EFFICIENT TOPOLOGY DESIGN METHODS
FOR NEXT GENERATION ACCESS NETWORKS

Attila Mitcsenkov
M.Sc. in Technical Informatics

Ph. D. Dissertation
Doctoral School of Informatic Sciences

Supervised by
Dr. Tibor Cinkler

High Speed Networks Laboratory
Department of Telecommunications and Media Informatics
Budapest University of Technology and Economics

Budapest, Hungary
2014
ABSTRACT

After conquering the core and transport networks, optical communications started to invade the access networks. In the recent years, the so-called not next generation access (NGA) networks became economically viable. Various network technologies exist and compete in this field, but a common feature of them is the use of optical fiber in some extent (Fiber-to-the-X, FTTx systems). Even though such network technologies became commercially available, we do not see rapid, large scale deployments in the access networks. More than 50% of the total investments in the ICT sector are made in the access, due to high density of equipment and the large areas to cover. Such high investment calls for wise economic decisions. Fundamental information for such decision is the estimation of the necessary investment, i.e. network deployment.

The generally accepted way to provide such cost estimations was the use of statistical measures of past deployments, or the application of geometric models to give a rough estimation on the “bill of material”, i.e. the necessary equipment and installation. My proposed, novel approach in this field is to make the cost estimation more accurate by integrating network planning in the process. If we are able to provide a “strategic network design” before actually deploying the network, we get a significantly more detailed and accurate image of the network itself, and such knowledge about the network topology and the “bill of material”, among others, support more accurate deployment cost estimation.

Obviously, this network planning process has to be automatic, not calling for a group of experienced network planning professionals even for the cost estimation process. Therefore, my research efforts were targeting algorithmic topology design for next generation access networks. The appearance of digital maps and geospatial databases was a significant “enabler” for such methodology, providing the necessary input for the computer.

In the first part of my work, I have transformed the engineering challenge to a mathematically formulated optimization problem, with a suitable graph model. Following the classification of NGA network architectures, I have identified a set of respective special cases. Using the formal model, I was able to analyze the complexity and approximability of the topology design problem and its special cases. Such analysis also revealed the key subproblems, which need to be solved wisely to deliver optimized network topologies.

After laying the theoretical fundamentals, I have investigated several techniques to actually solve the optimization problem. At first, I have formulated a (quadratically constrained) mathematical program, then by linearization and a modeling trick I was able to reduce its complexity – even though it is still restricted to small scale problem instances. I have also developed a generic heuristic solution, based on Simulated Annealing, which scales up to mid-scale scenarios. And finally, I proposed specialized, highly scalable heuristics, which are able to provide optimized network topologies even for large-scale, real world scenarios with up to tens of thousands of endpoints (which is a typical access network area). I have validated the proposed solutions on regular grid structures, and the evaluated them using real-world case studies (and data from industrial partners). I was able to show the practical applicability and remarkable performance of the proposed solutions, especially the specialized, highly scalable heuristics.
# Table of Contents

**Abstract** .................................................................................................................................. III

1. **Introduction** ........................................................................................................................... 1
   1.1. Next Generation Access Networks (NGA) .......................................................................... 1
   1.2. Network Planning .............................................................................................................. 2
       1.2.1. Techno-economic evaluation ...................................................................................... 3
   1.3. Related Work .................................................................................................................... 3
   1.4. Research objectives .......................................................................................................... 5
   1.5. Research methodology ..................................................................................................... 5
   1.6. Outline of the dissertation ............................................................................................... 6

2. **The Topology Design Problem** ............................................................................................ 7
   2.1. Parameterized graph model ............................................................................................ 7
   2.2. Cost function (Expenditures) ......................................................................................... 10
       2.2.1. Network equipment .................................................................................................... 10
       2.2.2. Cable plant ................................................................................................................ 10
       2.2.3. Topology dependence ............................................................................................... 11
   2.3. Technology dependent constraints ................................................................................ 12
   2.4. Optimization problem formulation ............................................................................... 13
   2.5. Impact of cost components and constraints ................................................................ 15
   2.6. Special cases .................................................................................................................. 16
       2.6.1. Special case #1 (Passive Optical Networks, PON) .................................................... 17
       2.6.2. Special case #2 (Active Optical Networks, AON) .................................................... 18
       2.6.3. Special case #3 (Digital Subscriber Line networks, DSL with optical feeder) ......... 20
       2.6.4. Special case #4 (Point to Point fiber access networks, P2P) .................................... 21
       2.6.5. Summary of special cases ......................................................................................... 22

3. **Algorithmic Analysis** .......................................................................................................... 23
   3.1. Complexity analysis ......................................................................................................... 23
       3.1.1. Special case #1 (Passive Optical Networks, PON) .................................................... 23
       3.1.2. Special case #2 (Active Optical Networks, AON) .................................................... 26
3.1.3. Special case #3 (Digital Subscriber Line networks, DSL) ..................................................... 28
3.1.4. Special case #4 (Point to Point fiber access networks, P2P) ............................................... 30
3.1.5. NTD problem in general ...................................................................................................... 32

3.2. Approximability studies ........................................................................................................... 32
3.2.1. Special case #1 (Passive Optical Networks, PON) ............................................................... 33
3.2.2. Special case #2 (Active Optical Networks, AON) ................................................................. 33
3.2.3. Special case #3 (Digital Subscriber Line networks, DSL) ..................................................... 33
3.2.4. Special case #4 (Point to Point fiber access networks, P2P) ............................................... 34
3.2.1. NTD problem in general ...................................................................................................... 34

3.3. Conclusion ................................................................................................................................. 35

4. CRITICALITY: A GRAPH THEORETIC APPROACH .................................................................. 36
4.1. Definition of criticality metric and ordering .............................................................................. 36
4.2. Matching and criticality: an upper bound on DU locations ................................................ 39
4.3. Lower bounds of DU set ........................................................................................................... 41

5. SPECIALIZED, HIGHLY EFFICIENT HEURISTICS .................................................................. 43
5.1. Decomposition into subproblems ............................................................................................ 44
5.2. Passive Optical Networks (PON) ............................................................................................. 45
5.2.1. Branch Contracting Algorithm (BCA) ................................................................................... 45
5.2.2. Complexity .......................................................................................................................... 48
5.2.3. Approximation performance .............................................................................................. 49
5.3. Active Optical Networks (AON) ............................................................................................... 49
5.3.1. Iterative Neighbor Contracting Algorithm (INCA) ............................................................... 50
5.3.2. Complexity .......................................................................................................................... 51
5.3.3. Approximation performance .............................................................................................. 52
5.4. Digital subscriber Line (DSL) networks .................................................................................. 52
5.4.1. Stepwise Allocation of Critical DUs (SACD) algorithm ...................................................... 52
5.4.2. Complexity .......................................................................................................................... 53
5.4.3. Approximation performance .............................................................................................. 54

6. EXACT OPTIMIZATION AND METAHEURISTICS .................................................................. 56
6.1. Mathematical programming ..................................................................................................... 57
6.1.1. Quadratic Programming (QP) formulation .......................................................................... 57
6.1.2. Mixed Integer Programming (MIP) formulation .................................................................. 59
6.1.3. Aggregated flows: a reduced linear programming formulation ............................................ 60
6.1.4. Overview of Mathematical Programming Formulations .................................................... 62
6.2. Metaheuristics ........................................................................................................................... 62
   6.2.1. Simulated Annealing ........................................................................................................... 62
   6.2.2. Simulated Annealing for the NGA topology design problem .............................................. 63
7. NUMERICAL RESULTS: VALIDATION AND EVALUATION ........................................ 67
   7.1. Validation: calculations on grid topology ................................................................................ 67
       7.1.1. Optimal solution: analytic calculations ........................................................................ 68
       7.1.2. Example grids ............................................................................................................... 72
       7.1.3. Numerical results ......................................................................................................... 72
   7.2. Evaluation ............................................................................................................................. 76
       7.2.1. Case studies ................................................................................................................. 76
       7.2.2. Approximation performance ....................................................................................... 79
       7.2.3. Resource requirements .............................................................................................. 91
   7.3. Large-scale scenarios and practical applications ................................................................. 93
8. CONCLUSION ............................................................................................................................ 96
   8.1. Contributions ....................................................................................................................... 96
   8.2. Applications ......................................................................................................................... 97
       8.2.1. NGAdesigner Framework ............................................................................................ 97
   8.3. Techno-economic studies ................................................................................................... 98
   8.4. Experience & reference ..................................................................................................... 99
9. ACKNOWLEDGEMENTS .......................................................................................................... 100
10. REFERENCES ......................................................................................................................... 101
LIST OF FIGURES

