrm{PON}} = 40$ km. For $l_{\mathrm{PON}} = 50$ km, however, only a minor improvement is visible, as in this scenario, the feasibility is mainly limited

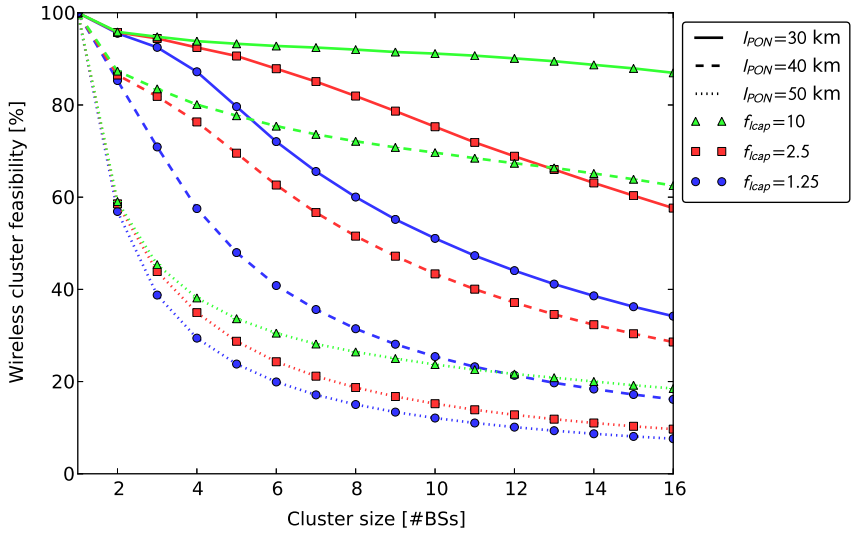
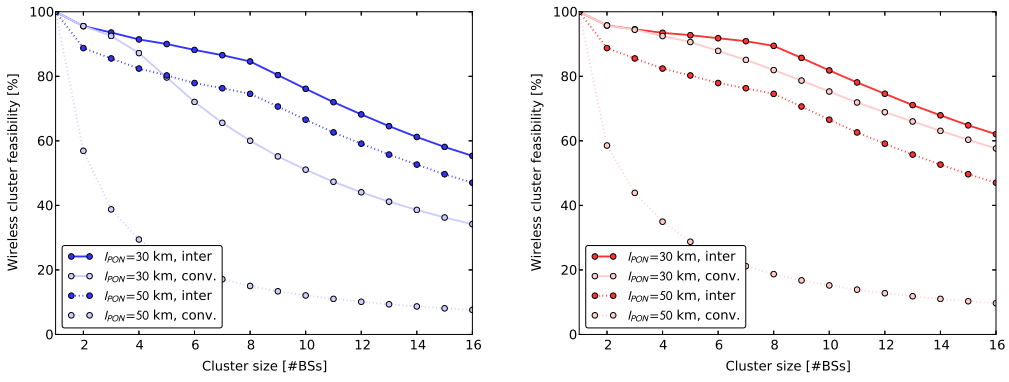


Figure 5.22: Wireles cluster feasibility depending on desired cluster size, maximum PON reach l_{PON}, and normalized link capacity f_{lcap}. The line style identifies the used value for l_{PON}, the line color and marker style for the value for f_{lcap}.

by the high propagation delay caused by the long fiber lengths. This insight implies that in order to properly support CoMP, we need an optical bypass to avoid the long propagation time at least within the same PON tree. This is achieved with the proposed direct inter-ONU links.

The influence of the direct inter-ONU transfers are illustrated in Figure 5.23. The plot compares the wireless cluster feasibility for a conventional WDM PON and a WDM PON with the proposed inter-ONU links, both assuming a maximum PON reach of $l_{\text{PON}} = 30\,\text{km}$ and $50\,\text{km}$ to show the advantage in long reach PON deployments.



(a) $f_{\text{lcap}} = 1.25$ (b) $f_{\text{lcap}} = 2.5$

Figure 5.23: Wireless cluster feasibility with and without the proposed optical inter-ONU data exchange WDM PON extension.

There is a clear improvement in cluster feasibility visible, which would directly result in an increased UE throughput. This benefit comes from the ability of bypassing the OLT via the RN, resulting in a much shorter delay.

In case of multicast transmission, i.e., one ONU wants to send the same data to multiple ONUs in the same PON, the proposed inter-ONU extension without additional broadband SLED source requires wavelength tuning for each ONU in the multicast destination group. Tuning or switching the optical wavelength in a CL usually takes a considerable amount of time. Therefore, it is important to investigate its influence on the CoMP cluster feasibility when using the proposed direct inter-ONU links. The resulting plot is shown in Figure 5.24.

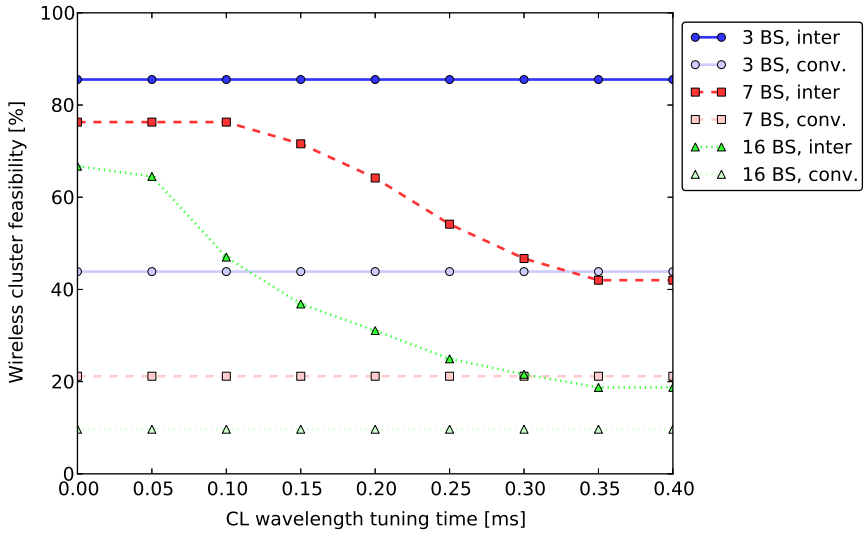


Figure 5.24: Wireless cluster feasibility with and without the proposed optical inter-ONU data exchange WDM PON extension depending on the CL wavelength tuning time. Small (3 BSs), medium (7 BSs), and large (16 BSs) desired wireless clusters are taken into account.

In general, the cluster feasibility degrades when the CLs' wavelength tuning time increases. For the small desired clusters consisting of 3 BSs, the feasibility does not degrade. The reason is that in this case only two inter-ONU transmissions are necessary from the serving BS to the other cluster members, i.e., only one optical wavelength switch is required. This single switch, which takes maximal 0.4 ms in this evaluation, can be conducted without violating the JP CoMP time constraint of 1 ms. Nevertheless, an improvement compared to conventional WDM PONs is always visible in the evaluated wavelength tuning time range. This shows that it is crucial to minimize the wavelength tuning time to support larger CoMP clusters. Even though several publications have reported CLs with tuning times on the order of microsecconds or even nanoseconds [88], most CLs used in WDM PON ONUs have wavelength tuning times of several milliseconds since conventional WDM PONs do not require such short tuning times.

Besides the mentioned lasers with microsecconds or even nanoseconds tuning times, using optical broadcasting in the proposed optical inter-ONU links could be one solution to alleviate the stringent wavelength tuning time requirement. As stated earlier, a broadspectrum optical source, like a SLED, can be added to the ONUs and used as inter-ONU optical transmitter. It is much cheaper than a CL but limits direct inter-ONU transfers to one source at a time because the whole inter-ONU waveband is occupied by a single transmission. Furthermore, the maximum modulation speed of SLEDs is usually lower than that of CLs.

Cost estimation In a conventional WDM PON, the following costs occur to serve a single ONU: 75 $ for the ONU unit, 55 $ for the OLT transceiver unit, and 20 $ for the AWG port at the RN [89]. This results in total costs of 3520 $ for the OLT, 1280 $ for the RN, and 4800 $ for the ONUs, assuming a network with 64 ONUs. In total, the equipment costs are 9600 $. The costs assume CLs at both the OLT and the ONUs.

To implement the small version of our approach, which reduces the inter-ONU latency without providing additional capacity, just an additional passive optical coupler is required at the RN. This coupler costs less than 50 $ for 64 ports, resulting in an cost increase of the RN with 64 ports from 1280 $ to 1330 $, which corresponds to 4 %. The relative increase of the total costs is roughly 1 %.

For the full implementation, leading to reduced inter-ONU latency *and* additional inter-ONU capacity, the ONUs need to be extended by an additional transceiver unit as well. The costs for such a component is roughly 50 $, like the costs for an optical transceiver at the OLT that includes additional 5 $ for switching components. This causes a cost increase for each ONU from 75 $ to 125 $, i.e., a raise by 67 %. In total, the system cost for a PON consisting of 64 ONUs is increased from 9600 $ to 12580 $, which corresponds to 31 %.

5.3 Summary

The results presented in Chapter 4 pointed out that desired BS clusters are not feasible in basically any realistic backhaul network deployment, even assuming future optical technologies. This chapter has addressed this problem and presented several mechanisms to improve cluster feasibility. These mechanisms are generic, i.e., they can be applied independently of the deployed backhaul network technology, or are tailored to PONs, which are a promising technology for implementing future backhaul networks.

The dynamic *serving BS reassignment* mechanism improves CoMP feasibility in any kind of backhaul network scenario. The mechanism does not require additional functions in the network as it exploits the hand-over features of, e.g., LTE or LTE-Advanced. This makes this mechanism very powerful, e.g., in the transition phase while backhaul networks are upgraded according to the requirements of the future mobile networks' requirements. But also after the upgrade, it still promises big gains in hot spot scenarios. Furthermore, the mechanism enables to overcome the missing multiplexing capability of WDM PONs by balancing load among BSs. Hence, combining CCR and WDM PONs

as backhauling technology provides a powerful and flexible basis for future cooperative mobile access networks.

The second generic mechanisms for improving CoMP feasibility is *backhaul network reconfiguration*. This approach combines the backhaul network pre-clustering heuristic from Section 4.3.2 with the available flexibilities of certain backhaul network implementation technologies. Based on the output of the pre-clustering heuristic, the system exploits flexibilities of backhaul network technologies to adapt the backhaul network to the wireless requirements. This way, the currently deployed backhaul hardware can be optimally exploited, which is especially important during the transition phase when the backhaul network limits the feasibility of CoMP.

The first mechanism for scenarios with PONs-based backhaul networks exploits *network coding* techniques. The evaluations have shown that the NC extension for TDM PONs increases downstream throughput by up to 50 %. Thus, PON capacity can be significantly increased without expensive hardware upgrades. Due to the resulting symmetric traffic between BSs in MU-MIMO JT CoMP, this technique promises high gains as it reduces the downstream load in the TDM PON interconnecting the BSs. The additionally introduced delay for buffering the packets to be coded can be limited by setting $d_{\max}$ accordingly. This ensures that CoMP's latency constraints are still fulfilled.

Finally, a novel extension for WDM PONs has been presented, which supports *direct optical transfers* between ONUs attached to the same PON. This way, latency is reduced and additional capacity is provided for unicast and multicast inter-BS data exchange. The system is fully downward compatible with conventional WDM PON systems. The extension increases the size of feasible CoMP clusters and allows to exploit the high reach of WDM PON systems. Hence, such systems can be used for backhauling in future cellular networks that use CoMP, which is hardly possible with conventional WDM PON systems.

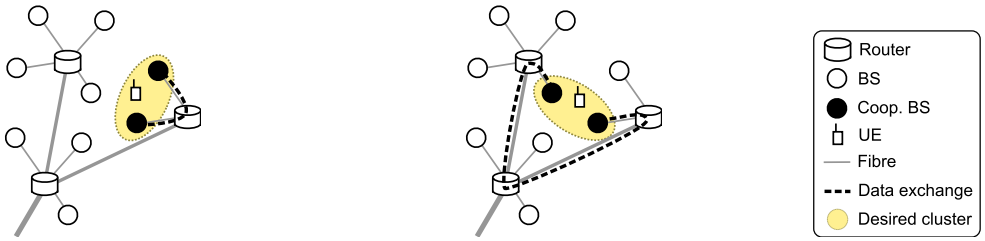
Chapter 6

Deploying CoMP on metro scale

Contents

In small-scale hot-spot scenarios, like airports or shopping malls, CoMP backhaul network requirements can be fulfilled by deploying Remote Radio Head (RRH) solutions where there is just a single, big BS unit with many remote antennas connected via point-to-point optical fiber. Alternatively, when multiple BSs are distributed in a small area, like pico cells or small macro cells, they can be interconnected using optical technologies, like Wavelength Division Multiplexing (WDM) Passive Optical Networks (PONs) [44]. This technology, especially with the extensions proposed in Section 5.2 and the generic mechanisms from Section 5.1, provides high link capacity and has very low latency for traffic that stays within a single PON.

When using cooperation techniques on a larger scale, e.g., to support JP among adjacent macro cells, BSs will be involved that are connected to different PON trees. In this case, inter-BS cooperation is challenging as UE data, CSI, and signaling data need to go one layer up in the wireline backhaul network hierarchy, i.e., via a metro or core network, before reaching the cooperating BS. As a consequence, a significantly larger part of the mobile network is involved in exchanging additional traffic with strict latency constraints (Figure 6.1). Queuing and processing delay at intermediate nodes like IP routers limit cooperation feasibility.



(a) Only local data exchange is necessary, e.g., via a PON.

(b) Data exchange via metro network is required.

Figure 6.1: JP cooperation scenarios and impact on the mobile backhaul infrastructure.

The two techniques of network coding in PONs (Section 5.2.1) and direct optical inter-ONU data exchange (Section 5.2.2) do not help in a metro-wide scenario. They are only relevant for CoMP among BSs that are connected to the same PON tree.