Figure 1 NGA technology classification ............................................................................................................ 2
Figure 2 Point-Multipoint Access Network Architecture .......................................................................................... 8
Figure 3 Graph model of the NGA topology design problem ...................................................................................... 9
Figure 4 Network Graph Model Example .............................................................................................................. 9
Figure 5 Cable plant costs ......................................................................................................................................... 11
Figure 6 Passive Optical Network (PON) ............................................................................................................ 17
Figure 7 Active Optical Network (AON) ............................................................................................................... 18
Figure 8 Digital Subscriber Line Network (DSL) ..................................................................................................... 20
Figure 9 Point-to-Point Fiber Access Network (P2P) ................................................................................................. 21
Figure 10 Special cases of the NGA topology design problem ..................................................................................... 22
Figure 11 Facility location problem – Network topology design problem transformation .............................................. 25
Figure 12 Graph transformation for CPMP .............................................................................................................. 27
Figure 13 Steiner tree transformation of DSL networks ............................................................................................ 28
Figure 14 Example Graph with Criticality values and components ................................................................................ 38
Figure 15 Simplified graph with components .......................................................................................................... 38
Figure 16 Bipartite view of the simplified graph ........................................................................................................ 38
Figure 17 Bipartite sample graph .................................................................................................................................. 39
Figure 18 Tight inequality example ............................................................................................................................ 40
Figure 19 Decomposition of the NGA Topology Design problem .................................................................................. 45
Figure 20 Shortest Path Tree ........................................................................................................................................ 46
Figure 21 Group formulation in BCA ........................................................................................................................ 47
Figure 22 DU allocation and connection establishment .............................................................................................. 47
Figure 23 Flowchart of the Branch Contracting Algorithm (BCA) ............................................................................. 48
Figure 24 Voronoi-diagram ......................................................................................................................................... 50
Figure 25 Flowchart of the Iterative Neighbor Contracting Algorithm (INCA) .................................................................. 51
Figure 26 Flowchart of the Stepwise Allocation of Critical DUs (SACD) algorithm ..................................................... 53
Figure 27 Bounding the optimality gap ..................................................................................................................... 56
Figure 28 Aggregated MIP: the flow problem ............................................................................................................ 60
Figure 29 Flowchart of the Simulated Annealing (SA) process .................................................................................... 63
Figure 30 Convergence speed vs. optimality ............................................................................................................ 65
Figure 31 Simulated annealing process dynamics ..................................................................................................... 66
Figure 32 Grid topology (N=8, L=100, k=4, d=10) ..................................................................................................... 67
Figure 33 Distribution fiber in the grid ........................................................................................................................ 68
Figure 34 Connectivity of demand points .................................................................................................................. 70
Figure 35 Minimal deployment calculations .............................................................................................................. 70
Figure 36 Validation of the BCA heuristics on grids: total cost ..................................................................................... 73
Figure 37 Cost component weights (GPON on grids) ................................................................................................. 73
Figure 38 PON heuristics vs. OPT – Details I ............................................................................................................. 74
Figure 39 PON heuristics vs. OPT – Details II ............................................................................................................ 74
Figure 40 Grid structure side-effect ........................................................................................................................ 74
Figure 41 Validation of the INCA heuristics on grids ................................................................................................. 75
Figure 42 Validation of the SACD heuristics on grids ............................................................................................... 75
Figure 43 reference points for evaluation .................................................................................................................. 76
Figure 44 Input data for a case study .................................................................................................... 76
Figure 45 Kőszeg map ............................................................................................................................ 78
Figure 46 Kecel map ............................................................................................................................... 78
Figure 47 Solymár map .......................................................................................................................... 78
Figure 48 Sashegy map .......................................................................................................................... 78
Figure 49 Újpalota map ......................................................................................................................... 79
Figure 50 MIP lower and upper bound dynamics #1 ............................................................................. 80
Figure 51 MIP lower and upper bound dynamics #2 ............................................................................. 80
Figure 52 total cost of GPON networks: BCA heuristics vs. MIP solution .............................................. 81
Figure 53 GPON cost components I ...................................................................................................... 82
Figure 54 GPON cost components II ................................................................................................... 82
Figure 55 Comparison of cost components: MIP vs. BCA heuristics I for GPON networks (Solymár) .. 82
Figure 56 Comparison of various cost components: MIP vs. BCA heuristics for GPON networks II..... 83
Figure 57 total cost of AETH networks: INCA heuristics vs. MIP ........................................................... 83
Figure 58 Cost component weights for AETH networks ....................................................................... 84
Figure 59 Comparison of cost components: MIP vs. INCA heuristics for AETH networks I. (Kőszeg) ... 84
Figure 60 Comparison of various cost components: MIP vs. INCA heuristics for AETH networks II. .... 85
Figure 61 INCA vs. MIP: excess Cost by distinct components .............................................................. 85
Figure 62 total cost of VDSL networks: SACD heuristics vs. MIP ........................................................... 86
Figure 63 Comparison of cost components: MIP vs. SACD heuristics for VDSL networks I. (Kecel) ...... 87
Figure 65 VDSL topology designed by SACD heuristics (Kecel) .............................................................. 88
Figure 66 Cost component weights for AETH networks ....................................................................... 88
Figure 67 INCA vs. MIP: Cost addition by distinct cost components ..................................................... 89
Figure 68 Cost comparison of MIP, SA and BCA heuristics .................................................................. 90
Figure 69 Cost comparison of MIP, SA and INCA heuristics ................................................................. 90
Figure 70 Cost comparison of MIP, SA and SACD heuristics ................................................................. 91
Figure 71 Time consumption of BCA, SA and MIP solutions ................................................................. 91
Figure 72 Time consumption analysis .................................................................................................. 93
Figure 73 Map of district XII ................................................................................................................ 94
Figure 74 Total cost comparison on a large scale scenario (District XII) .................................................. 95
Figure 75 Running time comparison on a large scale scenario (District XII) ........................................... 95
Figure 76 Deployment cost comparison of GPON, AETH and VDSL on the “Sashegy” scenario ........... 99
Figure 77 Cost component weights of GPON, AETH and VDSL on the “Sashegy” scenario ............... 99
LIST OF TABLES

Table 1 Outline of the dissertation .......................................................................................................... 6
Table 2 Summary of cost components ........................................................................................................ 12
Table 3 Topology dependent constraints .............................................................................................. 13
Table 4 Summary on theoretical background .......................................................................................... 35
Table 5 Overview of mathematical programming formulations ............................................................. 62
Table 6 Grid networks for validation ..................................................................................................... 72
Table 7 Scenarios / Case studies ............................................................................................................ 77
Table 8 Cost parameters ........................................................................................................................ 77
Table 9 Technology specifications and physical constraints .................................................................... 77
Table 10 CPLEX & computer specifications ........................................................................................... 80
Table 11 Specifications of “District XII” case study .............................................................................. 94
<table>
<thead>
<tr>
<th><strong>LIST OF ACRONYMS</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AETH</strong></td>
</tr>
<tr>
<td><strong>AON</strong></td>
</tr>
<tr>
<td><strong>BCA</strong></td>
</tr>
<tr>
<td><strong>CAPEX</strong></td>
</tr>
<tr>
<td><strong>CFL</strong></td>
</tr>
<tr>
<td><strong>CO</strong></td>
</tr>
<tr>
<td><strong>CPMP</strong></td>
</tr>
<tr>
<td><strong>DSL</strong></td>
</tr>
<tr>
<td><strong>DU</strong></td>
</tr>
<tr>
<td><strong>FTTx</strong></td>
</tr>
<tr>
<td><strong>GIS</strong></td>
</tr>
<tr>
<td><strong>GPON</strong></td>
</tr>
<tr>
<td><strong>INCA</strong></td>
</tr>
<tr>
<td><strong>MIP</strong></td>
</tr>
<tr>
<td><strong>NGA</strong></td>
</tr>
<tr>
<td><strong>NTD</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>OPEX</strong></td>
</tr>
<tr>
<td><strong>P2P</strong></td>
</tr>
<tr>
<td><strong>PON</strong></td>
</tr>
<tr>
<td><strong>QP</strong></td>
</tr>
<tr>
<td><strong>SA</strong></td>
</tr>
<tr>
<td><strong>SACD</strong></td>
</tr>
<tr>
<td><strong>SU</strong></td>
</tr>
<tr>
<td><strong>VDSL</strong></td>
</tr>
</tbody>
</table>
1. INTRODUCTION

Telecommunications research and development is motivated by the continuous evolution of service needs and requirements. New technologies are emerging, new principles are evolving, and the networks are facing new and new challenges. Beyond the primary challenge of increasing bandwidth needs, future (internet) services have several additional requirements, e.g. for latency or reliability of the communication network.

Evolution of core and access networks is strongly connected: core networks have to serve the traffic arriving from access networks, or in contrary: even if enormous bandwidth is available in the core network, customers without the necessary high speed access do not benefit from it. After the glorious last decade of optical transmission in core networks, time has come for fiber communications in the access networks. The technical and economic challenges were preventing the exploitation of optical transmission in the access network until the recent years. Optical network technologies based on simple, cheap equipment, such as passive optical networks are the fundamental enablers of the development. The term “Next Generation Access” (NGA) network refers to the fully or partly optical access networks, fulfilling the above mentioned requirements of future (internet) services.

1.1. NEXT GENERATION ACCESS NETWORKS (NGA)

Access networks of the (near) future have to face a set of service requirements that enforces substantial changes in the network technology: complete or partial replacement of the copper networks with optical fiber. In the short term, 100 Mbit/s bandwidth requirements have to be fulfilled on a per customer basis, and for the 2020 time horizon, the European goal is to keep up with capacity growth of at least 1 Gb/s in the wireless, and 10 Gb/s in the wired access. Offered new services, e.g. HD VoD (High Definition Video on Demand), VoIP (Voice over IP), videoconferencing and high speed internet access require high bandwidth and low delay simultaneously [1]-[2].

The depth of fiber installation in the network makes distinction between Fiber to the Home/Building (FTTH/FTTB) architectures, i.e. the complete replacement of the copper network with optical fiber; and Fiber to the Cabinet/Neighborhood (FTTC/FTTN) architectures, i.e. partial replacement of the copper network with optical fiber [3]. A differentiation can also be made between point-to-point (P2P) and point-multipoint (P2MP) systems. In a P2P point network, the demand points are connected directly to their respective Central Office via a dedicated optical fiber, while in a P2MP networks the demand points are connected to the Central Office (CO) or Point of Presence (PoP) through several aggregation/distribution nodes in the cable plant.

PASSIVE OPTICAL NETWORKS (PON)

Passive optical networks rely completely on optical connectivity. They have a point-to-multipoint architecture with passive devices as aggregation/distribution nodes within the cable plant. Currently existing PON technologies are e.g. APON [11], BPON [12], EPON [13] or GPON [14] and their 10G counterparts, 10G EPON [15] and XGPON [16], while WDM PON systems are just showing up [4]-[8].

ACTIVE OPTICAL NETWORKS (AON)

Active optical networks are similar to their passive counterparts, with the significant difference of using active aggregation/distribution equipment in the cable plant. Active Ethernet networks are a significant current example of the AON architecture [17].
**DIGITAL SUBSCRIBER LINE NETWORKS (DSL)**

Fiber to the Cabinet (FTTC) or Fiber to the Neighborhood (FTTN) architectures provide a reasonable tradeoff between investment and service level improvement simply by reusing the existing copper connectivity on the very last segment of the access networks. All of the xDSL (e.g. ADSL [20], ADSL2 [21], ADSL2+ [22], VDSL [23], VDSL2 [24]...) technologies represent the family of DSL networks: these have very similar network architecture, at least on a higher abstraction level [25].

**POINT-TO-POINT FIBER NETWORKS (P2P)**

In contrary to the point-multipoint PON/AON network architectures, in the case of P2P networks, the demand points are connected directly to their respective Central Office (i.e. Point of Presence) via a dedicated optical fiber [18]-[19].

![Network architecture](image)

**FIGURE 1 NGA TECHNOLOGY CLASSIFICATION**

1.2. **NETWORK PLANNING**

Optical networks provide a future proof platform for a wide range of services at the expense of replacing the cable plant. Unfortunately this “expense” is an enormous investment, which has to be justified by long term profitability; therefore, optimal network planning plays crucial role. Reducing cost of the network deployment (CAPEX – CAPital EXPenditure) is a natural requirement, and also the future OPEX (OPerational EXPenses) have to be considered, even though the latter is more difficult to see in advance. Coupling these with the administrative requirements of the operator and physical limitations of the technology leads to a really complex optimization problem. The current practice for access network planning could be described as a very time consuming “design guesswork” of highly experienced engineers, which is probably not an optimal use of the highly valuable human creativity.

My research is devoted to strategic topology design algorithms, which on the one hand could speed up the design process, and on the other hand, the mathematic interpretation allows evaluation of the network topology regarding optimality. The term “strategic topology design” stands for a high-level design, including location of the network elements, layout of the optical cable plant and a complete system design, but lacks the details of a low-level design. I have to shortly mention here that digital maps and GIS (Geographic Information System) databases are an “enabling technology” for the algorithmic, computer aided network design, solving the most important practical difficulty. Digital maps became available in the recent years for typically any access network service area, and as we will see, this geographic information serves as the primary input data of the optimization problem.

Novelty of the results presented in this dissertation lies in the fact that I am solving a problem algorithmically by computers, which was earlier done by a human guesswork, and no scalable algorithmic solution was known that was capable to handle problem sizes of practical interest.
1.2.1. **Techno-economic Evaluation**

As an important additional application, such a strategic topology design supports detailed preliminary cost estimation, and offers thorough techno-economic evaluation and comparison of the NGA technologies in focus. Techno-economic evaluation addresses the relationship between technical decisions and their economic impact: optical access networks clearly have technical advantages over the legacy copper networks, but the investment must be justified by economic considerations.

The first and foremost step of any economic evaluation is the estimation of the network deployment cost. The existence of an algorithmic network design methodology leads to a significant improvement in this field: knowing the optimized topology itself, helps to calculate the necessary expenses. Therefore availability of a topology design methodology for all viable NGA technologies supports the optimal choice among suitable network architectures.

The novelty of my approach is the integration and exploitation of network design in the techno-economic evaluation. The “state of the art” techno-economic methodologies are using simplified geometric models for cost estimation, instead of the optimized network topology itself. As our recent results have shown, the concept of integrating network design into techno-economic evaluation is a significantly more accurate and reliable methodology [52]-[53].

1.3. **Related Work**

Theory of network design in itself has a long history and a massive research background [26]. The above described high economic and technical impact brought wide audience to network design for optical access networks. However, algorithmic network design was not possible in the absence of digital maps and GIS databases, and the computational capacity also set tight restrictions for a long time. The recent advances in this field made it a viable opportunity: at the time of writing this dissertation there were some initial results in the literature, however all of them were constrained by the contradictory requirements of scalability and the need to avoid oversimplification.

S. U. Khan from the University of Texas (US) performed pioneer work in the field of algorithmic network design for PON networks [28] in 2005, even if his work was focusing on a Manhattan grid topology, which is a simplified approach to network design in general [29]. B. Lakic and M. Hajduczenia from Nokia-Siemens Networks Portugal have investigated the possibility to apply k-means clustering for demand points in the Euclidean space (neglecting the street system graph), and then genetic algorithm for path generation [30]. Despite the promising results, not considering the street system at all is an example of serious oversimplification: the access network must typically follow the cable paths along streets, not crossing e.g. private buildings and properties, or rivers for example.

E. Bonsma et al. from British Telecom have investigated an Evolutionary Algorithm approach, which had scalability problems, restricting its use to small schematic sample networks of a hundred demand points [31]. Another approach for use of genetic algorithm was published in [32] by A. Kokangul and A. Ari from Turkish Telecommunication Company, with similar scalability problems, just as the genetic algorithm solution published by K. F. Poon et al. [33]. A. Haidine, R. Zhao et al. published another metaheuristic technique, namely particle swarm optimization for the demand point clustering (partitioning) problem of a VDSL access network with feeder fiber [34].

Exact optimization, such as the highly complex Mixed Integer Programming (MIP) approach was demonstrated in [35] by S. Chamberland from École Polytechnique de Montréal, which promises high quality results, but suffers from serious scalability problems: the large number of defined constraints and the dimensions of the resulting MIP matrix limits its usable range for hundreds of demand points.
Later, concurrently with my research, similar research activities have started at University of Melbourne. Initially J. Li and G. Shen published in [36] a two-level random restart iterative heuristic algorithm considering geographic constraints in 2008. The published results were promising, therefore I have implemented their RALA algorithm (Recursive Allocation and Location Algorithm) for comparison, and I have found that it gives almost the same results as my methods (within 1-2%); however its time consumption is 2-3 orders of magnitude higher due to the numerous iteration steps. Later on, in [37] they presented another recursive method for “greenfield”, rather infrastructureless network planning, not considering any existing infrastructure, street system or geographic constraints at all.

Clustering methods, and in particular k-means algorithm was applied in various other publications for creating the demand point groups in point-multipoint networks. A. Agata and Y. Horiuchi from KDDI R&D Labs (Japan) have published a method using Voronoi-diagrams for splitting the service area, and then k-means for clustering. It is a highly effective solution, even if it does not solve the “bin packing” problem of PON splitters [38]: k-means is not designed for clustering problems with fixed size clusters.

NGA network planning has impressive literature addressing other aspects, e.g. marketing and economic considerations, or predicting future cost and demand parameters, however these do not belong closely to the scope of this dissertation, therefore I do not present them in details. An overall view on business and economic aspects is given in [41] by FTTH Council Europe.

Techno-economic evaluation of access networks addresses the tradeoff between technical superiority and economic viability. Typically a higher investment results in higher service quality, however the relationship is far from linear. A thorough investigation of options, considering costs, service requirements and viable technologies is necessary to find the most suitable solution for the given service area. This interdisciplinary research area covering communication technology and economy is referred to as techno-economics. We have given an overall description of a techno-economic evaluation framework in [42]. A nice techno-economic overview of European telecommunication investment options is outlined in [43] by B. T. Olsen (Telenor) et al.

Every techno-economic methodology focuses on the primary question of the initial investment. Two fundamental approaches exist for estimating the cost of network deployment: in cases where sufficient statistical data exists, the typical cost per customer multiplied by the number of customer premises gives an acceptable estimation, as the work of J. L. Riding, J. C. Ellershaw et al. [47] shows. Without such statistical data, topology of the network infrastructure has to be estimated, and the approximate values for fiber lengths and network equipment have to be summarized. Since statistical data obviously does not exist for new network technologies, modeling the network topology modeling becomes the only viable solution in this case.

The generally used geometric models are attractive due to their simplicity: using a few descriptors of the service area (e.g. diameter, population or density), a regular triangle or square network model is built, and with basic trigonometric and geometric formula, dimensions of the resulting cable plant and network equipment is derived. Such geometric models were developed by a sequence of EU research projects, e.g. TITAN, OPTIMUM, TERA or TONIC [48], and we also find recent results in [49] - [50].

As our recent results and publications have proven, this very attractive simplicity of geometric models reduces reliability and accuracy of the estimations (initial results in [51], short comparison in [52], in-depth comparison in [53]). It shows a really important field for algorithmic network design: the ability to create the specific network topology for the chosen service area will definitely lead to higher accuracy for network deployment cost estimation than using a geometric model.
1.4. RESEARCH OBJECTIVES

The initial research objective is to achieve a deep understanding of the problem, and to clarify its theoretic background. A fundamental principle of the mathematic interpretation is to avoid oversimplification that hides important practical aspects of the problem. Therefore accurate and realistic network and cost models are necessary, that allow technology agnostic, general theoretic discussion at the same time. Difficulty of the formal modeling lies in the conflicting requirements of general, theoretic approach and practical applications of the proposed methodology.

The model should be able to handle different current and future NGA architectures and technologies, at the necessary abstraction level for theoretic research. Therefore a general formulation and model for the NGA Topology Design (NTD) problem will be given, followed by the definition of its special cases for Passive Optical Networks (PON), Active Optical Networks (AON), Digital Subscriber Line (DSL) and Point-to-Point (P2P) networks, according to the above presented NGA technology classification. Based on the model and the formal representation, the mathematical problem should be analyzed, from the point of view of complexity and approximability [54].

Modeling, formulation and analysis supports the efforts towards an optimization methodology, which is as-fast-as-possible and as-good-as-possible at the same time. Normally methods with polynomial time complexity are accepted as fast algorithms. However, a large family of optimization problems, as the addressed NGA Topology Design (NTD) problem belongs to NP-hard problems, therefore “as-good-as-possible” in this case refers to an approximation instead of exact optimization.

A fundamental requirement is to develop methods that are scalable enough for real-world scenarios, i.e. large-scale topology design problems, with up to thousands or even tens of thousands of demand points. These large-scale problems have to be solved within reasonable time, even though reasonable time for the offline problem of topology design practically means it has to be solved, regardless of time. However, as we will see later, it is still a hard challenge, due to complexity reasons.

Finally, for evaluation purposes, and in order to provide a benchmark for the proposed scalable methodology, reference methods are necessary. These will be built on generally accepted concepts of optimization, e.g. quadratic/linear programming or widely known metaheuristic approaches.

In summary, goals of the research presented in this dissertation are the following:

- modeling and formulation
- analysis of the problem to be solved
- proposing solutions, solving the optimization problem
- evaluation of the proposed solutions

1.5. RESEARCH METHODOLOGY

Since the work described in this dissertation is focusing on algorithmic topology design for NGA networks, including the theoretic background, proposed solutions and their evaluation, the initial step will be the introduction of a formal graph model and the optimization problem formulation, including its constraints, objectives, parameters and special cases.

Once the problem is formulated, a complexity and approximability analysis will be carried out, providing in-depth knowledge on the most significant components and underlying subproblems of the NGA topology design problem. Tools of graph theory and algorithm theory will be applied in the sections devoted to these issues, and linear reductions will be constructed for complexity and
approximability analysis of the problems. The algorithmic analysis should provide reasonable requirements regarding scalability and accuracy of the heuristics.

The proposed methodology is using highly specialized **heuristic** algorithms, since the general optimization techniques did not meet the requirements for scalability and accuracy. **Decomposition** helps to separate the subproblems and handle the strong cross-dependence among them. For exact optimization, a **quadratic programming** formulation is given, and by linearization and a notable transformation, dimensions of the **linear programming** formulation reduced, in order to find lower bounds for numerical evaluation. Finally, a **simulated annealing** scheme will be presented.