On the other hand, backhaul network reconfiguration (Section 5.1.2), serving BS reassignment (Section 5.1.1), and backhaul network pre-clustering (Section 4.3.2) are able to improve CoMP feasibility on larger scales where clusters span over multiple access network parts. Nevertheless, the achievable gains of these techniques depend on the capabilities of the underlying backhaul network.

In this chapter, rather than dealing with and combating the limitations of an already deployed backhaul network, Luca Scalia and I have designed a cost-effective backhaul network architecture which inherently enables JP on a metro-wide scale, i.e., the architecture fulfills JP latency requirements for all connected BSs. For this, we have developed models of different backhaul network architectures to check whether they are able to support JP. The evaluated architectures either reuse existing metro ring infrastructures (Section 6.2) or are designed from scratch to include the requirements of JP (Section 6.3). For each of these architectures, we evaluate the area that can be covered such that JP is feasible. During the evaluation, we vary various system parameters and identify advantages and disadvantages of the different architectures.

The results of our analysis show that there exists an inevitable trade-off between reusing current infrastructures, enabling cooperation between all adjacent cells, and the limits of upcoming WDM PON technologies. Based on these insights, we consider possible enhancements to improve JP feasibility. Furthermore, we propose and analyze an architecture in Section 6.4 that reuses existing infrastructure to a large extent but

also exploits future optical technologies to provide a high coverage area in which JP can be applied among arbitrary BSs.

6.1 Modeling assumptions

Our study concentrates on the latency requirements that need to be fulfilled for JP. Latency is the main focus as it cannot be reduced as easily as capacity can be increased on the metro scale. For different metro scale backhaul architectures, we analyze the maximum area that can be covered by these architectures while still fulfilling the latency requirements of JP between BSs connected via these architectures. This is done by creating mathematical models for each of the architecture candidates, which will be described in detail in the following sections. The assumptions that all models have in common are listed in the following.

Passive Optical Networks (PONs) We use PONs [44] to connect the BSs to the wireline network infrastructure. This technology is the cheapest way to provide enough capacity for future JP-enabled systems.

In our scenario, deployed PONs have a splitting factor of $\mathrm{sf}_{\mathrm{PON}} = 32$, i.e., each OLT is connected to 32 ONUs via the remote node. In the evaluations, the splitting factor only affects the calculated number of PONs that are required to cover a certain area. Other properties are independent and are valid for any type of PON.

IP processing IP routers in the network need to make IP forwarding decisions. This introduces a delay of $d_{\mathrm{IP}} = 0.1\,\mathrm{ms}$ [63, 90], which also contains the queuing delay in the router. Notice that using technologies like Multiprotocol Label Switching (MPLS) does not lower the queuing delay [91]. In practice, both IP and MPLS show similar performance in terms of overall packet processing delay. Hence, all following results that are based on d_{IP} are also valid for label switching deployments, such as MPLS.

Wireless parameters We assume an MU-MIMO setup where the overall delay that is required for the wireless processing (Multiple Input Multiple Output (MIMO) processing, traversing the transceiver chains, etc.) is estimated at $d_{\mathrm{wls}} = 0.1\,\mathrm{ms}$. Furthermore, the maximum tolerable delay from measuring CSI at the UE until the point in time when the corresponding cooperative transmission takes place is $d_{\max} = 1\,\mathrm{ms}$.

BS deployment We assume an urban BS deployment scenario with an inter-BS distance of 1000 m such that each BS wirelessly covers an area of $A_{\mathrm{BS}} = 0.785\,\mathrm{km}^2$.

Metrics In the following three sections, we analyze, for three different metro-wide backhaul architectures, the area A that can be covered using JP. The size of A changes for the architectures due to their different topologies and technologies used to interconnect the BSs, which in turn influences to which degree JP latency requirements can be

fulfilled or not. For a network operator it is appealing to maximize A as JP can be applied dynamically for all UEs within this area, which improves their service quality.

We also consider the shape of the covered area. It is relevant for network operators as JP will be deployed mainly in areas with high traffic demand due to economical reasons. Therefore, different architectures might be better suited for certain deployment scenarios due to the shape of their coverage area, e.g., because they match the shape of an asymmetric city.

Architecture parameters In addition to the aspects introduced in this section, each architecture has some specific parameters that only pertain to it. We will introduce these aspects where they are needed.

6.2 Ring infrastructure

In the past, optical metro rings have been deployed to bridge the gap between the core and access network as the reach of access network technologies were limited to a few kilometers. For mobile operators, metro rings have been beneficial as they provide redundant high-speed connectivity for transporting data to and from the core network.

Today, the access part is often a bottleneck and will not be able to provide enough capacity for future wireless technologies like LTE or LTE-Advanced. At the same time, operators would like to reuse existing infrastructure as much as possible due the high deployment costs for new infrastructure. A possible solution to deal with these new requirements is to use PON technology while keeping the optical ring infrastructure as it is nowadays. The resulting architecture is illustrated in Figure 6.2; we refer to it as the *ring architecture* in the following.

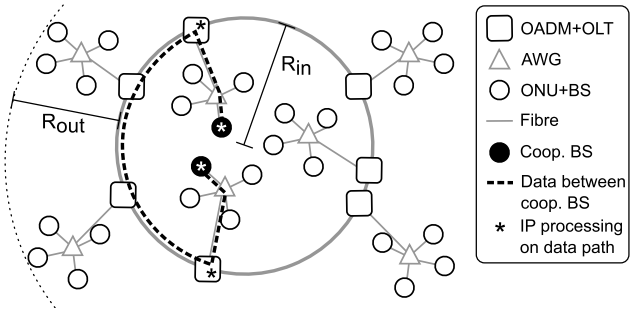


Figure 6.2: Ring architecture and delays that occur

The ring architecture consists of the metro ring and multiple PONs whose OLTs are connected to the ring using Optical Add-Drop Multiplexers (OADMs). The BSs are connected to the ONUs of the PONs. For two BSs that apply JP, data exchange either happens directly within the same PON or via two PONs and the ring in case they are connected to different PONs (example in Figure 6.2).

6.2.1 Ring architecture model

We assume that physical layer tunnels are available on the ring. In this case, IP/MPLS processing is only necessary at the ring ingress and egress OADM and at the ONUs/ BSs that are cooperating. The data will pass transparently through intermediate hops. Furthermore, data from one OADM to another one on the optical ring is always sent via the shortest path, exploiting the bidirectional nature of the ring. This has the effect that the maximum distance that has to be covered is half of the ring's circumference.

In practice, not all BSs outside the ring need to cooperate. Because of their large distance, the benefit of JP is limited, thus making cooperation, at least from the JP point of view, unappealing. Nonetheless, other forms of cooperation, like coordinated scheduling, could be deployed. To address this, we define an outer cooperation factor f_{out} and evaluate values of 0.25 and 0.5. This means that outside the ring, BSs connected to different PONs must be able to cooperate if the OLTs/OADMs of their PONs are located within the same half of the ring (0.25) or all need to cooperate (0.5).

Inside the ring, usually all BSs need to be able to cooperate as BSs/ONUs of different PONs are very close to each other. Analogously to f_{out}, this corresponds to an inner cooperation factor f_{in} of 0.5.

The radius of the optical ring is called R_{in}, which also equals the reach of PONs within the ring. Outside the ring, the PON reach R_{out} will be larger because $f_{\text{out}} < f_{\text{in}}$. Figure 6.2 on the preceding page illustrates R_{in} and R_{out}. To calculate the coverage area of the ring architecture, which is the area of the circle with radius $R_{\text{in}}+R_{\text{out}}$ in Figure 6.2 on the facing page, we first calculate the accumulated IP and wireless processing delay d_{proc} that occurs between two BSs applying JP. It consists of $8 \cdot d_{\text{IP}}$ (4 times in each direction; Figure 6.2 on the preceding page) and of d_{wls}.

$$d_{\text{proc}} = 8 \cdot d_{\text{IP}} + d_{\text{wls}} \tag{6.1}$$

The difference between $d_{\max}$ and d_{proc} remains for the propagation time in the optical fiber. Hence, the maximum possible fiber length s_{fiber} can be calculated according to Equation 6.2.

$$s_{\text{fiber}} = (d_{\max} - d_{\text{proc}}) \cdot \frac{c}{1.45} \tag{6.2}$$

As s_{fiber} is the fiber length for the full round trip, the fiber distance between two cooperating BSs must be below $s_{\text{fiber,BS}} = 0.5 \cdot s_{\text{fiber}}$. This distance must cover two times the PON reach R_{in} and the fraction f_{in} of the optical ring, such that $s_{\text{fiber,BS}} = 2 \cdot R_{\text{in}} + f_{\text{in}} \cdot 2 \cdot \pi \cdot R_{\text{in}}$. Based on this, we calculate R_{in} according to Equation 6.3 and R_{out} according to Equation 6.4.

$$R_{\text{in}} = \frac{s_{\text{fiber,BS}}}{2 \cdot (1 + f_{\text{in}}\pi)} = \frac{s_{\text{fiber}}}{4 \cdot (1 + f_{\text{in}}\pi)} = \\ = \frac{5}{29} c \frac{d_{\max} - 8d_{\text{IP}} - d_{\text{wls}}}{1 + f_{\text{in}}\pi} \tag{6.3}$$

$$R_{\text{out}} = \frac{s_{\text{fiber,BS}} - f_{\text{out}} 2\pi R_{\text{in}}}{2} = \frac{s_{\text{fiber}} - f_{\text{out}} 4\pi R_{\text{in}}}{4} = \\ = \frac{5}{29} c (d_{\max} - 8d_{\text{IP}} - d_{\text{wls}}) \frac{1 + (f_{\text{in}} - f_{\text{out}})\pi}{1 + f_{\text{in}}\pi} \tag{6.4}$$

Based on the two distances R_{in} and R_{out}, the overall coverage area of the ring architecture is calculated according to Equation 6.5. The required number of PONs to cover this area is derived in Equation 6.6.

$$A = \pi \cdot (R_{\text{in}} + R_{\text{out}})^2 \tag{6.5}$$

$$n_{\text{PON}} = \frac{A}{A_{\text{BS}} \cdot \text{sf}_{\text{PON}}} \tag{6.6}$$

6.2.2 Results

Based on the presented model and assumptions, the resulting architecture and its coverage area is illustrated in Figure 6.3 for two typical parameter configurations. The plot is intended to give a first impression on the shape and size of the coverage area and to compare them to other architecture candidates later on.

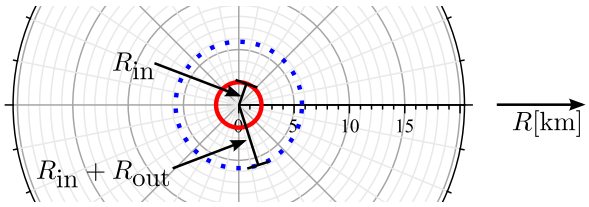


Figure 6.3: Size and shape of ring architecture; $d_{\text{IP}} = 0.1\,\text{ms}$, $f_{\text{out}} = 0.25$. The solid red circle (R_{in}) marks the optical fiber ring, the dashed blue one ($R_{\text{in}} + R_{\text{out}}$) shows the maximum range of the outer PONs.

We also evaluated the ring architecture while varying the parameters d_{IP} and f_{out}. The resulting plot is shown in Figure 6.4 on the next page. The left y axis (black) is scaled in km for R_{in} and R_{out} and the right y axis (gray) is scaled in km^2 for A. Additionally, another y axis shows the number of PONs that are required to cover the corresponding area, implementing the linear relation of Equation 6.6.

The plot shows the expected basic behavior: decreasing d_{IP} and f_{out} leads to a larger coverage area A and hence a higher number of PONs. Looking at the absolute values reveals that with today's IP routing equipment ($d_{\text{IP}} = 0.1\,\text{ms}$) JP is only feasible up to

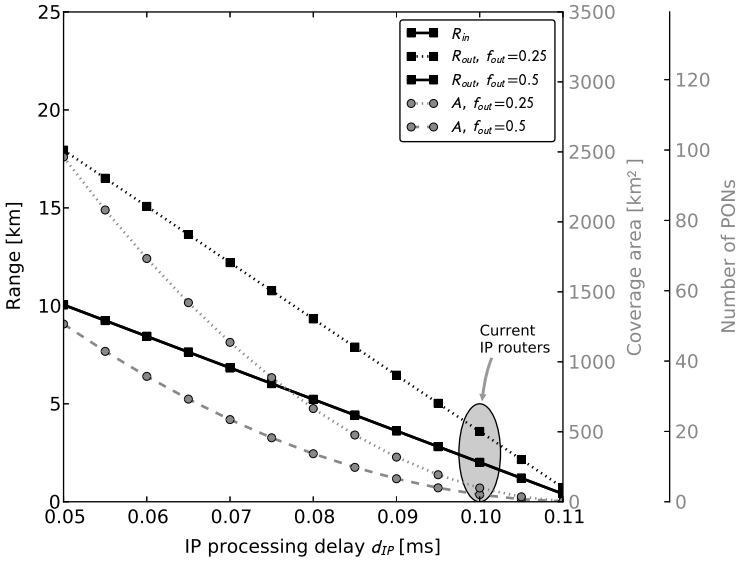


Figure 6.4: Feasible ring architecture size depending on d_{IP} and f_{out}

a ring radius of $R_{\text{in}} \approx 2\,\text{km}$. Depending on f_{out}, the overall area that can be covered by the ring is roughly between $100\,\text{km}^2$ and $130\,\text{km}^2$, which corresponds to 4 to 5 PONs in an urban BS deployment.

The coverage area achieved by this standard ring architecture is not satisfying for a mobile network operator. For example, to cover the area of Berlin (about $900\,\text{km}^2$) with $f_{\text{out}} = 0.25$, 10 ring structures would be required, 12 for New York (about $1200\,\text{km}^2$), and 22 for Tokyo (about $2200\,\text{km}^2$).