### 1.6. Outline of the Dissertation

The presented results and findings have a two dimensional structure, which makes the necessary “linearization” for a written dissertation difficult. The modeling and formal representation for the NGA Topology Design (NTD) problem in general is described in Chapter 2, which also contains the definition of its special cases for PON, AON, DSL and P2P networks. These special cases add the second dimension to the structure of the dissertation: the modeling, analysis, proposed algorithms and evaluation chapters follow these special cases.

The general problem and the special cases are investigated in Chapter 3 from the aspect of their complexity and approximability features. Chapter 4 is devoted to the graph-theoretic investigation of the modeling graph itself, which leads to the notion of criticality and identification of the critical nodes in the graph. Chapter 5 plays central role, since the proposed, highly efficient topology optimization algorithms are presented in this section, along with the algorithmic analysis of the proposed solutions. In Chapter 6, the “reference solutions” are introduced: (1) the quadratic and linear programming approach promising exact optimum, at least for smaller problems; and (2) a simulated annealing scheme, used as benchmark for the specialized heuristics. Finally, Chapter 7 presents the numerical results, validation and evaluation. Conclusions are drawn in Chapter 0, which also presents briefly the application possibilities and references.

### TABLE 1 OUTLINE OF THE DISSERTATION

<table>
<thead>
<tr>
<th>Chapter / section topics of the dissertation</th>
<th>Formulation</th>
<th>Complexity analysis</th>
<th>Approximability analysis</th>
<th>Efficient heuristics</th>
<th>Reference methods</th>
<th>Validation</th>
<th>Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>NTD in general</td>
<td>2.4</td>
<td>3.1</td>
<td>3.2</td>
<td>5.3</td>
<td>6</td>
<td>7.1</td>
<td>7.2</td>
</tr>
<tr>
<td>PON</td>
<td>2.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AON</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P2P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2. THE TOPOLOGY DESIGN PROBLEM

The introductory section outlines the topic and the wider area, which is addressed by the research described in the dissertation. This section is devoted to the formal and detailed description of the addressed, investigated and later solved problem, the NGA Topology Design (NTD) problem, as it will be called throughout the dissertation.

The work described in the dissertation is focusing clarifying the theoretic background of algorithmic topology design (optimization) for NGA networks, and then, based on the theoretic work, a methodology will be proposed for it that fulfills the necessary requirements for scalability and accuracy. In order to carry out mathematical analysis, the problem has to be formulated first. A well-defined model and formal representation is a prerequisite of the comprehensive complexity and approximability analysis, and also supports identification of subproblems and key points.

Therefore in the first subsection the formal model is described, that has to be as realistic as possible, representing all decisive characteristics of the NTD problem. The problem will be then formulated as a standard optimization problem, defined by the solution space, the model (variables), the objective and the constraints in the second subsection. Finally the most important special cases are defined, based on the classification of various present and future NGA technologies.

2.1. PARAMETERIZED GRAPH MODEL

Network designers are facing really diverse problems for various NGA technologies; however by finding the proper level of abstraction these problems can be represented by the same formal model. The applied network graph model is intended to represent all the significant information for the topology design process: the geographic and infrastructural data of the area where the network will be deployed, the technology specific constraints and the cost parameters. These are explained in the following paragraphs.

Typically an access network topology consists of a Central Office (CO), a set of Subscriber Units (SU), and a cable plant connecting the demand points (households/subscribers) to the central office. In a point-multipoint structure, these demand points are organized into groups, and demand points within a single group share a Distribution Unit (DU), which aggregates their traffic towards the CO. The network segment interconnecting the DUs and the CO are referred to as feeder network segment, while the distribution network connects the demand points (households) to the DUs. Schematic view of such a point-multipoint access network is depicted on Figure 2.

The service area itself, where the network is to be installed, may have many attributes that are important for various stages of the network deployment process. Focusing on the topology design itself, traces or paths along which network connections may be realized represent the most important information. Typically network links cannot be built wherever desired, they must follow existing cable paths, the street system or other infrastructure: the cable deployment and civil works (trenching, digging) is allowed in publicly owned land or on the existing infrastructure. The set of these so-called “available network links” (where cables may be deployed) serves as the edges of the graph model.

Several important nodes are to be identified in the graph, with respect to their role, e.g. demand points, location of the central office, or the set of locations where the distribution units of a point-multipoint network may be installed (e.g. cabinets with power supply, manholes, etc). The latter set will be referred to as “available DU locations”.

| The topology design problem | 7 |
Therefore, the graph model of an NGA topology design (NTD) problem contains the map of the service area (with the available network links), the demand point nodes, the CO location and potential locations of DUs, represented by the following elements as its nodes and edges (Figure 3):

- **nodes** \( (V) \)
  - street crossings and internal nodes of the street system
  - available DU locations \( (\Omega) \)
  - position of the Central Office (CO)
  - demand point locations \( (S) \)
  - connection points of the drop cables
- **edges** \( (E) \) interconnecting these nodes, i.e.
  - edges of the street system graph (not only between neighboring street crossing, but e.g. the curved street segments are composed of a sequence of edges in most digital maps)
  - drop cables, connecting the demand points to the street
  - edges connecting the CO or the DU locations to the street system graph

These altogether define a network graph \( G = (V,E) \), the set of DU locations \( \Omega = \{DU_i\} \subseteq V \) and the set of demand points (potential subscribers) \( S = \{s_i\} \subseteq V \). The set of graph edges represent all the potential cable/fiber routes possibilities, i.e. the access network topology must follow these edges to connect the demand points to the DUs, and the DUs to the CO.

The “virtual topology” of the access network connects the demand points to DUs (located at available DU locations), and the DUs to the CO. The physical network topology is the realization of these connections as “walks” along the edges of the graph model (examples are shown with green dashed lines on the figure).
The process of creating the modeling graph is explained on an example. The graph model of the schematic network of Figure 2 is given on Figure 4/A. A set of parameters is assigned to these network elements, e.g., length and cost of edges, location (coordinates) of demand points, or capacity of distribution unit locations. By these parameters and properties of nodes and edges, the various network elements, and the role of nodes or edges in the network graph model is identified. Figure 4/B shows the parameterized network graph model, prepared for calculations, while on Figure 4/C, a respective network topology is given, with DU locations, demand point groups and the set of actually used network links.

Obviously the model is flexible in the sense that the optional parameters may be defined and assigned to graph elements. Such flexibility allows the use of any specific network inventory and GIS database, while the clear schematic graph structure keeps the network modeling straightforward.
2.2. COST FUNCTION (EXPENDITURES)

The necessary investment is composed of capital expenditures (CAPEX), i.e. the cost of deploying the network; and operational expenditures (OPEX) for maintenance and operation costs [44]. Beyond this, both CAPEX and OPEX may be further divided into topology dependent and topology independent components. E.g. fiber costs obviously depend on the topology, while rental of the CO building does not. Table 2 and the following paragraphs give an overview of the decisive network deployment cost factors, their significance and topology dependency. Since the dissertation is devoted to topology design and optimization, the focus will be on topology-dependent network deployment costs (CAPEX).

2.2.1. NETWORK EQUIPMENT

The central office (CO) serves as the interconnection between the access network and higher network segments (e.g. metro / core network). Active equipment is used in the CO that requires power supply and cooling. Their price may be significant, and partially topology-dependent: the more distribution units (Splitter, Switch or DSLAM) are deployed, the more complex central office infrastructure is necessary.

Demand points are connected to the CO via distribution units (DUs). In PON networks, the demand points are served through relatively cheap passive optical splitters, compared to the active DUs used in AON and DSL networks that are expensive and require power supply (increasing OPEX). The number of necessary distribution units depends on the structure of the point-multipoint topology.

Subscriber units (SU) serve as the termination of the access network at the demand points. The contribution of these to the total cost is determined by the number of demand points, thus independent from the network topology. Formally, the equipment cost is composed of the central office (CO), the distribution unit (DU) and the subscriber unit (SU) costs:

\[ C_{\text{Equipment}} = C_{\text{CO}} + \sum_{\text{Subscribers}} C_{\text{SU}} + \sum_{\text{Groups}} C_{\text{DU}} \quad (1) \]

2.2.2. CABLE PLANT

The cost of the cable plant has two fundamental components: the labor cost of deploying the cables, and the raw material cost of the cables deployed. Cable deployment is typically the most significant among all cost factors, and obviously both are heavily topology-dependent. Deployment cost may be further detailed: (1) trenching and (2) installing a cable in an existing cable duct (installation cost, \( C_0 \)) is independent from the amount of fibers to be deployed, while a smaller, additive cost represents the price of each fiber itself (fiber cost, \( C_v \)).

Co-existence of these components results in a stepwise cost function, as depicted on Figure 5. Difference between labor and installation costs may be overwhelming if complete trenching (digging) is necessary, but moderate, if existing cable ducts are used. More formally, by summing up the cost of deployment and fiber:

\[ C_{\text{cable plant}} = C_{\text{deployment}} + C_{\text{fiber}} = \sum_{e : F(e) > 0} \left( C_0(e) + \sum_{e} [F(e) \cdot C_v(e)] \right) \quad (2) \]

Here \( F(e) \) is an auxiliary function indicating the installed capacity upon link \( e \) (obviously the labor cost is to be paid only for installed network links). The trenching and installation costs are aggregated into \( C_0(e) \) cable deployment cost. The fiber costs on link \( e \) are represented by \( C_v(e) \).
This cost function, with possibly different values for every network link, or group of network links offers the possibility to include infrastructure information into the network model: various cable deployment costs can be assigned to various areas, subject to cabling technology or civil work required. E.g. existing cable ducts may be re-used with \( C_0(e) = 0 \) labor cost, or aerial cables may be cheaper to install than trenching. The cost function hence allows the consideration of infrastructure information, and the cost decreasing (increasing) effect of existing (missing) network elements, which again extends the flexibility of the model, but preserves its formal and abstract nature.

\[
C_{\text{Total}} = C_{\text{CO}} + \sum_{\text{Subscribers}} C_{\text{SU}} + \sum_{\text{Groups}} C_{\text{DU}} + \sum_{e:F(e) > 0} (C_0(e) + \sum_{F(e)} C_v(e)) \tag{3}
\]

Namely the total cost is the sum of the \{Central Office costs\} + \{Subscriber Unit costs\} + \{Distribution Unit costs\} + \{Cable plant costs\}. Considering the optimization problem, these costs are divided into topology dependent and topology independent costs. Note that topology independent costs are constants in the optimization problem itself, but these will play an important role during the cost estimation and techno-economic evaluation later.

Regarding equipment, summarized cost of subscriber units (\( \sum C_{\text{SU}} \)) does not depend on the topology (only on the number of demand points), therefore these may be removed from the optimization objective as constants. Cost contribution of the distribution units (\( \sum C_{\text{DU}} \)) increases with the number of demand point groups. Cost of the central office (\( C_{\text{CO}} \)) typically depends on the number of demand points (topology-independent, constant), and also on the number of demand point groups. The latter component will be merged with the cost of the distribution units (\( C_{\text{DU}} \)) into a new variable for this combined cost (\( C_{\text{DU}}^* \)). Therefore the optimization problem reduces to:

\[
C_{\text{Topology dependent}} = \sum_{\text{Groups}} C_{\text{DU}}^* + \sum_{e:F(e) > 0} (C_0(e) + \sum_{F(e)} C_v(e))
\]

Table 2 summarizes the main cost factors, indicates their topology dependency, and also their significance or weight relative to other cost factors. The latter needs further explanation, which is given in Section 2.6 – it naturally follows the characteristics of the respective network technologies, e.g. active network elements have higher cost than passive ones.
### TABLE 2 SUMMARY OF COST COMPONENTS

<table>
<thead>
<tr>
<th>Cost component</th>
<th>Topology dependency</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Network equipment</strong></td>
<td>Central Office</td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td>Distribution Unit</td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td>Subscriber Unit</td>
<td>None</td>
</tr>
<tr>
<td><strong>Cable plant</strong></td>
<td>Labor cost (trenching)</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>Cable installation</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>Fiber</td>
<td>High</td>
</tr>
</tbody>
</table>

### 2.3. TECHNOLGY DEPENDENT CONSTRAINTS

Once the graph model and its parameters are defined, additional rules constraining the set of “valid” topologies may be specified. The discussed NGA network technologies, according to their physical capabilities, set different constraints on the topology. Two primary limitations have to be considered, namely network range and capacity of the distribution units.

Network range leads to various network deployment rules for every technology, in some cases the overall CO-demand point distance ($L_{\text{max}}$), in other cases the length of the feeder ($L_{\text{feed}}$) or distribution network segment ($L_{\text{dist}}$) is constrained. The distribution unit capacity constraints ($K$) are similar for all technologies, at least on the abstraction level: the maximal splitting ratio of an optical splitter or the switching capacity of an active distribution unit has to be considered.

During the topology design process, not only the previously mentioned costs are to be minimized, but also these constraints are to be fulfilled, which determine the set of valid topologies. In the next paragraphs, the specific constraints are discussed for each type of NGA technologies.

Passive Optical Networks have network range for the complete access network segment, i.e. the demand point-CO distances are limited ($L_{\text{max}}$). This $L_{\text{max}}$ value depends on the optical splitting ratio. Obviously, the maximal split ratio of the optical splitter ($K$) is also bounded. Typical values for recent GPON systems are 20-60 km network range with 1:64 splitting ratio [14].

Active Optical Networks (e.g. Active Ethernet systems) are constrained by the capacity of the switching unit ($K$), by the range of the feeder ($L_{\text{feed}}$) and distribution network segments ($L_{\text{dist}}$) respectively [17].

These constraints are more strict for DSL networks, since the length of the copper loop (i.e. the distribution segment, $L_{\text{dist}}$) decisively affects the available bandwidth. For recent VDSL networks, broadband access limits the DSLAM-demand point distance typically in the range of 300-1000 m [23]. Capacity of the DSLAM ($K$) also constrains the topology.

The point-to-point dedicated fiber architecture is only limited by the length of the optical fiber ($L_{\text{max}}$). Furthermore, even this single constraint is a loose constraint: it may cover sufficiently long distances in practice.
Table 3 concludes these constraints. I note that we are aiming at a higher level of abstraction in the modeling and problem formulation, in order to keep distance from present NGA technology specifications. These constraints offer realistic restrictions for valid NGA topologies. Flexibility of the graph model and its parameters supports virtually any additional physical or administrative limitations to the optimization problem as specific additional constraints.

### TABLE 3 TOPOLOGY DEPENDENT CONSTRAINTS

<table>
<thead>
<tr>
<th>Network technology</th>
<th>Network range (segments)</th>
<th>Distribution unit capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Passive Optical Networks (PON)</td>
<td>$L_{\text{max}}$</td>
<td>$K$</td>
</tr>
<tr>
<td>Active Optical Networks (AON)</td>
<td>$L_{\text{feed}}$</td>
<td>$L_{\text{dist}}$</td>
</tr>
<tr>
<td>Digital Subscriber Line (DSL)</td>
<td>$L_{\text{max}}$</td>
<td>$K$</td>
</tr>
<tr>
<td>Point-to Point Fiber (P2P)</td>
<td>$L_{\text{max}}$</td>
<td>-</td>
</tr>
</tbody>
</table>

2.4. **OPTIMIZATION PROBLEM FORMULATION**

With the above described representation, the NTD problem can be interpreted as a (minimal cost) subgraph of $G$, that connects the demand points to the CO in a point-multipoint (or point-to-point) architecture through a subset of network links, using several properly located distribution units, fulfilling all the DU capacity, connection length and network capacity constraints.

Formally, we are given a network graph $G = (V, E)$, consisting of edges $E$ and nodes $V$ representing the traces/paths, along which a network link could be built, the set of available DU locations $\Omega = \{DU_j\} \subseteq V$ and the set of demand points $S = \{s_i\} \subseteq V$.

All the edges $e \in E$ have a nonnegative length $l(e)$, a cable deployment cost $C_0(e)$ and fiber cost $C_\mu(e)$. These costs are typically but not necessarily proportional to the length of the link, and the cost reflects the different cabling technologies and existing infrastructure conditions. The cost of deploying a distribution unit (DU) is $C_{DU}$.

**CONSTANTS**

\[
\forall e \in E \quad C_0(e) \quad \text{Cost of cable deployment on link } e
\]

\[
\forall e \in E \quad C_\mu(e) \quad \text{Cost of fiber on link } e
\]

\[
\forall e \in E \quad l(e) \quad \text{Length of link } e
\]

\[
C_{DU} \quad \text{Cost of Distribution Unit (DU)s}
\]

**VARIABLES**

\[
\forall i \in S, \forall j \in \Omega \quad \omega_i^j \in (0, 1) \quad \text{Indicator of the demand point-DU location assignment, value is 1 only if the demand point } i \text{ is connected to DU location } j.
\]

\[
\forall j \in \Omega \quad n_j \in \mathbb{N} \quad \text{The number of DUs at location } j.
\]
∀i ∈ S, ∀e ∈ E \ x_e^i \in (0,1) \ \text{Indicator of edge e on the path between demand point i and its assigned DU location (Distribution network)}

∀j ∈ Ω, ∀e ∈ E \ y_e^j \in (0,1) \ \text{Indicator of edge e on the path between DU_j and the CO (Feeder network)}

∀e ∈ E \ F(e) \ \text{Number of connections over edge e in the feeder and in the distribution networks altogether}

∀e ∈ E \ I_e \in \{0,1\} \ \text{Indicator variable for existence of edge e either in the feeder or in the distribution network segments}

N \ \text{Total number of DUs deployed in the network}

OBJECTIVE

minimize \ C_{\text{topology dependent}} = N \cdot C_{DU} + \sum_e \left( I_e \cdot C_0(e) + F(e) \cdot C_v(e) \right)

CONSTRAINTS

(1) \ ∀i \in S \ \sum_{j \in \Omega} \omega_j^i = 1

(2) \ ∀j \in \Omega \ \sum_{i \in S} \omega_j^i \leq n_j \cdot K

(3) \ ∀v \in V, ∀i \in S \ \sum_{e : v \to i} x_e^i - \sum_{e : i \to v} x_e^i = \begin{cases} \omega_j^i & v \in \Omega \\ -1 & v = i \\ 0 & \text{otherwise} \end{cases}

(4) \ ∀v \in V, ∀j \in \Omega \ \sum_{e : v \to j} y_e^j - \sum_{e : j \to v} y_e^j = \begin{cases} -n_j & v = DU_j \\ 0 & \text{otherwise} \\ +n_j & v = CO \end{cases}

(5) \ ∀i \in S \ \sum_{e \in E} l(e) \cdot x_e^i + \sum_{j \in \Omega} \left( \omega_j^i \cdot \sum_{e \in E} l(e) \cdot y_e^j \right) \leq L_{\text{max}}

(5a) \ ∀i \in S \ \sum_{e \in E} l(e) \cdot x_e^i \leq L_{\text{dist \ max}}

(5b) \ ∀i \in S \ \sum_{j \in \Omega} \left( \omega_j^i \cdot \sum_{e \in E} l(e) \cdot y_e^j \right) \leq L_{\text{feed \ max}}

(6) \ ∀e \in E \ F(e) = \sum_{i \in S} x_e^i + \sum_{j \in \Omega} y_e^j

(7) \ ∀e \in E \ F(e) \leq I_e \cdot (S + \Omega)

(8) \ N = \sum_{j \in \Omega} n_j
Constraints (1) ensure that every demand point is served by exactly one DU, (2) represents the capacity constraints for DUs: the number of demand points connected to a DU location is bounded from above by the summarized total capacity of the DUs located at the given location. The flow conservation (Kirchhoff) constraints (3) and (4) keep the flow from splitting. These ensure that every demand point has a dedicated unitary flow towards its DU (distribution network), and every DU locations are connected to the CO by a flow of $n_i$ units (feeder network). Constraints (5), (5a) and (5b) provide the network reach (distance) limits for the overall, feeder and distribution network segments, respectively. I note that constraints (5) and (5b) make the formulation quadratic. Finally, constraints (6), (7) and (8) provide the auxiliary data for the cost function: the $F(e)$ installed fiber count values for every link, the indicator variable for link deployment and the number of DUs.

This Quadratic Programming (QP) formulation, its linearization and relaxation opportunities are discussed in Chapter 6.1. Clearly, the novelty of my approach and model is not the existence of a graph model in itself, but as it was concluded in the introductory on related work (Section 1.3), such detailed and realistic network model, optimization objective and constraints, including all important characteristics of an NGA network deployment, i.e. cable plant and equipment costs, network infrastructure and the map of the service area, was not addressed earlier.