6.2.3 Improvements

As the propagation delay in the fiber cannot be reduced easily, the way to have more time for signal propagation is to reduce the processing delay in the network nodes. As the major delay comes from IP/MPLS processing and queuing on the path between cooperating BSs, we propose several approaches to reduce this delay.

The first approach is to reduce the IP/MPLS processing and queuing delay at all nodes (OADM/OLT, ONU/BS), e.g., by improved hardware components. This leads to significant improvements (Figure 6.4) but is more difficult to realize in reality as hardware components need to be exchanged not only at central locations in the network but also in the field.

Another approach is to remove the IP/MPLS processing at the OADMs/OLTs. This can be achieved, e.g., by bypassing the traffic between cooperating BSs on the optical layer such that there is no delay caused by IP/MPLS processing. Then, Equation 6.1 changes to Equation 6.7, i.e., the total IP/MPLS delay is halved.

$$d_{\text{proc}} = 4 \cdot d_{\text{IP}} + d_{\text{wls}} \tag{6.7}$$

The same effect occurs if IP/MPLS processing is removed at the ONU/BS locations. This, however, has the same disadvantage as the described reduction of the IP/MPLS processing delay in general: equipment needs to be exchanged in the field. Furthermore, no IP/MPLS processing at the ONU complicates reusing the same PON infrastructure also for, e.g., Fiber To The Home (FTTH), and attaching multiple BSs to a single ONU. This is a clear disadvantage of removing IP/MPLS processing at the ONU locations.

Furthermore, the queuing delay can be reduced by increasing the link speeds. This is difficult in the access part of the network, i.e., in the PONs, but feasible on the links that form the metro ring.

The effects of the proposed approaches are summarized in Figure 6.5. The plot contains the case with IP processing at all locations ($\mathrm{IP_{OLT}}$, $\mathrm{IP_{ONU}}$), only at the ONUs ($\mathrm{IP_{ONU}}$), and without any IP processing (no IP).

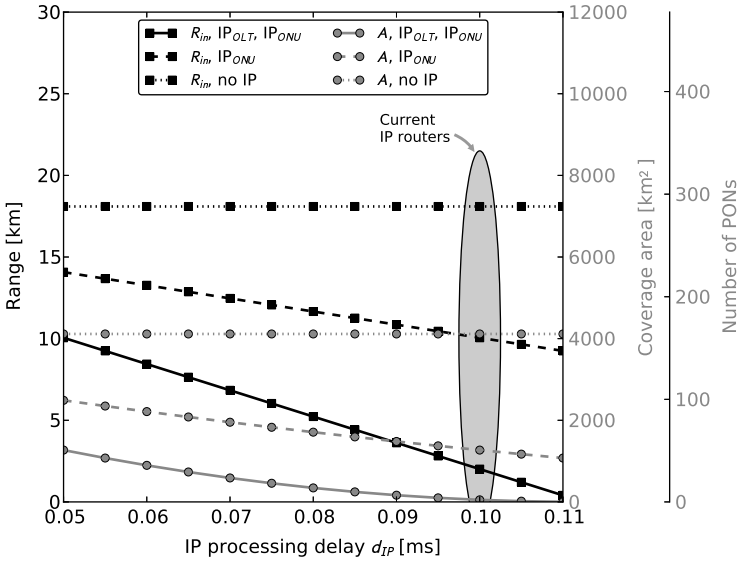


Figure 6.5: Feasible ring architecture size depending on d_{IP}, assuming proposed IP processing improvements at OLT and ONU; $f_{\mathrm{out}} = 0.5$

Removing IP processing at the OADMs/OLTs leads to an increased ring coverage area by a factor of about 25. This means that with today's IP router equipment at the ONU/BS locations, an area of roughly 1270 km^2 can be covered with 50 PONs.

From the coverage perspective, this improvement is a big step. Unfortunately, equipment for implementing this at the OADMs/OLTs is not available yet. Current OADMs support only a limited number of optical channels (on the order of 100) on the ring. This is not enough for having separate channels for each pair of cooperating BSs.

6.3 Star infrastructure

In the past, metro networks have been deployed to cover the remaining distance between the core network and the customer as the reach of access network technologies were

limited to a few kilometers. Using WDM PONs, which are able to provide a long reach and capacities that are high enough to support JP, implementing a single network covering the old metro and access part is straightforward. The resulting architecture looks like a star and is illustrated in Figure 6.6.

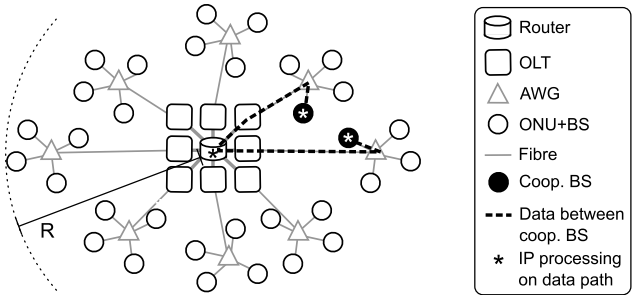


Figure 6.6: Star topology and delays that occur

As the OLTs are now all co-located in a single location, there is no need for a separate router for each of them as it was the case in the metro ring architecture in Section 6.2. All OLTs can share a single, big router which quickens IP/MPLS processing and queuing.

6.3.1 Star architecture model

All OLTs and the IP/MPLS router are co-located. This means that there is no optical propagation delay between the OLTs and the router.

The PON reach, i.e., the radius of the area that is covered by the star architecture is called R (Figure 6.6). Calculating the coverage area of the star architecture, which is the area of the circle with radius R in Figure 6.6, is similar to the calculation procedure in the ring architecture. Here, the accumulated processing delay just consists of $6 \cdot d_{\text{IP}}$ and of d_{wls}. The reduced IP/MPLS processing is a result of the single, centralized IP/MPLS router (cf. Figure 6.6) at the OLTs.

$$d_{\text{proc}} = 6 \cdot d_{\text{IP}} + d_{\text{wls}} \tag{6.8}$$

The resulting maximum possible fiber length s_{fiber} is calculated exactly as already described in Equation 6.2.

As for the ring architecture, s_{fiber} is the fiber length for the full round trip. Similarly, the fiber distance between two cooperating BSs must be below $s_{\text{fiber,BS}} = 0.5 \cdot s_{\text{fiber}}$ again. This distance must cover two times the PON reach R in the star architecture. Consequently, R can be calculated according to Equation 6.9.

$$\begin{aligned} R &= \frac{s_{\text{fiber,BS}}}{2} = \frac{s_{\text{fiber}}}{4} = \\ &= \frac{5}{29} c(d_{\max} - 6d_{\text{IP}} - d_{\text{wls}}) \end{aligned} \tag{6.9}$$

The overall coverage area of the star architecture is calculated according to Equation 6.10. The number of required PONs are derived according to Equation 6.6.

$$A = \pi R^2 \tag{6.10}$$

6.3.2 Results

We did the same evaluation for the star architecture that we have done for the ring architecture. Now, however, the results only depend on the parameter d_{IP}.

Based on the model and assumptions, the star architecture's coverage area is illustrated in Figure 6.7 for the typical IP/MPLS and queuing delay. Compared to the legacy metro ring architecture in Figure 6.3 on page 114, a clear increase in the coverage area is visible. The coverage area depending on the IP/MPLS delay d_{IP} is shown in Figure 6.8 on the facing page ($\mathrm{IP}_{\mathrm{OLT}}$, $\mathrm{IP}_{\mathrm{ONU}}$).

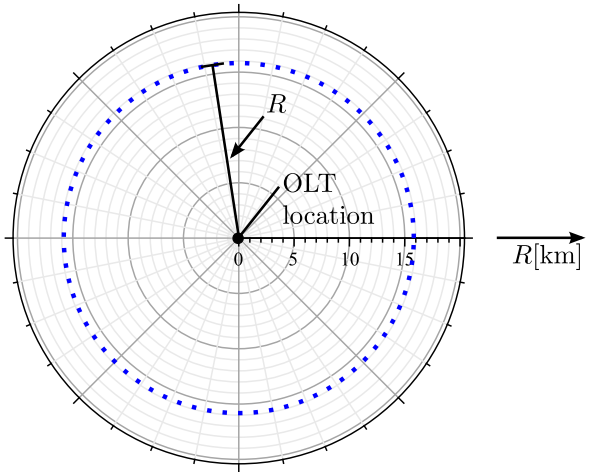


Figure 6.7: Size and shape of star topology; $d_{\mathrm{IP}} = 0.1\,\mathrm{ms}$. All ONUs are co-located at the origin. The dashed blue circle (R) shows the maximum range of the PONs.

With today's IP/MPLS router equipment, the star architecture can cover an area of approximately $750\,\mathrm{km}^2$ ($R \approx 15.5\,\mathrm{km}$) with 30 PONs. This advantage compared to the ring architecture ($130\,\mathrm{km}^2$) with current hardware components is a result of the colocation of all OLTs. The required router that interconnects all OLTs is no problem; products are available today.

6.3.3 Improvements

For the star architecture, the same improvements can be considered that have been proposed for the ring architecture. Besides reducing the IP/MPLS processing delay in general or removing it at the ONUs/BSs, which both have the same serious disadvantages as in the ring architecture (cf. Section 6.2.3), it is easier to remove the IP/MPLS

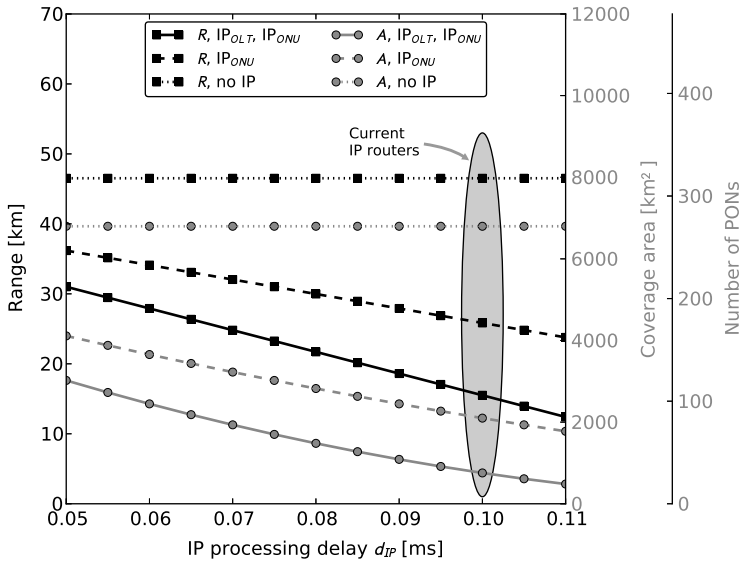


Figure 6.8: Feasible star architecture size depending on d_{IP}.

processing at a small amount of central locations. As all OLTs and the IP/MPLS router are co-located in a single data center, modifying, e.g., all OLTs can be done with low effort. The resulting coverage area using such a bypassing technique at the OLTs is shown in Figure 6.8 ($\mathrm{IP_{ONU}}$). The plot also contains the case without any IP/MPLS processing in the network (no IP) for comparison.

The plot shows that bypassing the IP/MPLS router in the OLT data center for traffic between cooperating BSs leads to significant improvement of the architecture coverage area. The area that can be covered increases from 750 km^2 to 2100 km^2; an improvement by a factor of 2.8.

6.4 Merging ring and star concept

The distributed ring architecture with OADMs on the metro ring to which PONs are connected is not a solution for the future. Without improving hardware components and without extensive link capacity over-provisioning, requirements of JP cannot be fulfilled such that it can be applied on a metro-wide scale.

On the other hand, the star architecture with centralized OLTs and a single IP/MPLS router co-located with the OLTs provides a very good performance already with today's hardware. However, fiber deployment costs are between 15,000 and 17,000 USD per kilometer on average [54]. These costs make it very unlikely to completely replace existing infrastructure like available optical metro rings. They need to be reused as much as possible.

In line with that, we propose an architecture that combines the idea of colocating all OLTs and using a single, big router (star architecture) with existing ring fiber (ring architecture). This reduces the required time for IP/MPLS processing and queuing at

intermediate nodes, which can be spent for propagation within the fiber to extend the coverage area. Existing pipes and fiber of metro rings are used to connect the centralized OLTs to the AWGs. We call this the *star-via-ring architecture* (Figure 6.9).

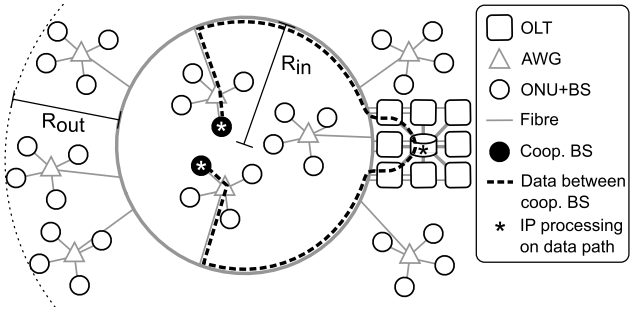


Figure 6.9: Star-via-ring topology and delays that occur. Multiple parallel PON fibers on the ring are drawn as one thick line.

Unlike the previous two architectures, this one is not symmetric. The location of the co-located OLTs on the ring plays an important role for the coverage area, as will be shown in the next section.

6.4.1 Star-via-ring architecture model

The accumulated IP/MPLS and wireless processing delay d_{proc} that occurs between two BSs applying JP is identical to the star architecture. It consists of $6 \cdot d_{\text{IP}}$ (3 times in each direction, cf. Figure 6.9) and of d_{wls}.

The difference between $d_{\max}$ and d_{proc} again remains for propagation in the fiber and leads to the maximum possible fiber length s_{fiber}. Its calculation is identical to the previous architectures (Equation 6.2). The fiber length between two BSs must be below $s_{\text{fiber,BS}} = 0.5 \cdot s_{\text{fiber}}$, as s_{fiber} is the full round-trip fiber length.