2.5. Impact of Cost Components and Constraints

The optimization problem has its objective (cost function) that differentiates between a “good” and a “bad” solution; and its constraints that differentiate between valid and invalid solutions. These cost components, and the various constraints have various effects on the optimal solution. Certain weights of each cost component, or significance and tightness of a constraint may emphasize different characters of the optimization problem. The classification of technology-specific special cases, and the efficiency of the later proposed, specialized heuristics is a consequence of understanding the impact of these cost components and constraints. Therefore, in this section, I briefly review the cost factors and constraints, and highlight the effect of different typical values.

Cost Components

**DU costs ($C_{DU}$):** high distribution unit costs require maximal DU utilization, therefore all the $K$ capacity of them has to be fulfilled. Especially for large $K$ values, it enforces large distribution unit segments. In contrary, DU costs substantially lower than cable plant costs result in simpler distribution unit segments, i.e. small star topologies around the DUs.

The subproblem of formulating groups of demand points around distribution units may be interpreted as a graph clustering problem, with a significant specialty: the cluster sizes are bounded from above by the $K$ capacity value. The higher the DU costs are, the closer the cluster sizes should be to this $K$ value. The problem of making almost equal-size segments makes a significant difference between traditional clustering problems, and the group formulation subproblem.

**Deployment costs ($C_{dep}$):** High deployment costs have a well noticeable effect on the optimal solution. Minimizing length of the traces above fiber lengths leads to a clear Steiner-tree problem, as it will be further detailed at the complexity studies (chapter 3.1).

**Fiber costs ($C_{fib}$):** Fiber costs, in contrary to deployment costs enforce topologies more like a shortest path tree instead of Steiner trees, where all nodes are connected to the CO along their shortest paths, even if it increases overall trace length.
Access network topologies typically have a tree structure, but these two cost factors make
the difference between the two extremities, i.e. trace minimizing Steiner trees, or distance minimizing
shortest path trees.

**CONSTRAINTS**

**DU capacity ($K$):** The effect of DU capacity is not independent from the DU costs, but high DU cost
makes the capacity constraints significant. High capacity DUs result in really wide groups and long
distribution network segments, that may conflict with distance limitations (see below), and also
increases cable plant costs. On the other hand, low DU capacities lead to really small demand point
groups. If the DU capacity is not a magnitude higher than the typical building size, an additional bin
packing problem arises, since granularity issues make filling DUs difficult. As an extreme example,
splitter with capacity for 64 demand points may be utilized only at around 50% with buildings having
33 demand points.

**Network range ($L_{\text{max}}$):** This constraint has a straightforward consequence on the graph. The
network range is the diameter of the feasible network topology around the CO; any demand points
outside this region are excluded from the valid solutions.

**Feeder network range ($L_{\text{feed}}$):** The shorter the feeder network segments are constrained, the
closer the DUs are to the CO. As an extremity, point-multipoint topologies become point-to-point
topologies with zero length feeder network segments, however assuming short feeder, and long
distribution network range is not realistic.

**Distribution network range ($L_{\text{dist}}$):** As the most interesting and complicated element of
the constraint set, the distribution network range, and its consequences will be investigated later: Section
4 is dedicated to this topic, including several observations for short distribution network segment
constraints.

2.6. **SPECIAL CASES**

The optimization problem was defined by its variables, objective function and constraints in the
previous section. In its general form it covers all the discussed NGA technology types: this general
approach is beneficial for theoretic modeling and analysis of the problem.

On the other hand, a classification of various present and future access network technologies is
possible. Among the permanently evolving network technologies and standards different families are
noticeable. Based on the different concepts driving these developments, a primary classification
differentiates between Passive Optical Networks (PON), Active Optical Networks (AON), and DSL
network in the field of point-multipoint access networks, and a completely different approach of Point
to Point (P2P) optical networks, offering a dedicated fiber infrastructure for every demand point.

Without losing the necessary abstraction level or committing ourselves to a specific recent standard,
these classes of present and future NGA network technologies will be described as special cases of the
optimization problem formulated in the previous section, by identifying the most prioritized parts of
the objective (cost) function or the constraint set. This classification, outlined in the following
subsections is a significant contribution towards highly efficient, specialized optimization algorithms,
therefore serves as a fundamental part of the dissertation.
In a passive optical network, the distribution unit equipment is passive (e.g. a power splitter in TDM PONs or a wavelength switch in WDM PONs), hence the feeder and distribution network segments are in the same optical domain, without signal regeneration at the DU. Therefore the network range limitations stand for the complete optical network segment, i.e. between the CO and the SUs.

However, due to beneficial characteristics of the optical fiber, these network range limitations are fairly permissive for fully optical access networks compared to other access network technologies, e.g. DSL networks. On the other hand, these are not negligible, especially due to the power splitters of passive optical networks: the attenuation depends on the splitting ratio. High capacity distribution units decrease the available network reach.

Capacity constraints of the optical splitters have more significant effect on the topology, in passive optical networks the DU capacities are typically lower than for their active counterparts. Primarily these constraints are affecting the set of valid topologies.

Regarding topology dependent network deployment cost, fiber deployment in the distribution network, between every demand point and its corresponding distribution unit (splitter) dominates over the relatively cheap, passive optical splitters, and the less significant feeder network. Moreover, cable plant costs may be further refined. Typically every demand point has to be connected to the access network, thus fiber installation is necessary along the whole street system. Hence cable deployment costs are more or less topology independent. Therefore, during the optimization process, cable plant costs will be incorporated in a single link cost $C^*_v(e)$ which stands for the fiber cost and increased by the respective installation costs.

More formally, in this special case the connection length constraints are relaxed, the $C_0(e)$ and $C_v(e)$ values are substituted by increased $C^*_v(e)$ values, while the $C_{DU}$ distribution unit costs are still considered. This way we get a challenging two-component optimization problem, where the cost function is a combination of two components: DU and cable plant costs. These altogether result in a slightly simplified special case of the NTD problem:
OBJECTIVE:

\[
\text{minimize } C_{\text{topology dependent}} = N \cdot C_{\text{PU}} + \sum_{F(e)} C_{\text{o}}(e)
\]

CONSTRAINTS:

(1) \quad \forall i \in S \quad \sum_{j \in \Omega} \omega_{ij} = 1

(2) \quad \forall j \in \Omega \quad \sum_{i \in S} \omega_{ij} \leq n_j \cdot K

(3) \quad \forall e \in E, \forall i \in S \quad \sum_{e \rightarrow v} x^i_e - \sum_{e \rightarrow v} x^j_e = \begin{cases} \omega_{ij} & v \in \Omega \\ -1 & v = i \\ 0 & \text{otherwise} \end{cases}

(4) \quad \forall v \in V, \forall j \in \Omega \quad \sum_{e \rightarrow v} y^j_e - \sum_{e \rightarrow v} y^i_e = \begin{cases} -1 & v = SP_j \\ 0 & \text{otherwise} \\ +1 & v = CO \end{cases}

(5) \quad \forall i \in S \quad \sum_{e \in E} l(e) \cdot x^i_e + \sum_{j \in \Omega} \left( \omega_{ij} \cdot \sum_{e \in E} l(e) \cdot y^j_e \right) \leq L_{\text{max}} \quad \text{relaxed}

(6) \quad \forall e \in E \quad F(e) = \sum_{i \in S} x^i_e + \sum_{j \in \Omega} y^j_e

(7) \quad N = \sum_{j \in \Omega}

2.6.2. SPECIAL CASE #2 (ACTIVE OPTICAL NETWORKS, AON)

FIGURE 7 ACTIVE OPTICAL NETWORK (AON)
Due to the fully optical cable plant infrastructure, the network range limitations are really permissive, even more then for passive optical networks, since the active distribution units in this case do not increase attenuation, rather make signal regeneration. Moreover, the active equipment divides the optical layer in two, resulting in distinct network range limitations for the feeder and distribution network segments.

As stated earlier, capacity of the active distribution units typically exceeds that of the passive equipment. These altogether move the focus mostly on the cost function, since the constraints permit a considerably wide set of valid topologies.

On the other hand, the active distribution units, compared to the passive ones require significantly higher investment, especially if also the operation and maintenance costs are considered, due to the necessary power and cooling supply, and also due to the higher risk of breakdowns. The cable plant costs, particularly the distribution network has also noticeable contribution to the topology dependent network costs.

According to the problem formulation, in this case the length constraints are relaxed, the DU capacity constraints are kept. The special case is mostly characterized by the cost function: the overwhelming DU costs \( C_{DU} \), and also the lower distribution network costs are considered, primarily through the fiber \( C_F \) costs, as described for PON networks.

The formal representation of the special optimization problem:

**OBJECTIVE:**

\[
\text{minimize } C_{\text{topology dependent}} = N \cdot C_{DU} + \sum_{e \in E} C_F(e)
\]

**CONSTRAINTS:**

1. \( \forall i \in S \sum_{j \in \Omega} \omega_i^j = 1 \)
2. \( \forall j \in \Omega \sum_{i \in S} \omega_i^j \leq n_j \cdot K \)  
   [low significance]
3. \( \forall v \in V, \forall i \in S \sum_{e \in v} x_i^e - \sum_{e \rightarrow v} x_i^e = \begin{cases} 
\omega_i^v & v \in \Omega \\
-1 & v = i \\
0 & \text{otherwise}
\end{cases} \)
4. \( \forall v \in V, \forall j \in \Omega \sum_{e \in v} y_i^e - \sum_{e \rightarrow v} y_i^e = \begin{cases} 
-1 & v = SP_j \\
0 & \text{otherwise} \\
+1 & v = CO
\end{cases} \)
5. \( \forall i \in S \sum_{e \in E} l(e) \cdot x_i^e + \sum_{j \in \Omega} \left( \omega_i^j \cdot \sum_{e \in E} l(e) \cdot y_i^e \right) \leq L_{\text{max}} \)  
   [relaxed]
6. \( \forall e \in E F(e) = \sum_{i \in S} x_i^e + \sum_{j \in \Omega} y_i^e \)
7. \( N = \sum_{j \in \Omega} n_j \)
2.6.3. Special case #3 (Digital Subscriber Line Networks, DSL with optical feeder)

Physical limitations on the (copper) distribution network segment are dominating the network reach constraints in this case, and obviously the DU capacities are also limited.

Regarding the cost function, it is important to note that these networks are typically installed in areas covered by already existing copper infrastructure, therefore cost of the distribution network is significantly lower due to re-use of copper instead of new optical fiber deployment. The active distribution units, as described in the previous section, require a high capital as well as high operational investment (CAPEX & OPEX), which actually dominates the topology dependent network costs, since the distribution network costs are almost negligible, not like optical access network technologies.

Fulfilling the copper loop constraints, and concurrently considering the capacity of the distribution units is in the heart of the problem in this case; according to the problem formulation, the distribution loop length ($L_{\text{dist}}$) and DU capacity constraints are kept, and the DU costs ($C_{DU}$), are considered for minimization.

In terms of optimization problem formulation:

**Objective:**

$$\text{minimize } c_{\text{topology dependent}} = N \cdot C_{DU}$$

**Constraints:**

(1) \[ \forall i \in S \sum_{j \in \Omega} \omega_j = 1 \]
2.6.4. SPECIAL CASE #4 (POINT TO POINT FIBER ACCESS NETWORKS, P2P)

Most of the discussed NGA network technologies have point-multipoint architecture. However, dedicated fiber access is a reasonable choice in some cases (e.g. for business users, FTTB systems or mobile backhaul applications). Moreover, the feeder part of any point-multipoint system is treated as a dedicated point-to-point network, connecting the distribution units to the central office. Therefore we need to discuss the optimization of point-to-point network topologies.

In a point-to-point optical access network, we have only one physical constraint for the length of the optical connections, between the CO and the demand points. This is typically a really permissive constraint: the dedicated fiber architecture offers the most robust optical transmission among the presented technologies, in terms of power budget and attenuation aspects. Therefore, when designing the optimal network topology, the length constraints are typically negligible.
Absence of DUs restricts the cost function to cable plant costs, without the feeder vs. distribution network distinction. It can be interpreted as the distribution network segment, with the CO serving as a DU. Topology dependent cost is reduced to the distribution network: primarily the dominating cable deployment ($C_D$) costs, and the fiber ($C_F$) costs with lower priority. In terms of problem formulation, the problem is significantly reduced, mostly due to the absence of the feeder network:

**OBJECTIVE:**

$$\text{minimize } C_{\text{topology dependent}} = \sum_{e \in F(e) > 0} (C_D(e) + \sum_{F(e)} C_F(e))$$

**CONSTRAINTS:**

(3) $\forall v \in V, \forall i \in S \sum_{e: v \to i} x^i_e - \sum_{e: i \to v} x^i_e = \begin{cases} |S| - 1 & v = CO \\ 0 & v = i \\ - & \text{otherwise} \end{cases}$

(5) $\forall i \in S \sum_e l(e) \cdot x^i_e \leq L_{\text{max}}$

(6) $\forall e \in E F(e) = \sum_{i \in S} x^i_e$

### 2.6.5. SUMMARY OF SPECIAL CASES

A graphical overview of these special cases is given in the following figure, highlighting the constraints and cost factors of higher significance. The relaxed constraints and cost (objective) components are represented by clear cells in the table. Significance of the non-relaxed cost components and constraints are indicated by vertical bars and darkness of the table cells.

<table>
<thead>
<tr>
<th>SPECIAL CASES: Significance map</th>
<th>Passive Optical Networks (PON)</th>
<th>Active Optical Networks (AON)</th>
<th>Digital Subscriber Line (DSL)</th>
<th>Point-to-Point fiber access (P2P)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constraints</td>
<td>Distance (end-to-end)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Distance (feeder)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Distance (distribution)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DU capacity</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs</td>
<td>Cable deployment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fiber</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cable deployment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fiber</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DU cost</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE 10 SPECIAL CASES OF THE NGA TOPOLOGY DESIGN PROBLEM**
3. ALGORITHMIC ANALYSIS

The NGA Topology Design problem was formulated in the previous section, followed by its most important special cases. The next step is the algorithmic analysis of the formulated, mathematical problem. The first straightforward question is on complexity: is it possible to solve the problem in reasonable time at all?

The “reasonable time” obviously is not an exact term, it needs further clarification. Network planning or topology design is an offline problem; hence time complexity requirements are not strict. However, the typical problem size is the service area of a single central office, i.e. graphs with thousands or even tens of thousands of nodes. These large scale problem instances do not allow algorithms with exponential complexity. Therefore, the fundamental question is not on the running time, but on general applicability: algorithms with exponential complexity will not provide solutions for the addressed problem size. Hence “reasonable complexity” refers to algorithms that are capable to handle scenarios with 10,000+ nodes.

What follows is the algorithmic analysis of the NGA Topology Design problem in its general form and special cases: is it possible to find the exact optimum in polynomial time or not (Section 3.1)? In cases where the exact optimization turns to be intractable, the next important question addresses approximability: how close can we get to the optimum within polynomial time (Section 3.2)?

3.1. COMPLEXITY ANALYSIS

Since the problem is now formulated as an optimization problem, the primary question addresses its complexity. The generally used methodology for proving complexity results is to construct reduction schemes. The so-called Karp-reduction supports proof of statements like “the problem is at least as complex as problem P”, which P used to be an appropriate known NP-hard problem. Via bi-directional proofs the Karp-reduction also supports NP-completeness statements [57].

More information can be derived from the so-called linear reduction (L-reduction) scheme that maintains not only complexity, but also approximability features of the problem, being bi-directional in every case. A linear reduction scheme between \( P_1 \) and \( P_2 \) shows equivalence of two problems: any solution for \( P_1 \) directly leads to a solution of \( P_2 \) via the L-reduction scheme [58].

In the sequel, such linear reductions will be given for the earlier defined special cases of the NTD problem, in order to prove that their complexity is equal to other already known mathematical problems. The following subsections have a uniform structure: at first, an already known (NP-complete) mathematical problem is defined (“base problem”). Then a mutual linear reduction is given, between the respective special case and the base problem, and conclusion is drawn on complexity of the addressed NTD special case, according to the complexity of the base problem.

Finally, complexity features for the NTD problem in general are also addressed.

3.1.1. SPECIAL CASE #1 (PASSIVE OPTICAL NETWORKS, PON)

**Theorem 1:** The special case of NGA Topology Design (NTD) problem for PON networks is NP-hard.

The Capacitated Facility Location (CFL) problem is a well-known optimization problem, with numerous applications, e.g. in the field of logistics, transportation, hub location or even network design problems. In the heart of the problem lies the decision about locations where facilities will be opened, and demand points which will be served through these facilities, e.g. a set of grocery stores
for the residents of a city. There is a cost assigned with opening a facility (i.e. the cost of opening a store), and also with the connection between a demand points and a facility (e.g. the distance between a home and the corresponding store). In the capacitated version, the facilities may serve a limited number of demand points, even though multiple facilities may be opened at the same location, increasing the facility costs.

**Definition 1 (base problem):** Capacitated Facility Location (CFL) problem [67]

In a capacitated facility location problem we are given two sets: \( F \), the set of facilities and \( C \), the set of clients (i.e. demand points). There is a specified distance \( c_{ij} \geq 0 \) between every pair \( i, j \in F \cup C \), costs \( f_i \geq 0 \) for opening facilities \( i \in F \), and capacities \( u_i \) respectively. Multiple facilities can be opened at the same site, and each copy incurs cost \( f_i \), and serves \( u_i \) clients. A multi-set \( S \) of facilities has to be identified to serve the clients in \( C \) by the facilities in \( S \) such that the total facility cost plus the total service cost is minimized. That is, if a client \( j \in C \) is assigned to a facility \( \sigma(j) \in S \) with cost \( C_{\sigma(j),j} \), then we want to minimize \( \text{cost}(S) = \sum_{i \in S} f_i + \sum_{j \in C} c_{\sigma(j),j} \).

**Linear Reduction**

**Lemma 1:** The NGA Topology Design (NTD) problem in its special case for PON networks, and the capacitated facility location (CFL) problems are equivalent under linear reductions (L-reductions). In this special case the \( C_0(e) \) values are merged into the increased \( C_{\sigma}^*(e) \) values, and the cost of the \( CO - DU \) (feeder) connections are merged into the DU costs.

**Proof 1:** A polynomial time reduction scheme is given for both directions.

**CFL → NTD**

Given a CFL problem instance, including the facilities, clients, distances, etc. An NTD problem will be constructed, which is equivalent to the CFL problem, i.e. it has the same optimum.

The construction ensures that the two problems are identical, i.e. they have coincident optimum. This NTD problem instance can be constructed in polynomial time obviously, and an optimal solution for it also solves the corresponding CFL problem. It both problems, we have points to be assigned to some nodes in central position. What makes a significant difference is the existence of the graph (that has to be followed by the connections) in the case of the NTD problem, while in a CFL problem the Euclidean space may be used, without restrictions. Figure 11 shows a pair of CFL and NTD problems, in which the clients / demand points, and the facilities / DUs are at the same position. The original graph of the NTD problem is faded but visible at the CFL problem.

The CFL → NTD\(_{PON}\) construction is composed of the following steps:

- create the NTD graph \( G = (V,E) \):
  - \( V = F \cup C \), i.e. from the set of clients and facilities
  - \( E = \{(i,j)\mid \forall i, \forall j \in V\} \), i.e. create a full graph,
- the set of DU locations represent the facilities:
  - \( \Omega = F \),
- the set of demand points represents the and clients:
  - \( S = C \),
- edge cost values are to be assigned as:
  - \( C^*_\sigma(e = (i,j)) = c_{ij} \), i.e. the NTD edge costs will be equal to the CFL costs (distances)
- DU capacities should be the same as the corresponding facility capacities:
  - \( K_i = u_i \)
This construction ensures equivalence of the two problems, i.e. their optimum coincides.