As for the star architecture, we assume that all BSs should be able to cooperate in the star-via-ring architecture. Starting from this assumption, we can calculate the maximum possible radius of the fiber ring R_{in}. The distance $s_{\text{fiber,BS}}$ needs to cover two times the inner PON reach R_{in} and two times half of the optical ring, such that $s_{\text{fiber,BS}} = 2 \cdot R_{\text{in}} + 2 \cdot \pi \cdot R_{\text{in}}$. This is the worst case when two BSs cooperate that are attached to PONs whose fiber leaves the ring at the opposite of the OLTs' location (left side in Figure 6.9). Hence, we calculate R_{in} according to Equation 6.11.

$$\begin{aligned} R_{\text{in}} &= \frac{s_{\text{fiber,BS}}}{2 \cdot (1+\pi)} = \frac{s_{\text{fiber}}}{4 \cdot (1+\pi)} \\ &= \frac{5}{29} c \frac{d_{\max} - 6d_{\text{IP}} - d_{\text{wls}}}{1+\pi} \end{aligned} \tag{6.11}$$

The location where a PON fiber leaves the ring plays an important role. The worst case is opposite the OLTs. Here, half of the ring has to be traversed and the remaining

part, which is equal to R_{in}, can be used to depart to the outside of the ring. When moving the point where the fiber leaves the ring closer to the co-located OLTs, a larger part of the available fiber distance can be spent to depart from the ring. In Figure 6.9 on the preceding page, this would result in a larger R_{out} on the right side compared to the left side, as the OLTs are co-located on the right side. Hence, the outer radius depends on α, which is the angle between the location of the co-located OLTs, the center of the ring, and the location where the PON fiber leaves the ring. We measure this angle in both directions from the co-located OLTs, i.e., $0 \leq \alpha \leq \pi$. It is illustrated in Figure 6.10 on the following page.

To calculate the outer radius $R_{\text{out}}(\alpha)$, we look at the worst case where the cooperating BS A's PON leaves the ring at the opposite of the co-located OLTs and a second BS B's PON leaves at angle α. The total available fiber length $s_{\text{fiber,BS}}$ needs to cover half the ring and R_{in} (from the OLT to BS A) and the fraction of the ring depending on α and $R_{\text{out}}(\alpha)$ (from the OLT to BS B). This is contained in Equation 6.12 and leads to $R_{\text{out}}(\alpha)$ in Equation 6.13.

$$s_{\text{fiber,BS}} = R_{\text{in}} + \frac{1}{2}2\pi R_{\text{in}} + \frac{\alpha}{2\pi}2\pi R_{\text{in}} + R_{\text{out}}(\alpha) \tag{6.12}$$

$$\begin{aligned} R_{\text{out}}(\alpha) &= s_{\text{fiber,BS}} - R_{\text{in}} \cdot (1 + \alpha + \pi) = \\ &= \frac{s_{\text{fiber}}}{2} - R_{\text{in}} \cdot (1 + \alpha + \pi) = \\ &= \frac{5}{29}c(d_{\max} - 6d_{\text{IP}} - d_{\text{wls}})\frac{1 + \pi - \alpha}{1 + \pi} \end{aligned} \tag{6.13}$$

Based on the two distances R_{in} and $R_{\text{out}}(\alpha)$, the overall coverage area of the star-via-ring architecture is calculated according to Equation 6.14 using polar integration.

$$\begin{aligned} A &= \int_0^{\pi} (R_{\text{in}} + R_{\text{out}}(\alpha))^2 \, \mathrm{d}\alpha = \\ &= \frac{25}{2523}c^2(6d_{\text{IP}} + d_{\text{wls}} - d_{\max})^2 \cdot \frac{\pi(12 + 6\pi + \pi^2)}{(1 + \pi)^2} \end{aligned} \tag{6.14}$$

In case an existing ring deployment has a radius R_{in} that is larger than the maximum supported radius, as calculated in Equation (6.11), it is not possible anymore that all BSs within the ring can cooperate (i.e., $f_{\text{in}} < 0.5$). This drawback cannot be addressed by deploying multiple OLT hotels on the ring such that their individual reach on the ring is overlapping, and such that BSs can be assigned to the best OLT hotel for the desired cooperation cluster. This approach does not work as the PON trees are passive, i.e., when the fiber leaves the ring towards the AWG, no forwarding decision can be made. Hence, individual BSs cannot be assigned dynamically to different OLT hotels. Instead, BSs could be multi-homed by assigning them to multiple PONs at the same time, e.g., by using multiple different wavelengths that come from different OLT hotels and terminate in the same ONU.

6.4.2 Results

The evaluation of the star-via-ring architecture is similar to the ring and star architecture. Based on the model and assumptions, the star-via-ring architecture's coverage area is shown in Figure 6.10 for the typical IP/MPLS and queuing delay of $d_{\mathrm{IP}} = 0.1\,\mathrm{ms}$.

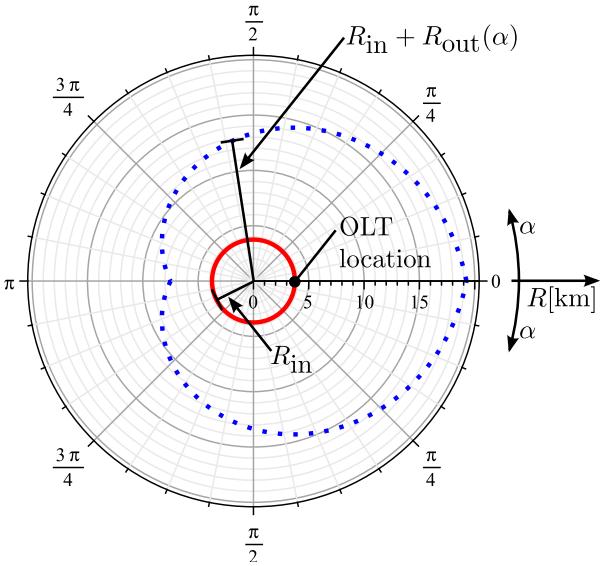


Figure 6.10: Size and shape of star-via-ring topology; OLTs are all co-located on the right side of the ring, $d_{\mathrm{IP}} = 0.1\,\mathrm{ms}$. The solid circle marks the optical fiber ring, the dashed one shows the maximum range of the outer PONs.

The most noticeable change in the coverage area is the asymmetry compared to the fully symmetric, circle-shaped coverage areas of the ring and star architectures. The coverage area is heart-shaped now, which is caused by the co-location of all OLTs in one location on the fiber ring (at $\alpha = 0$ in Figure 6.10). The coverage area depending on the IP/MPLS delay d_{IP} is shown in Figure 6.11 on the next page ($\mathrm{IP}_{\mathrm{OLT}}$, $\mathrm{IP}_{\mathrm{ONU}}$).

Assuming today's IP/MPLS router equipment, the star-via-ring architecture is able to cover an area of approximately $600\,\mathrm{km}^2$ ($R_{\mathrm{in}} \approx 3.7\,\mathrm{km}$, $3.7\,\mathrm{km} \leq R_{\mathrm{out}} \leq 15.5\,\mathrm{km}$) with 24 PONs. This advantage compared to the ring architecture ($130\,\mathrm{km}^2$) is, as for the star architecture case, a result of colocating all OLTs and using a single IP/MPLS router. Compared to the star architecture ($755\,\mathrm{km}^2$), which is optimal in terms of coverage area as the fiber can be deployed without indirections, the star-via-ring architecture looses due to the fiber's "forced detour" via the ring. Nevertheless, it achieves 79 % of the star's coverage area and 607 % of the distributed ring architecture while reusing existing fiber. The coverage areas of the three architectures are summarized in Figure 6.12 on the facing page. For better comparability, the shape of the star-via-ring architecture has been aligned according to both architectures' optical fiber rings (Figure 6.12a on the next page) and to the OLT locations (Figure 6.12b on the facing page). In both situations, the star-via-ring approach covers the full area of the legacy ring architecture.

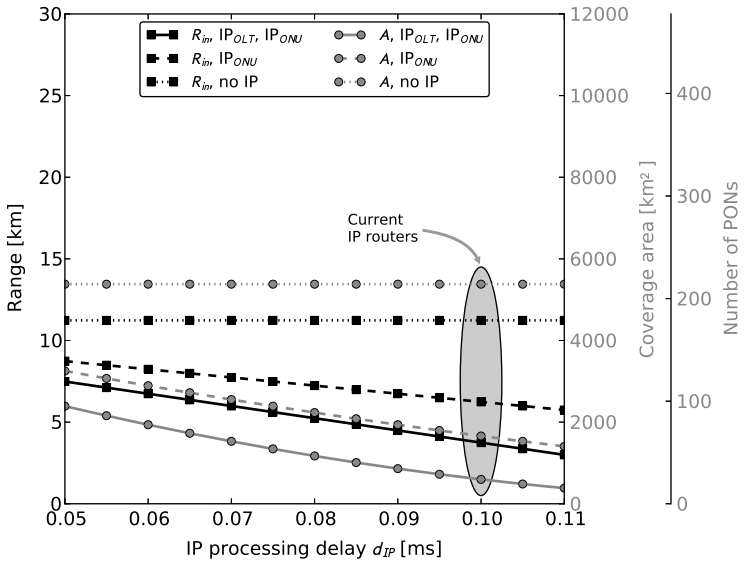
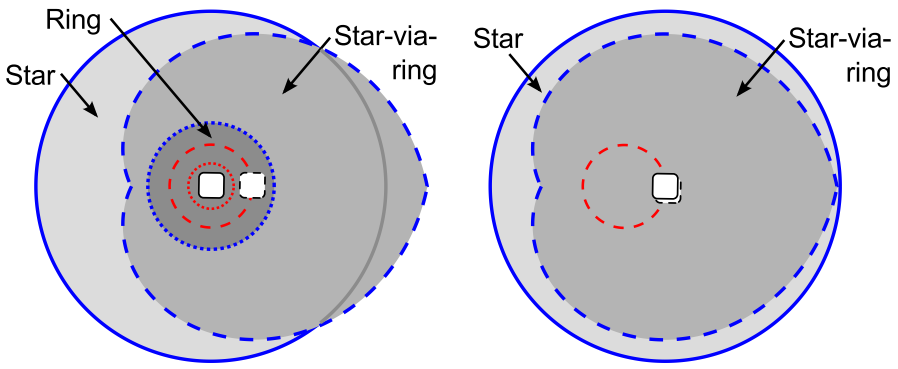


Figure 6.11: Feasible star-via-ring architecture size depending on d_{IP}.



(a) Centered around optical fiber rings' midpoint.

(b) Centered around co-located ONUs' location.

Figure 6.12: Coverage of ring (dotted blue), star (solid blue), and star-via-ring architecture (dashed blue); $d_{\mathrm{IP}} = 0.1\,\mathrm{ms}$, $f_{\mathrm{out}} = 0.25$. Fiber rings (ring and star-via-ring architecture) are drawn in the same line styles in red color. Co-located ONUs (star and star-via-ring architecture) are marked by the ONU symbol in the same line style.

6.4.3 Improvements

Using improvements already discussed for the star architecture (reducing IP/MPLS processing and queuing delay in general, at the ONUs/BSs, or at the OLTs) is possible for the star-via-ring architecture, too. The effects have been summarized in Figure 6.11.

6.5 Summary

Addressing the backhaul requirements of future mobile access systems is not simply a matter of upgrading the networking hardware. Based on the results, the support of demanding CoMP mechanisms like JP goes beyond the adoption of high-capacity access transport technologies like WDM PONs.

We have shown that latency is the major concern for inter-BS cooperation. On metro scale, 1 ms delay for BS-to-BS data exchange is easily exceeded. Considering this requirement, a first conclusion is that JP becomes feasible as soon as queuing and IP processing delay between cooperating BSs is limited, e.g., by restricting the scope in which JP is used. In small deployment scenarios, like public hot-spots and buildings, cooperation is limited to BSs connected to the same PON infrastructure, thus not traversing the routers present in higher layers of the wireline backhaul network hierarchy.

Deploying JP for macro cells on a metro-wide scale requires to take into account a major part of the wireline backhaul network infrastructure. Our latency analysis has shown that reusing today's ring infrastructure is feasible only with improved system components and optical bypassing techniques, which both are not available yet. The star architecture, on the other hand, allows to use JP in areas up to 2100 km^2, e.g., to cover Tokyo, even with today's technology. The drawback is that major parts of an existing backhaul network infrastructure cannot be reused. In between, the proposed star-via-ring architecture offers a reasonable trade-off between coverage area and reuse of existing ring fiber infrastructure.

For network planning, not only the area that a certain architecture can cover is important. What plays a crucial role is also the amount of population that a certain architecture can reach, not to waste money for upgrading infrastructure in areas from where no revenues can be expected. Hence, it not only matters *where* people live but also *how* people are distributed in a certain territory. For example, if the distribution of the population is non-uniform, then the asymmetry of the star-via-ring architecture's coverage shape can be exploited to cover the most densely populated areas at reasonable CAPEX costs. On the other hand, the ring architecture is appealing in uniformly populated, small to medium-sized cities with an area of less than 100 km^2. As an example, Figure 6.13 on the facing page overlays the population density distribution of Berlin's metro area[1] to the coverage areas of the three architectures. The star-via-ring architecture is the most appealing solution as it covers most population while reusing an existing fiber ring. Furthermore, it can be rotated to cover the desired area.

[1] The population density map was taken from `http://www.stadtentwicklung.berlin.de/planen/basisdaten_stadtentwicklung/atlas/de/bevoelkerung.shtml`.

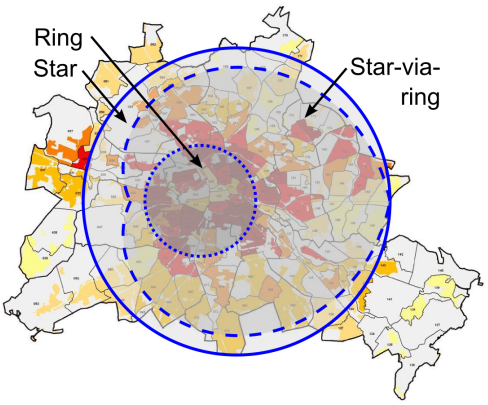


Figure 6.13: Population in Berlin's metro area (darker color indicates higher density) with overlaid architecture coverage areas (all drawn in the same scale).

Chapter 7

Combining clustering and feasibility improvement techniques

Contents

In previous chapters, mechanisms to deal with backhaul network limitations (Chapter 3 and 4) and to improve CoMP feasibility (Chapter 5) have been proposed. In this chapter, these mechanisms are set into relation to form an overall system architecture that combines the advantages of the individual approaches. The single CoMP cluster case is discussed in Section 7.1 and the overall system to deal with multiple clusters in an overall system is addressed in Section 7.2.

7.1 Single CoMP cluster

The individual solutions proposed in the previous chapters all have different complexity, require different information as input parameters, and influence the overall network in various ways. If the techniques are applied only individually, i.e., only one of the proposed techniques is used in the mobile network, the consequences can be foreseen as no interdependency between multiple techniques comes into play. Using multiple techniques at the same time in the network can lead to unexpected effects due to their interplay. Furthermore, it is unclear in which order to apply the proposed techniques to optimize the system, like avoiding unnecessary overhead and reducing costs.

In this section, I consider a single desired wireless CoMP cluster that can be used to serve a single UE or a group of UEs using MU-MIMO. The interrelation between multiple clusters in a system will be discussed in the next section.

Figure 7.1 gives an overview of a system architecture comprising all individual components that have been proposed in the course of this thesis. This architecture has been developed with the goal of exploiting network capabilities as much as possible, and at the same time reducing overhead for exploiting additional features as much as possible. The reasons for positioning the individual components as proposed, their relation to each other, and alternatives to the proposals will be discussed in the following.

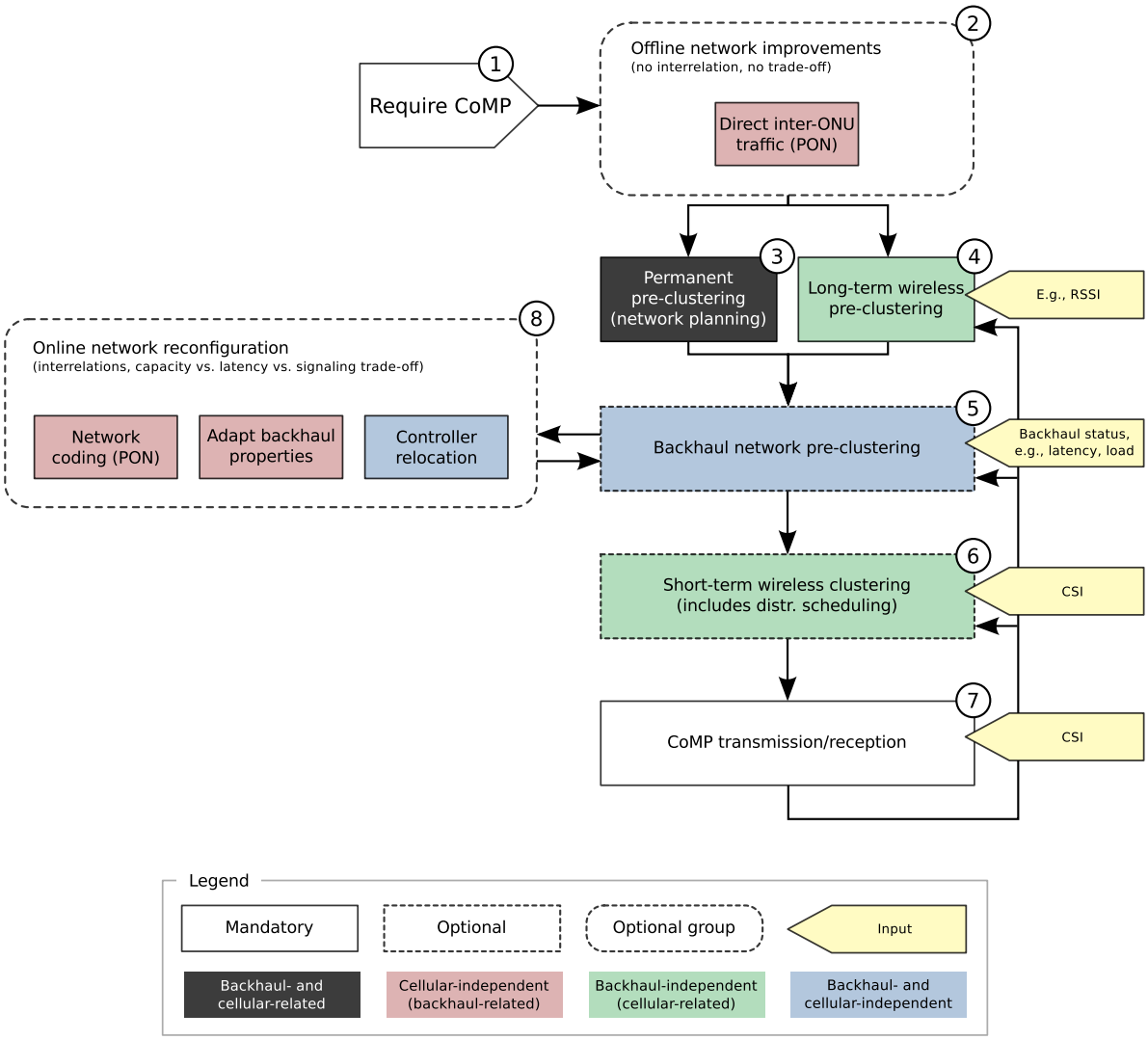


Figure 7.1: System architecture combining the individual components introduced in the previous chapters. The border style of the boxes indicates whether the step is mandatory or optional. The fill style shows the domain in which the step is operating.

Block 1 (Require CoMP) This block is the starting point of the diagram. It indicates that the necessity to use CoMP for certain UEs in the cellular network has been detected. This necessity can be discovered online, i.e., while the system is already operat-

ing, e.g., because the high interference level for a cell-edge UE prevents high throughput. Another point in time of discovering the necessity for CoMP is during the network planning phase when it is clear that conventional non-cooperative transmission/reception is not sufficient to fulfill UE demands.

Block 2 (Offline network improvements) Improving PONs with optical inter-ONU data exchange (Section 5.2.2) is possible whenever additional capacity and lower latency is needed compared to the conventional PON, e.g., because it is known that CoMP techniques shall be used in the cellular network. As this technique requires to deploy new hardware components in the field, I call this group *offline network improvements*. Such techniques cannot be activated on the fly if they have not been installed in advance. After required hardware has been deployed, the inter-ONU transfer can be activated and deactivated on demand using the network reconfiguration mechanisms (to save energy at the ONUs).

The advantage of the inter-ONU data exchange is that, besides minor additional costs for the PON hardware, there is no trade-off, which, e.g., results in increased latency due to the increased capacity. Hence, these techniques can be deployed without interacting with other techniques.

Alternatively, instead of deploying new hardware in the whole network, it might make sense to limit it to selected areas where it is required most. This is possible by deploying the offline network improvements after executing the step that determines the desired wireless clusters (Block (3) or (4)). Then, the backhaul network that interconnects the BSs in the desired clusters is upgraded first.

Block 3 (Permanent pre-clustering) In the beginning of a CoMP transmission/reception, BS clusters that should cooperate need to be selected. The simplest form of defining such desired cooperation clusters is to define them in a kind of network planning step. In this case, clusters are statically selected, i.e., they do not change over time (Chapter 3). There are many possible criteria based on which they can be selected, like wireless measurements in a running non-cooperative cellular system that reveal weaknesses in the service quality that can be solved by applying CoMP.

In case permanent pre-clustering is used, there are no individual clusters for certain (groups of) UEs that need CoMP. UEs have to be assigned to the best-fitting available pre-defined cluster. This may lead to a non-optimal service for the UEs but is simpler to implement. During network operation, much less coordination and signaling is needed. This Block (3) is an alternative to Block (4), described next.

Block 4 (Long-term wireless pre-clustering) Instead of pre-defining permanent clusters, as it is done in Block (3), individual clusters are chosen for (groups of) UEs in this step (Section 4.3.1). These desired clusters are selected based on, e.g., wireless measurements of the wireless channel qualities between the BSs and the UEs. Such measurements are already part of the LTE standard for hand-over purposes and are

repeated every 10 ms. As the channel qualities change over time, the desired clusters change, too. They have to be recalculated in regular intervals.

The reason why this step is done first is that the required input parameters, which provide information about the long-term wireless channel quality, are anyway available at the BSs. These parameters are needed for the conventional hand-over procedure. Hence, this information can be reused "for free" to decide which BSs are suitable to cooperatively serve a certain (group of) UEs. The output of Block (4) are the reasonable desired clusters from the wireless point of view. Based on this information, other blocks in the architecture try to make these desired clusters possible or further shrink the clusters in case the backhaul network does not support them.

Instead of returning just one desired cluster, Block (4) can also return multiple desired clusters, each annotated with a priority. If the cluster with highest priority cannot be implemented due to backhaul network limitations, others with slightly lower performance gain might be feasible. This way, the backhaul network clustering in Block (5) checks the desired clusters for their feasibility in descending priority order. The best feasible cluster is implemented eventually.

Depending on the strategy of the network operator, at least one of the two Blocks (3) and (4) should be implemented. In case of going for Block (3), the system is less complex but also less performant because cooperating BSs are not selected according to the actual wireless channel conditions. This, however, does not mean that backhaul network requirements are relaxed. In case Block (4) is implemented, the requirements in terms of coordination and signaling are higher but also higher gains can be expected. It is also possible to implement both approaches such that they complement each other. Then, in case dynamic clustering in Block (4) cannot be applied for some reason, the system falls back to the static pre-planned clusters.

Leaving out both Block (3) and (4) is theoretically possible. Then, desired clusters are determined just based on CSI, which will be collected before the cooperative transmission in Block (6). The consequence is that the backhaul clustering step in Block (5) has no information on desired clusters. Consequently, the determined feasible cluster might not lead to the desired performance gain. This lack could be circumvented by moving up Block (6) such that information on desired wireless clusters is known when checking the backhaul network for its capabilities. Doing so, however, puts much stricter delay requirements on Block (5) and Block (8), as the measured CSI can expire quickly.

Block 5 (Backhaul network pre-clustering) This step contains the pre-clustering algorithm that checks which BSs in a desired cluster (determined in Block (3) or (4)) actually *can* cooperate from the backhaul network point of view (Section 4.3.2). This is done to reduce overhead for transferring UE and signaling data via the backhaul network to BSs that cannot participate in CoMP eventually. This step requires backhaul network information like link load and link latencies as input. Such information can easily be made available via the network's control plane.

In case Block (3) has been implemented instead of Block (4), Block (5) can be skipped if the backhaul network infrastructure has been planned and dimensioned such that it is

able to support all desired CoMP clusters. If the pre-clustering from Block (4) is used, it is not really possible to predict all desired CoMP clusters that may result from arbitrary UE positions, movements, etc. The only way to avoid implementing Block (5) then is to do costly over-provisioning in the backhaul network. This, however, still does not solve latency problems due to propagation in the backhaul network. This means that Block (5) is required unless over-provisioning guarantees low delay, which is not realistic.

After executing the pre-clustering algorithm in Block (5), a *feasible* cluster, which is a subset of the *desired* cluster, is known. In addition, in case some BSs of the desired cluster cannot participate in the CoMP transmission/reception due to backhaul network limitations, the exact reason for this is known, e.g., missing capacity on a certain link (Section 5.1.2.1). Such information is used in Block (8), which will be described later.

As already mentioned for Block (4), there can be multiple desired clusters for each UE group to be served cooperatively. In this case, the backhaul network clustering will determine the best feasible cluster from the given set of desired clusters.

Block 6 (Short-term wireless pre-clustering) After finding the feasible cluster from the backhaul network perspective in Block (5), the final cluster, which is used for the CoMP transmission/reception, can be further limited to avoid unnecessary overhead. As so far only averaged long-term wireless channel quality information has been used in Block (4) to get a first idea on a reasonable cluster and the CSI which CoMP bases on is only valid for a very short time (on the order of 1 ms), the final decision needs to be made based on current CSI (assuming that a kind of JT or JB CoMP scheme is used). The CSI is measured by the UEs only for the BSs in the feasible cluster and sent to the corresponding BSs. Based on this up-to-date channel information, the BSs for a single CoMP transmission/reception operation are selected, which are again a subset of the feasible cluster determined in Block (5). This also involves a scheduling step to find a (free) common Physical Resource Block (PRB) among the BSs in which the CoMP operation takes place.

The time interval at which measurements and clustering are repeated, i.e., Block (6) is re-executed, depends on the actual fluctuation of the wireless channel properties. In high mobility scenarios, this needs to be done up to every 1 ms; in low-mobility cases several milliseconds will suffice. In case the update interval is too low, wireless performance decreases and the CoMP gain diminishes [41].

Block (6) is required only if the used CoMP technique requires very detailed and up-to-date wireless channel state information. This is the case, e.g., for JT or JB. Simpler techniques, like JS, do not require CSI. Long-term measurements, as anyway collected for hand-over decisions and used in Block (4), are sufficient.

Block 7 (CoMP transmission/reception) In this block, the actual CoMP transmission/reception takes place. This means that based on the current CSI, which can be reused from Block (6), and the finally selected cooperation cluster, precoding information is calculated and distributed to the BSs in the cluster together with the UE data.

UE data needs to be exchanged in case of JT only, otherwise the precoding and/or scheduling information is sufficient.

After the sending operation, different next steps are possible, indicated by the arrows going back to one of the three clustering steps. In case there is more data to be sent, a new short-term wireless clustering step is required if the previously collected CSI is outdated meanwhile. The validity can be estimated by comparing previsously measured CSI. According to this estimation, new short-term wireless clustering steps are executed when necessary (at maximum for each scheduling interval, i.e., 1 ms in LTE).

As the backhaul network conditions also change over time, it is necessary to do the backhaul network clustering step in Block (5) again, too. The backhaul latency conditions usually do not change as quickly as the wireless channel conditions. The load, however, can change at the same speed as the wireless channel conditions change, depending on the applied buffering in the network. Hence, the frequency of re-executing Block (5) is between those of Block (4) and Block (6). The long-term wireless pre-clustering in Block (4) has to be repeated, too. The decisions in this block are based on the regular channel measurements that are included in current and future standards for hand-over purpose. Such channel measurements are done regularly in intervals of 10 ms, which suggests to do the long-term wireless pre-clustering in the same time intervals. Following that, the interval of re-executing the backhaul network pre-clustering step in Block (5) is between 1 ms and 10 ms, which are the re-execution intervals of the two surrounding clustering steps in Blocks (4) and (6).

Block 8 (Online network reconfiguration) In case the desired clusters from Block (4) are found to be infeasible in Block (5), there are several possibilities of reconfiguring the network online, i.e., while it is operating, to improve CoMP feasibility. As mentioned earlier, the algorithm doing the backhaul network pre-clustering in Block (5) also outputs the reasons why a certain BS is not able to join a cluster. This information is very valuable as many backhaul network technologies provide flexibilities that can be exploited to overcome the limitations that prohibit the BSs to join the cluster. Therefore, Block (5) and Block (8) can interact by applying reconfiguration steps until the desired cluster is feasible or the gap is narrowed. Such mechanisms are CoMP Controller Reassignment (CCR) (Section 5.1.1), generic backhaul network reconfiguration (Section 5.1.2), as well as network coding in PONs (Section 5.2.1). All these techniques are grouped in Block (8) and all introduce a trade-off between network capacity, link latency, and signaling overhead. This is the reason why it is not desirable to activate them continuously, like the mechanisms in Block (2), but only when required.

The individual reconfiguration techniques in Block (8) all influence the backhaul properties and hence the feasibility of the CoMP cluster, which causes inter-dependencies among the different online reconfiguration techniques. Hence, after applying one technique, another pre-clustering step needs to be executed to see the effect before applying an additional technique, e.g., to further improve cluster feasibility. Otherwise, reconfigurations may have undesired effects, neutralize each other, or just produce additional overhead without improving CoMP feasibility. For example, provisioning additional

wavelengths for an overloaded ONU in a PON, which connects a BS that controls a CoMP transmission, and at the same time shifting load from this BS to another BS by doing a forced hand-over at the same time does not make sense.

Concerning the order in which the reconfiguration techniques are applied, it makes sense to start with those that do not limit other existing and future clusters. Examples for this are to enable direct inter-ONU transfers in PONs or provisioning additional capacity/wavelengths on demand.

In a second step, those mechanisms should be applied that re-allocate *unused* resources in the backhaul network. This happens when some of the backhaul network properties are changed, like adapting the DBA in TDM PONs or enabling the CoMP Controller Reassignment (CCR) mechanism. These mechanisms do not limit other *existing* clusters but may limit the feasibility of additional clusters in the *future*.

Finally, all mechanisms that influence both existing *and* future clusters shall be applied. This is the case, e.g, for the NC approach in PONs. Here, not only the new cluster has to be checked for its feasibility but also all existing clusters that might become infeasible due to the changed backhaul network properties.

7.2 Extending to multiple CoMP clusters

In a cellular system, CoMP is used to improve the UEs' service quality. Hence, it is likely that CoMP is required for multiple UEs at the same time, which results in multiple (possibly overlapping) CoMP clusters that have to be operated in parallel. For these multiple clusters, it is desirable to adapt the CoMP activities to each other to achieve an optimal overall system performance.

Section 7.1 discussed the relation between different CoMP techniques for a *single* CoMP cluster. In the general case, where multiple CoMP clusters are used in the system in parallel, the same mechanisms can be used for each cluster individually. This means that decisions are made consecutively, without regard to interaction with other CoMP activities in the system. Hence, actions are not adapted to each other to achieve a desired overall system behavior. One simple possibility to achieve a certain degree of global optimization is to change the order in which multiple clusters are processed. This way, certain clusters can get a higher priority than others. However, more sophisticated mechanisms, like the following examples, are difficult to achieve:

- Guarantee global fairness among UEs by adapting existing clusters, e.g., shrinking, when new UEs join the system.
- Choosing CoMP clusters that optimize the overall system performance, like wireless throughput, instead of only optimizing per cluster.

To allow such global optimizations, multiple CoMP activities in the system have to be aligned more closely than just giving them different priorities. This can be achieved in different ways. According to the CoMP architecture discussions earlier in this thesis, two solution approaches are possible. In case the CoMP controllers are implemented in

a distributed manner, they need to coordinate their actions and need to exchange a lot of information for this. If the controllers are implemented in a more centralized way, i.e., there is a big CoMP controller that is responsible for many BS, this controller already has aggregated knowledge for a large area and can perform inter-cluster optimizations within this area. Optimizations outside this area require a distributed approach again, as described before. Implementing global knowledge and eventually solving the resulting optimization problems for the multiple cluster case is not addressed in this thesis.

7.3 Summary

This chapter has presented and discussed a full architecture that contains all techniques that have been developed in the previous chapters. The individual mechanisms have been set into relation and have been connected such that the resulting system is optimized by minimizing unnecessary overhead. It also targets a simple system implementation and the best possible UE performance according to the CoMP deployment strategy chosen by the network operator (static vs. dynamic clusters).

During the discussion, it turned out that the offline feasibility improvement techniques and all the pre-clustering mechanisms can and should be applied in the described order to achieve best possible performance from a given network deployment. Which online network reconfiguration techniques are applied, however, needs to be selected carefully. They can be applied in parallel but they influence each other, which can lead to undesired effects. One way of avoiding such effects is to execute different techniques sequentially, followed by a new backhaul network pre-clustering step to consider the network changes of one technique before applying another.

Chapter 8

Conclusions and future research

Contents

This thesis approaches CoMP from a perspective that has mostly been neglected so far. It sheds light on CoMP's feasibility in real-world deployments, especially on the influence of the backhaul network infrastructure. We investigated the detailed requirements of different CoMP schemes, evaluated the feasibility of CoMP in various realistic backhauling scenarios, and presented mechanisms to deal with the shortcomings resulting from backhaul network limitations. Based on the conducted analysis and simulations, we draw the following conclusions.

8.1 Conclusions

There are strong limitations, even with future backhauling technologies. The evaluations have shown that in the promising dynamic clustering scenario even for small BS clusters of 3 BSs, the desired BSs often cannot cooperate due to backhaul limitations in terms of missing capacity or high latency. This is also true under the assumption that the backhaul network uses future optical PON technology. Networks using these technologies are already saturated in non-cooperative LTE-A networks.

A cross-domain approach is required. Neglecting the limited backhaul just leads to further load in the network although cooperative transmissions cannot happen in the end, i.e., the UE does not benefit, as required information arrives too late at the cooperating BSs or are dropped completely. On the other hand, in case a mobile network operator only wants small clusters (2-3 BSs) and provides a well-dimensioned backhaul network architecture, the feasible clusters from the backhaul perspective can be much larger than the desired ones. These differences between desired and feasible cluster sizes

lead to significant overhead in the backhaul and wireless network. Hence, the current status of the backhaul network needs to be taken into account when deciding when, where, and how to cooperate among BSs. A cross-domain decision, taking into account the wireless channel conditions *and* the current properties of the backhaul network, is needed. This way, unnecessary overhead can be reduced by up to 85 %.

The cross-domain approach needs to be fast. Due to the strict delay requirements of CoMP, the cross-domain approach needs to bring together information from the wireless and backhauling domain very quickly to find out which BSs should cooperate in the end. The proposed system consisting of wireless long-term pre-clustering and the heuristic algorithm for the backhaul network pre-clustering achieves this. Furthermore, the resulting clustering decisions are close to the optimum.

This allows to actually deploy CoMP techniques in current and future mobile networks without the need to immediately upgrading the whole backhauling infrastructure. Existing backhaul network resources are automatically exploited for CoMP as much as they allow and CoMP is not used where it is not feasible. After upgrading the backhaul infrastructure, it can be automatically used to support CoMP without the need to change the implemented CoMP mechanisms again.

For a ubiquitous use of CoMP, mechanisms are required to improve its feasibility. It has been shown in this thesis that upgrading the backhaul network infrastructure to use standard future PON technologies is not enough to fulfill CoMP requirements in mobile networks, like LTE-A and beyond. Therefore, to be able to exploit the full power of CoMP, mechanisms are required that further tweak these technologies to improve this shortcoming or mechanisms that exploit some degrees of freedom of the mobile network, which are unused so far.

We presented both generic mechanisms, which improve CoMP feasibility in arbitrary backhaul deployments, and mechanisms for PONs, which in addition exploit PON-specific advantages. Some of these mechanisms can even be applied in parallel to combine their advantages and to optimally exploit backhaul network infrastructures for CoMP. This pushes the feasibility of CoMP to a new level, which was not possible before.

Just considering the access part of the backhaul network is not enough. We have shown that CoMP even among just two BSs can easily involve more than just the access part of the backhaul network infrastructure. The exchanged data has to go one level up to the metro-wide network. This is problematic with today's commonly used ring networks on the metro level as the additional latency prohibits CoMP.

To avoid this limitation, we evaluated new, exclusively PON-based infrastructures, which, due to their high reach, cover both the metro and access network part. They show much higher CoMP feasibility but require high CAPEX to replace existing infrastructure, which is not a reasonable solution for network operators. Therefore, we developed a transition architecture that is based on novel PON techniques but reuses existing fiber

deployments. The low deployment costs compared to the clean-slate approach are traded off against a slightly reduced area in which CoMP is fully supported.

Consequently, using the proposed approach, the feasibility of CoMP can be improved not only locally within one branch of the backhaul network's access part, but also on a metro-wide scale. This is the base for providing CoMP techniques ubiquitously.

8.2 Future research topics

Based on the insight gained during the research work in the context of this thesis, I suggest the following fields for future research activities.

Extend static clustering mechanisms We presented mechanisms to calculate optimal as well as approximated configurations for static clustering environments. Due to the superiority of the dynamic clustering approach in terms of performance gain, the main focus of this thesis was shifted to this approach. Nevertheless, with some extended concepts, like overlapping clusters that create several wireless cluster layers in different physical resources [13], the static approach might be appealing to network operators due to the reduced complexity. In this case, the tools for static clustering, which have been presented in this work, need to be extended to include schemes like overlapping cluster layers, positioning controllers independently of BS locations, taking into account shared resources in tree or ring topologies.

CoMP/Non-CoMP interaction and policies In future mobile networks there will be at least two different classes of UEs: those that need CoMP and those that do not. This raises an important practical question for operating such a mixed system: Which UEs should be preferred? For example, instead of supporting cooperative transmissions to cell-edge UEs of a neighbor cell, which requires a lot of backhaul capacity in case of JP, multiple UEs could be served in the own cell without BS cooperation instead. This results in a big optimization problem, which spans over multiple cells and BSs and involves many different aspects, like scheduling, the backhaul network conditions, etc.

Develop more sophisticated multi-user policies The presented pre-clustering mechanisms are a powerful tool to exploit available network resources for CoMP as good as possible, avoiding the risk of congesting the backhaul network infrastructure. Without additional scheduling policies, however, UEs are assigned available resources according to their order in which they require CoMP transmission/reception, following a First Come First Serve (FCFS) policy. This, however, is usually not desired by mobile operators. Instead, resources shall be shared equally among UEs independently of their time of arrival or there should be defined priorities for them. Such scheduling also requires some kind of preemption in case an UE with higher priority joins.

Similar to that problem, when network reconfiguration techniques are used, the changed backhaul network properties do not only influence the cluster for which the

reconfiguration has been triggered. It can influence other BSs which are not even involved in the CoMP operation. Hence, policies are needed to decide how to cleverly resolve such multi-user issues to optimize the system behavior.

Interaction with admission control mechanisms Admission control is a mechanism for ensuring a minimum service quality for UEs in a cell. This is achieved by limiting, e.g., the total number of ongoing calls or the total throughput for all UEs within a cell. This procedure in conventional, non-cooperative cellular networks is based on the assumption that the wireless interface is the bottleneck for each cell. This, however, is not true anymore in future systems, like LTE-A, in particular when they apply CoMP techniques. Hence, it is not sufficient to just take the available wireless resources into account when performing admission control for new UEs in a cell. Available resources in the backhaul network have to be checked, too.

Evaluation based on real network topologies Although the assumptions on the backhaul network topologies in this thesis have been chosen to take into account much more details and to be much more realistic compared to existing evaluations, the simulations are still not based on "real" topologies. Topologies are generated based on properties of technologies, which are used in real deployments, but the deployment strategy might be different.

Prototyping proposed mechanisms and techniques The next step towards deployment of the proposed mechanisms is to create a CoMP testbed or prototype that includes these techniques. Only a few real-time testbed measurements on the wireless CoMP performance happened in the past. They, however, focus on wireless aspects and do not demonstrate the performance in limited backhaul scenarios. The influence of the backhaul network on CoMP's performance visible for the UEs has only been demonstrated in simulations. A testbed implementation, taking into account realistic backhaul limitations, and implementing the concepts proposed in this thesis, would eventually demonstrate the feasibility and benefits of our mechanisms in real systems, which has been shown in a first step by simulation in this thesis.

Influence on existing network functions Integrating the proposed mechanisms into a prototype implementation, which also takes into account other mechanisms of the mobile network, like hand-over or admission control, requires detailed planning of the interactions between the new CoMP features and the existing non-cooperative mechanisms. E.g., it might be required to adapt existing functions to take into account cooperative UEs in a special way, like the policies and scheduling issues regarding multiple users discussed earlier. Some existing features could even be changed completely or get much less important. Hand-over, e.g., is not that important anymore when serving UEs via CoMP as their movement to the cell edge does not necessarily imply a bad service quality. Such interactions open up many new research topics.

Appendix A

Passive Optical Networks

Due to their high bandwidth requirements, applications like P2P file sharing, high definition television, video on demand, and social networking turned out to be a challenge for communication networks [92]. While backbone capacity has increased to meet these requirements, high costs prevented access networks from scaling up accordingly.

The currently most promising architecture for access networks is a Passive Optical Network (PON) [93, 44]. PONs have only passive, i.e., non-powered, equipment in the field, which reduces energy, deployment, and maintenance costs. This way, high data rates can be provided at relatively low costs for, e.g., Fiber To The X (FTTx) applications (where x stands for various locations like home, curb, cabinet, or building) or backhauling BSs of wireless access networks.

The PON architecture is illustrated in Figure A.1. PONs have a physical tree topology with the root, the Optical Line Terminal (OLT), usually residing at the network operator. The OLT is connected to several ONUs in the field via an optical fiber that is split up using 1:N optical splitters/combiners or other, more sophisticated devices at the Remote Node (RN). The distance between the OLT and the RN is usually much longer than between the RN and the ONUs.

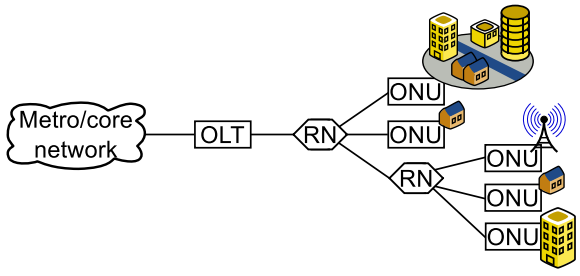


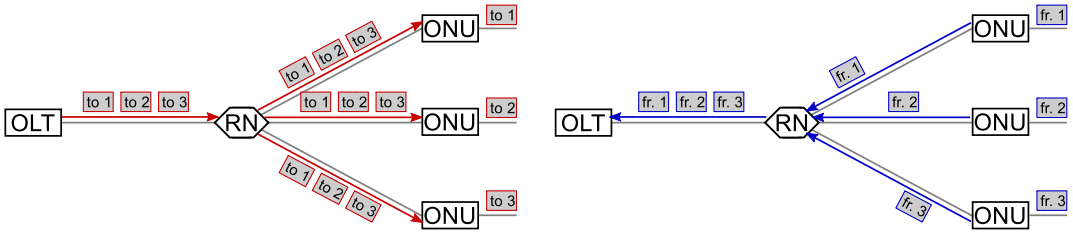
Figure A.1: Example PON tree. The OLT is connected to four ONUs via two RNs. A single ONU can connect a house, a company, a whole district, or (a set of) BSs.

PON systems can be divided into two main categories: TDM PONs and WDM PONs. Furthermore, there are two popular variants of the TDM PON approach, which

have been used for mass deployments so far: Gigabit PONs (GPONs) according to the ITU-T G.984 standard; it is widely deployed in the U.S. and Europe. Furthermore, there are EPONs according to the IEEE 802.3ah standard. They can often be found in Japan and Korea.

A.1 TDM PONs

A Time Division Multiplexing (TDM) PON uses two optical wavelengths, one for the uplink and another one for the downlink. Multiple ONUs are multiplexed within the two bands using TDM techniques. The resulting data transport is illustrated in Figure A.2.



(a) Downlink (b) Uplink

Figure A.2: Principle of TDM PONs. There are two optical wavelengths in use: one for the downlink (red arrows) and one for the uplink (blue arrows).

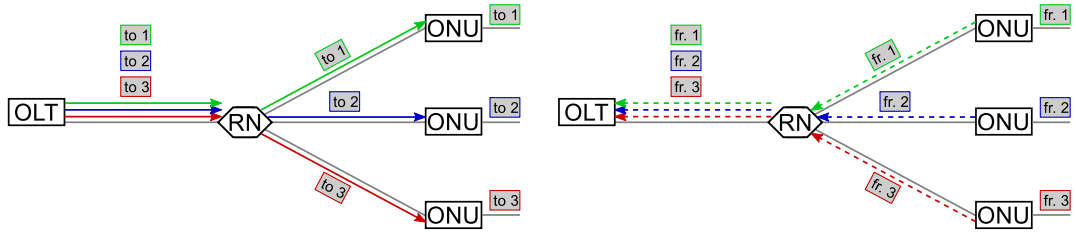
The RN of a TDM PON consists of a passive 1:N splitter that simply splits up the optical signal, and hence the power, that comes from the OLT. It also combines all signals that arrive from the ONUs. The power budget limits both the PON splitting ratio and the distance between the OLT and the ONUs. As a result, it will be hard to fulfill the requirements of future access networks using conventional TDM PONs as the aggregated bandwidth is not sufficient and the reach is too small.

The EPON architecture is defined in the IEEE 802.3 standard and follows the TDM approach. Upstream collisions are avoided by the Multi-Point Control Protocol. It is controlled by the DBA process that resides at the OLT and applies time division multiplexing among ONUs [93]. The upstream bandwidth allocation algorithm itself is not part of the standard. Several alternatives were proposed in the literature; one of the most prominent ones is IPACT [81]. It assigns timeslots to ONUs in a round-robin manner, based on the reported queue lengths.

Within EPONs, Ethernet frames are used to transport data. To address a certain ONU, a LLID is prepended to each frame, replacing two octets of the preamble. This way, the broadcast medium can be exploited for "free" single-copy multicast operations, like in wireless, by introducing multicast LLIDs (up to $2^{15} - 1 = 32767$ ONU groups). ONUs filter all incoming frames and pass only those frames to the MAC that carry a proper (group) LLID.

A.2 WDM PONs

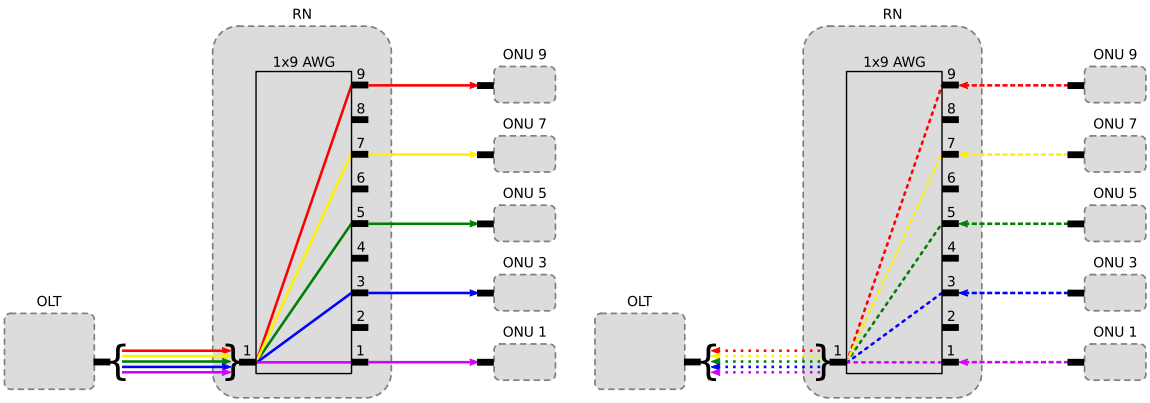
Problems of the TDM PON have been addressed in Wavelength Division Multiplexing (WDM) PONs. In these systems, each ONU is assigned an individual wavelength for uplink and downlink. This allows higher capacity for each ONU and a much lower optical power loss at the RN as the downlink optical energy is not split up among all ONUs but routed using an Array Waveguide Grating (AWG). Furthermore, since ONUs are separated physically via own wavelengths, privacy and security issues among multiple ONUs are inherently addressed. The WDM PON approach is shown in Figure A.3.



(a) Downlink (b) Uplink

Figure A.3: Principle of WDM PONs. There are $2N$ optical wavelengths in use, where N is the number of ONUs: one for each ONU's downlink (solid arrows) and uplink (dashed arrows).

What happens in detail at the AWG within the RN is shown in Figure A.4. In this example, the AWG has 1 uplink port to the OLT and 9 downlink ports to the ONUs, i.e., in total 10 fibers are connected to the RN.



(a) Downlink (b) Uplink

Figure A.4: Optical signal flow in a conventional WDM PON system. The different colors of the arrows indicate different wavelengths. Their line styles stand for the used optical waveband.

In a WDM PON, 2 different optical wavebands are used for downlink (L-band) and uplink (C-band) transmissions. The individual wavelength of ONU x for downlink

transmission $\lambda_{x,\text{down}}$ and for uplink $\lambda_{x,\text{up}}$ are exactly separated by a multiple of the AWG's FSR. This way, the diversion of the individual uplink and downlink wavelengths within the AWG for one ONU are identical, which results in signal propagation as shown in Figure A.4 on the preceding page.

WDM PONs can also be combined with additional TDM techniques, e.g., those that are already used in the EPON and GPON standards. This leads to hybrid WDM-TDM PON architectures, which have an improved scalability as splitting factors of up to 1000 become possible. Moreover, WDM PONs can be complemented with common WDM equipment, like optical broadband amplification. This allows distances on the order of 100 km and leads to the concept of "active PONs". Such solutions are especially interesting for future mobile backhaul networks and metro-access networks, i.e., PONs that cover the full distance from the core network to the user, which was not possible before due to the limited reach of a few kilometers of old access network technologies.

Bibliography

[1] Cisco Systems, Inc., "Cisco visual networking index: Forecast and methodology, 2010-2015," June 2011. White paper.

[2] ITU-R Rep. M.2134, "Requirements related to technical performance for IMT-advanced radio interface(s)," 2008. Report.

[3] V. Jungnickel, L. Thiele, T. Wirth, T. Haustein, S. Schiffermüller, A. Forck, S. Wahls, S. Jaeckel, S. Schubert, H. Gäbler, and Others, "Coordinated multipoint trials in the downlink," in *Proc. 5th IEEE Broadband Wireless Access Workshop (BWAWS)*, Nov. 2009.

[4] V. Jungnickel, M. Schellmann, L. Thiele, T. Wirth, T. Haustein, O. Koch, W. Zirwas, and E. Schulz, "Interference-aware scheduling in the multiuser MIMO-OFDM downlink," *IEEE Communications Magazine*, vol. 47, pp. 56–66, June 2009.

[5] S. Venkatesan, H. Huang, A. Lozano, and R. Valenzuela, "A WiMAX-based implementation of network MIMO for indoor wireless systems," *EURASIP Journal on Advances in Signal Processing*, 2009.

[6] W. Yu, T. Kwon, and C. Shin, "Multicell coordination via joint scheduling, beamforming and power spectrum adaptation," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, June 2011.

[7] J. He and M. Salehi, "Low-complexity coordinated interference-aware beamforming for MIMO broadcast channels," in *Proc. IEEE Vehicular Technology Conference (VTC-Fall)*, pp. 685–689, Sept. 2007.

[8] Y.-H. Nam, L. Liu, Y. Wang, C. Zhang, J. Cho, and J.-K. Han, "Cooperative communication technologies for LTE-advanced," in *Proc. IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pp. 5610–5613, Mar. 2010.

[9] W. Choi and J. G. Andrews, "The Capacity Gain from Intercell Scheduling in Multi-Antenna Systems," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 714–725, Feb. 2008.

[10] J. G. Andrews, A. Ghosh, and R. W. Heath, "Networked MIMO with clustered linear precoding," *IEEE Transactions on Wireless Communications*, vol. 8, pp. 1910–1921, Apr. 2009.

[11] V. Jungnickel, T. Wirth, M. Schellmann, T. Haustein, and W. Zirwas, "Synchronization of cooperative base stations," in *IEEE International Symposium on Wireless Communication Systems (ISWCS)*, pp. 329–334, 2008.

[12] J. Hoydis, M. Kobayashi, and M. Debbah, "On the optimal number of cooperative base stations in network MIMO systems," *Arxiv preprint arXiv:1003.0332*, 2010.

[13] G. Caire, S. A. Ramprashad, and H. C. Papadopoulos, "Rethinking Network MIMO: Cost of CSIT, Performance Analysis, and Architecture Comparisons," in *Proc. IEEE Information Theory and Applications Workshop (ITA)*, Jan. 2010.

[14] L. Thiele, M. Schellmann, S. Schiffermuller, V. Jungnickel, and W. Zirwas, "Multicell channel estimation using virtual pilots," *Proc. IEEE Vehicular Technology Conference (VTC-Spring)*, pp. 1211–1215, May 2008.

[15] O. Simeone, O. Somekh, H. V. Poor, and S. Shamai, "Downlink multicell processing with limited-backhaul capacity," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, 2009.

[16] T. Biermann, L. Scalia, C. Choi, H. Karl, and W. Kellerer, "CoMP clustering and backhaul limitations in cooperative cellular mobile access networks," *Elsevier Pervasive and Mobile Computing*, 2012.

[17] K. Miller, T. Biermann, H. Woesner, and H. Karl, "Network coding in passive optical networks," in *Proc. IEEE International Symposium on Network Coding (NetCod)*, IEEE, June 2010.

[18] T. Biermann, L. Scalia, J. Widmer, and H. Karl, "Backhaul design and controller placement for cooperative mobile access networks," in *Proc. IEEE Vehicular Technology Conference (VTC Spring)*, IEEE, May 2011.

[19] T. Biermann, L. Scalia, C. Choi, H. Karl, and W. Kellerer, "Backhaul network pre-clustering in cooperative cellular mobile access networks," in *Proc. IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, IEEE, June 2011.

[20] T. Biermann, L. Scalia, C. Choi, H. Karl, and W. Kellerer, "Improving CoMP cluster feasibility by dynamic serving base station reassignment," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, IEEE, Sept. 2011.

[21] C. Choi, L. Scalia, T. Biermann, and S. Mizuta, "Coordinated multipoint multiuser-MIMO transmissions over backhaul-constrained mobile access networks," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, IEEE, Sept. 2011.

[22] T. Biermann, L. Scalia, and H. Karl, "Designing optical metro and access networks for future cooperative cellular systems," in *Proc. ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pp. 265–274, ACM, Oct. 2011.

[23] C. Choi, Q. Wei, T. Biermann, and L. Scalia, "Mobile WDM backhaul access networks with physical inter-base-station links for coordinated multipoint transmission/reception systems," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, IEEE, Dec. 2011.

[24] W. Kellerer, W. Kiess, L. Scalia, T. Biermann, C. Choi, and K. Kozu, "Novel cellular optical access network and convergence with FTTH," in *Proc. Optical Fiber Communication Conference and Exposition (OFC) and The National Fiber Optic Engineers Conference (NFOEC)*, Mar. 2012. Invited paper.

[25] M. Dräxler, T. Biermann, H. Karl, and W. Kellerer, "Cooperating base station set selection and network reconfiguration in limited backhaul networks," in *Proc. IEEE Personal Indoor Mobile Radio Communications (PIMRC)*, 2012.

[26] T. Biermann, L. Scalia, C. Choi, K. Kozu, and W. Kellerer, "Apparatus and method for determining a core network configuration of a wireless communication system," 2010.

[27] L. Scalia, T. Biermann, C. Choi, K. Kozu, and W. Kellerer, "Apparatus and method for controlling a node of a wireless communication system," 2010.

[28] C. Choi, Q. Wei, and T. Biermann, "Communication systems and method for directly transmitting signals between nodes of a communication system, node and optical multiplexer/demultiplexer device," 2010.

[29] L. Scalia, T. Biermann, and C. Choi, "Apparatus and method for determining a control unit using feasibility requests and feasibility response," 2011.

[30] C. Choi, L. Scalia, T. Biermann, and S. Mizuta, "Method for coordinated multipoint communication in wireless communication network, wireless communication system, base station and network controller," 2011.

[31] C. Choi, T. Biermann, and Q. Wei, "Central node for a communication system, wireless communication system, optical multiplexer/demultiplexer device, and method for transmitting data to one or more groups of nodes in a communication system," 2011.

[32] L. Scalia, T. Biermann, and C. Choi, "Apparatus and method for determining a cluster of base stations," 2011.

[33] T. Biermann, L. Scalia, C. Choi, H. Karl, and W. Kellerer, "On the feasibility of coordinated multi-point base station clusters in cellular networks," *IEEE Communications Magazine*, 2012. Submitted for publication.

[34] Huawei Technologies Co. Ltd., "Huawei response to ACMA's paper: Towards 2020 – future spectrum requirements for mobile broadband," 2011.

[35] J. Zhang, R. Chen, J. Andrews, A. Ghosh, and R. Heath, "Networked MIMO with clustered linear precoding," *IEEE Transactions on Wireless Communications*, vol. 8, pp. 1910–1921, Apr. 2009.

[36] I. D. Garcia, N. Kusashima, K. Sakaguchi, K. Araki, S. Kaneko, and Y. Kishi, "Impact of base station cooperation on cell planning," in *Proc. IEEE International Conference on Communications (ICC)*, 2011.

[37] Papadogiannis, D. Gesbert, and E. Hardouin, "A dynamic clustering approach in wireless networks with multi-cell cooperative processing," in *Proc. IEEE International Conference on Communications (ICC)*, 2008.

[38] A. Papadogiannis and G. C. Alexandropoulos, "The value of dynamic clustering of base stations for future wireless networks," in *Proc. IEEE International Conference on Fuzzy Systems (FUZZ)*, 2010.

[39] P. Marsch and G. Fettweis, "Static clustering for cooperative multi-point (CoMP) in mobile communications," *EURASIP Journal on Wireless Communications and Networking*, 2010.

[40] P. Marsch and G. P. Fettweis, *Coordinated Multi-Point in Mobile Communications*. Cambridge University Press, 2011.

[41] J. G. Stefan Brueck, Lu Zhao and M. A. Amin, "Centralized scheduling for joint transmission coordinated multi-point in LTE-advanced," in *Proc. International ITG Workshop on Smart Antennas (WSA)*, 2010.

[42] M. Goldenbaum, R. Abi Akl, and S. Valentin, "On the effect of feedback delay in the downlink of multiuser OFDM systems," in *Proc. Conference on Information Sciences and Systems (CISS)*, 2011.

[43] P. Croy, "LTE backhaul requirements: A reality check," Feb. 2011. White paper, Aviat.

[44] K. Grobe and J.-P. Elbers, "PON in adolescence: From TDMA to WDM-PON," *IEEE Communications Magazine*, vol. 46, pp. 26–34, Jan. 2008.

[45] P. Marsch and G. Fettweis, "On base station cooperation schemes for downlink network MIMO under a constrained backhaul," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, 2008.

[46] D. Samardzija and H. Huang, "Determining backhaul bandwidth requirements for network MIMO," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2009.

[47] R. Zakhour and D. Gesbert, "On the value of data sharing in constrained-backhaul network MIMO," in *Proc. Int. Zurich Seminar on Communications (IZS)*, 2010.

[48] A. Sanderovich, O. Somekh, and S. Shamai, "Uplink macro diversity with limited backhaul capacity," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, 2007.

[49] F. Boccardi, H. Huang, and A. Alexiou, "Network MIMO with reduced backhaul requirements by MAC coordination," in *Proc. Asilomar Conference on Signals, Systems, and Computers*, 2008.

[50] P. Marsch and G. Fettweis, "On downlink network MIMO under a constrained backhaul and imperfect channel knowledge," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, 2009.

[51] H. Salama, D. Reeves, and Y. Viniotis, "Evaluation of multicast routing algorithms for real-time communication on high-speed networks," *Selected Areas in Communications*, vol. 15, no. 3, pp. 332–345, 2002.

[52] W. Zhengying, S. Bingxin, and Z. Erdun, "Bandwidth-delay-constrained least-cost multicast routing based on heuristic genetic algorithm," *Computer Communications*, vol. 24, no. 7-8, pp. 685–692, 2001.

[53] G. Feng and T.-s. P. Yum, "Efficient multicast routing with delay constraints," *International Journal of Communication Systems*, pp. 181–195, 1999.

[54] T. Cohen and R. Southwood, "Extending open access to national fibre backbones in developing countries," in *International Telecommunication Union (ITU) 8th Global Symposium for Regulators*, 2008.

[55] L. Minas and B. Ellison, "The problem of power consumption in servers," in *Energy Efficiency for Information Technology*, Intel Press, 2009.

[56] L. Thiele, F. Boccardi, C. Botella, T. Svensson, and M. Boldi, "Scheduling-assisted joint processing for CoMP in the framework of the WINNER+ project," in *Future Network and Mobile Summit*, IEEE, 2010.

[57] J. N. Laneman, *Cooperative Diversity in Wireless Networks: Algorithms and Architectures*. PhD thesis, Massachusetts Institute of Technology, 2002.

[58] J. Laneman and D. Tse, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *Information Theory, IEEE*, vol. 50, no. 12, pp. 3062–3080, 2004.

[59] T. Abe, "3GPP self-evaluation methodology and results – assumptions," 2009. Presenation at 3GPP LTE-Advanced Evaluation Workshop.

[60] A. Benjebbour, M. Shirakabe, Y. Ohwatari, J. Hagiwara, and T. Ohya, "Evaluation of user throughput for MU-MIMO coordinated wireless networks," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2008.

[61] I. Forkel, M. Schinnenburg, and M. Ang, "Generation of two-dimensional correlated shadowing for mobile radio network simulation," in *Proc. International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 2004.

[62] V. Jungnickel, S. Jaeckel, K. Börner, M. Schlosser, and L. Thiele, "Estimating the mobile backhaul traffic in distributed coordinated multi-point systems," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, 2012.

[63] A. Popescu and D. Constantinescu, "Measurement of one-way transit time in IP routers," tutorial, Blekinge Institute of Technology (BTH), Blekinge, Sweden, 2005.

[64] C. Hoymann, L. Falconetti, and R. Gupta, "Distributed uplink signal processing of cooperating base stations based on IQ sample exchange," in *Proc. IEEE International Conference on Communications (ICC)*, 2009.

[65] L. Falconetti, C. Hoymann, and R. Gupta, "Distributed uplink macro diversity for cooperating base stations," in *Proc. International Workshop on LTE Evolution*, 2009.

[66] A. Papadogiannis, D. Gesbert, and E. Hardouin, "A dynamic clustering approach in wireless networks with multi-cell cooperative processing," in *Proc. IEEE International Conference on Communications (ICC)*, 2008.

[67] C. E. Leiserson and T. B. Schardl, "A work-efficient parallel breadth-first search algorithm (or how to cope with the nondeterminism of reducers)," in *Proc. ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pp. 303–314, 2010.

[68] 3GPP, "Requirements for further advancements for E-UTRA (LTE-Advanced) (TR 36.913)," 2009.

[69] M. Rumney, "3GPP LTE: Introducing single-carrier FDMA," *Agilent Measurement Journal*, vol. 4, pp. 18–27, 2008.

[70] H. Falaki, D. Lymberopoulos, R. Mahajan, S. Kandula, and D. Estrin, "A first look at traffic on smartphones," in *Proc. ACM Conference on Internet Measurement (IMC)*, 2010.

[71] NEC Corporation, "Evolving microwave mobile backhaul for next-generation networks," 2008. White paper.

[72] S. Ghosh, P. De, K. Basu, and S. K. Das, "PeterNet: A free space optics based backhaul architecture for next generation cellular networks," tech. rep., Center for Research in Wireless Mobility and Networking, Department of Computer Science and Engineering, The University of Texas at Arlington, 2011.

[73] N. M. K. Chowdhury and R. Boutaba, "A survey of network virtualization," *Computer Networks*, vol. 54, pp. 862–876, Apr. 2010.

[74] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Transactions on Information Theory*, vol. 49, pp. 371–381, 2003.

[75] S. Katti, H. Rahul, H. Wenjun, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," *IEEE/ACM Transactions on Networking*, vol. 16, pp. 497–510, 2008.

[76] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications* . Prentice Hall, 1993.

[77] M. Belzner and H. Haunstein, "Network coding in passive optical networks," in *Proc. European Conference and Exhibition on Optical Communication (ECOC)*, 2009.

[78] T. Biermann, M. Dräxler, and H. Karl, "Flow synchronization for network coding," *Journal of Communications*, vol. 4, no. 11, pp. 873–884, 2009.

[79] B. Ahlgren, M. D'Ambrosio, M. Marchisio, I. Marsh, C. Dannewitz, B. Ohlman, K. Pentikousis, O. Strandberg, R. Rembarz, and V. Vercellone, "Design considerations for a network of information," in *Proc. ACM International Conference on Emerging Networking Experiments and Technologies (CoNEXT)*, 2008.

[80] H. A. David, *Order statistics*. New York: Wiley, 1970.

[81] G. Kramer, B. Mukherjee, and G. Pesavento, "IPACT: A dynamic protocol for an Ethernet PON (EPON)," *IEEE Communications Magazine*, vol. 40, pp. 74–80, 2002.

[82] S. Kim, J. Y. Lee, and D. K. Sung, "A shifted gamma distribution model for long-range dependent Internet traffic," *IEEE Communications Letters*, vol. 7, no. 3, pp. 124–126, 2003.

[83] J. Seedorf and E. Burger, "Application-layer traffic optimization (ALTO) problem statement." IETF Draft (Informational), Apr. 2009.

[84] V. Aggarwal and A. Feldmann, "Locality-aware P2P query search with ISP collaboration," *Networks and Heterogeneous Media*, vol. 3, no. 2, pp. 251–265, 2008.

[85] 3GPP, "Reply LS to R3-070527/R1-071242 on Backhaul (X2 interface) Delay (R3-070702/R1-071804)," 2007.

[86] Next Generation Mobile Networks (NGMN) Alliance, "Guidelines for LTE backhaul traffic estimation," 2011. White paper.

[87] T. Pfeiffer, "Converged heterogeneous optical metro-access networks," in *Proc. European Conference and Exhibition on Optical Communication (ECOC)*, 2010.

[88] J. E. Simsarian, P. Bernasconi, J. Gripp, M. C. Larson, D. T. Neilson, and J. Sinsky, "Fast tunable lasers for optical routers and networks," in *Proc. Conference on Lasers and Electro-Optics (CLEO) and Quantum Electronics and Laser Science Conference (QELS)*, 2006.

[89] K. Grobe, "Performance, cost, and energy consumption in next-generation WDM-based access," in *Proc. International Conference on Broadband Communications, Networks, and Systems (BROADNETS)*, 2010.

[90] N. Hohn, K. Papagiannaki, and D. Veitch, "Capturing router congestion and delay," *IEEE/ACM Transactions on Networking*, vol. 17, no. 3, pp. 789–802, 2009.

[91] S. Nakazawa, H. Tamura, K. Kawahara, and Y. Oie, "Performance analysis of IP datagram transmission delay in MPLS: Impact of both the number and the bandwidth of LSPs of layer 2," *Proc. IEEE International Conference on Communications (ICC)*, pp. 1006–1010, 2002.

[92] Cisco Systems, Inc., "Cisco visual networking index – forecast and methodology, 2008-2013," June 2009. White paper.

[93] M. P. McGarry, M. Reisslein, and M. Maier, "Ethernet passive optical network architectures and dynamic bandwidth allocation algorithms," *IEEE Communications Surveys and Tutorials*, vol. 10, pp. 46–60, 2008.