**NTD\textsubscript{PON} → CFL:** In this case, we are given a NTD problem instance, including the graph, demand points, DU locations and edge costs. Now a CFL problem instance with the same optimum value and same solution will be constructed as follows:

- the set of CFL clients will be equivalent to the set of NTD demand points:
  - \( C = S \)
- the set of facilities will be equivalent to the set of DU locations:
  - \( F = \Omega \)
- all other nodes of the NTD graph (i.e. nodes without any specific role such as \( S, \Omega \) and \( \text{CO} \), defining just the underlying street system graph) are unnecessary, since the CFL problem does not follow the graph itself: remove nodes of \( V \setminus S \cup \Omega \)
- the cost of assigning a client to a facility in the CFL problem will be equivalent to the cost of the shortest path (\( P_G \)) between the corresponding demand point and DU location in graph \( G \) of the NTD problem:
  - \( c_{ij} = \sum_{e \in P_G} C'_C(e) \)
- the \( u_i \) capacity values of the facilities should be equal to the respective \( K \) capacity of the DUs:
  - \( u_i = K \)
- the \( f_i \) cost of the facilities is calculated as the sum of the \( C_{DU} \) DU cost and the cost of the feeder link, i.e. the shortest path from the CO to DU location \( DU_i \):
  - \( f_i = C_{DU_i} + \sum_{e \in P_{\text{CO-DU}}} C'_C(e) \)

This construction results in a CFL problem instance. Its optimum coincides with the original NTD problem. I note that shortest path algorithms are running in polynomial time, therefore this reduction has polynomial complexity, and it provides identical optimality features for the CFL problem and the original NTD problem.

**Corollary 1a:** Since the facility location problem is known to be NP-hard \cite{67}, according to the Linear reduction, the \( NTD_{PON} \) special case itself is also NP-hard.
3.1.2. SPECIAL CASE #2 (ACTIVE OPTICAL NETWORKS, AON)

**Theorem 2:** The special case of NGA Topology Design (NTD) problem for AON networks is NP-hard.

Obviously there are similarities between the $NTD_{AON}$ and $NTD_{PON}$ problems. The most significant difference is due to the distribution units: in a PON network, the DUs are relatively cheap, having low capacity. In an AON network, the DUs have much higher capacity, and the DU costs are much higher at the same time. It has an important impact on the optimization problem: minimizing the number of DUs comes into focus. Due to their high capacity, an AON network typically requires just a few DUs to be deployed, not a large set of DUs as for PON networks.

These altogether imply that in $NTD_{AON}$ problem, the number of necessary DUs (denoted by $p$) will be treated as a pre-defined constant. Since the permissive distance constraints allow an extent of demand points being assigned to DUs that completely fills the DU capacity, the minimal number of necessary DUs is derived from the population ($N$) and the DU capacity ($K$): $p = N/K$. The overwhelming cost of DUs enforces their maximal utilization, i.e. (near) minimal number of DUs leads to the minimal cost topology. A well-known algorithmic problem, the capacitated p-median problem (CPMP) meets this interpretation of the $NTD_{AON}$ problem, and it will be used in the linear reductions:

**Definition 2 (base problem):** Capacitated p-median problem (CPMP) [70]

In a p-median problem, a set of points $N$, distance values $\{(d(i,j))|i, j \in N\}$, and an integer value $p$ is given. The problem is how to select $p$ points of $K \subseteq N$ as medians: all other points in $N \setminus K$ have to be assigned to their nearest median. Objective of the optimization problem is to minimize the sum of distances between all points in $N \setminus K$ and their respective medians. In the capacitated version of the problem, each median can have a maximum of $C$ points assigned.

The relationship between the capacitated facility location (CFL) and the capacitated p-median problems (CPMP) is interesting: the number of medians is given as a constraint for p-median problems, while it is involved in the optimization process for facility location problems. Therefore opening a facility is similar to violating that constraint, and it results in additional cost for that. This technique is known as Lagrangean relaxation: some constraints are integrated in the objective itself with a “penalty” cost for violating them, i.e. the facility cost for facility location problems (see [81], p. 167).

**LINEAR REDUCTION**

**Lemma 2:** The NGA Topology Design (NTD) problem in its special case for AON networks, i.e. when length constraints are relaxed, mainly the DU capacity constraints and costs ($C_{DU}$) are characterizing the problem, and the capacitated p-median problems (CPMP) are equivalent under linear reductions (L-reductions).

**Proof 2:** A polynomial time reduction scheme is given for both directions.

**CPMP $\rightarrow$ NTD$_{AON}$:** Given a capacitated p-median problem (CPMP) instance, an equivalent NTD problem instance will be is constructed, which has a coincident optimum:

- given the set of nodes $N$ for the CPMP problem, the graph $G = (V, E)$ of NTD is constructed:
  - $V = N$,
  - $E = \{(i, j) | \forall i, \forall j \in N\}$, i.e. it will be a full graph
- all nodes of the CPMP problem will be treated as available DU locations in the NTD problem:
  - $\Omega = N$,
• similarly, every node of the graph is a demand point at the same time:
  o $S = N$
• the edge cost values will be equal to the distance between the two nodes of the edge in the CPMP problem:
  o $C_e((i, j)) = d(i, j)$,
• the DU capacities will be set to $|N|/p$, according to the explanation above:
  o $K = |N|/p$
• DU costs will be set to a value which is high enough to dominate fiber costs:
  o $C_{DU} \gg d(i, j)$
• an arbitrary node in $S$ will be chosen as the CO.

This construction leads to an NTD problem in which the DUs are representing the (capacitated) medians. While DUs have a capacity of $|N|/p$, added that the DU costs are dominating the fiber costs, an optimal solution of the NTD problem requires exactly $p$ DUs to serve the $N$ demand points. These DUs will be located at positions minimizing the distribution network costs, i.e. distance sum of the CPMP problem. These altogether ensure coincidence of the optimum for the two problems.

The construction requires polynomial time itself, and a solution for the constructed NTD problem is a solution for the initial CPMP problem as well.

**NTD$_{AON}$ → CPMP:** Given an NTD problem instance, a capacitated plant location problem will be constructed with the same optimum.

In order to create the respective CPMP problem, an initial graph transformation step will be made on the NTD problem: the shortest path between every DU location and demand points within $l_{max}^{dist}$ distance will be substituted by a single edge (Figure 12). These length of these edges will be equal to the length of the corresponding shortest paths, with respect to fiber costs in $G$: $l'(i, j) = dist_G(i, j)$. This transformed graph will be referred to as $G^* = (V^*, E^*)$.

![Figure 12 Graph Transformation for CPMP](image)

By this transformation of the NTD problem, the equivalent CPMP problem is constructed as follows:

• the set of points $N$ for the CPMP problem will be identical to the nodes of $G^*$:
  o $N = V^*$
• distance between nodes of $N$ should be equal to the length of edges between them in $G^*$ if that edge exists, and $\infty$ otherwise:
  o $d(i, j) = \begin{cases} l'(i, j) & \text{if } (i, j) \in E^* \\ \infty & \text{if } (i, j) \notin E^* \end{cases}$
• capacity of the medians will be equal to the capacity of the DUs:
  o $C = K$
• the integer constant \( p \), i.e. the number of medians is calculated based on the DU capacities \((K)\) and the population:
  o \( p = |S|/K \)

This construction leads to a CPMP problem with \( p \) medians, i.e. the minimally necessary number of DUs in the \( NTD_{AON} \) problem, minimizing the distribution network fiber costs at the same time. The transformation clearly works in polynomial time, and the resulting \( p \)-median problem has identical optimality features as the original \( NTD_{AON} \) problem.

\[ \textit{Corollary 2a:} \text{ Since the (capacitated) k-median problem is NP-complete [72], according to the Linear reduction, \( NTD_{AON} \) special case of the NTD problem remains NP-complete.} \]

I note that if the assumption about DU costs dominating over link costs does not stand, we get a problem similar to that described for PON networks, which was also NP-hard.

3.1.3. SPECIAL CASE \#3 (DIGITAL SUBSCRIBER LINE NETWORKS, DSL)

\[ \textit{Theorem 3:} \text{ The special case of NGA Topology Design (NTD) problem for DSL networks is NP-hard.} \]

In this special case, the Set Cover problem will be used as the \textit{base problem}. In a Set Cover problem, we are given a set of elements, a series of subsets, and we are intended to find a minimum (cost) set of these subsets that covers the whole universe. Formally:

\[ \textit{Definition 3 (base problem):} \text{ Set Cover problem [54]} \]

In a Set Cover problem we are given a set of elements, or universe \( U = \{u_1; u_2; \ldots; u_n\} \), subsets \( S_1, S_2, \ldots, S_k \subseteq U \), and costs \( c_1, c_2, \ldots, c_k \). The optimization goal is to find a set \( I \subseteq \{1,2,\ldots,k\} \), that minimizes \( \sum_{i \in I} c_i \), such that \( \bigcup_{i \in I} S_i = U \).

The Set Cover problem will be used as the \textit{base problem} to prove NP-hardness, with a special graph transformation: the original graph \( G \) of the NTD problem is transformed to a \( G^* \) graph, as shown on Figure 13. This transformed graph is a 2-level tree, having the central office (CO) as its root, the DU locations \((\Omega)\) in the middle level, and the demand points \((S)\) as its leaves. All the DU locations are connected to the CO (with an equal \( C_{DU} \) link cost), but only those demand points are connected to a particular \( \omega_i \) DU location, which are within reach of it, e.g. the distribution loop length does not exceed its limit: \( d(\omega_i, s_j) \leq L_{\text{dist}} \).

![Figure 13 Steiner Tree Transformation of DSL Networks]
This graph transformation could be made in polynomial time, and it maintains optimality and validity: the same DU selection and demand point-DU assignment is the optimal result for both the original and the transformed problems, since the optimization objective has the same value (DU costs) and the distribution loop length constraints are fulfilled, through the construction of $G^*$. What follows is that solution of the NTD problem in $G^*$ and in the original graph is equivalent.

An optimal solution of the NTD problem in $G^*$ requires a minimal set of DUs connecting all the demand points – which is on the other hand a Set Cover problem, considering the demand points as the elements of the Universe, and the DUs as the subsets (where each subset contains the elements, i.e. demand points, which are connected to the respective DU). Adding a DU to the topology means adding the subset with its corresponding cost ($C_{DU}$), therefore it leads to the same cost increase in both problems. Therefore, the minimal cost set cover is equivalent to the minimal set of DUs.

**LINEAR REDUCTION**

**Lemma 3**: The NGA Topology Design (NTD) problem in its special case for DSL networks, i.e. when loop length ($l_{max}^{dist}$) and DU capacity constraints are kept, and primarily the DU costs ($C_{DU}$) are considered for minimization, and the Set Cover problems are equivalent under linear reductions (L-reductions).

**Proof 3**: A polynomial time reduction scheme is given for both directions.

$NTD_{DSL} \rightarrow Set\ Cover$: The above described graph transformation is necessary for the linear reduction.

The Set Cover problem will be then constructed in the following steps:

- elements of the **universe** represent the set of demand points:
  - $U = S$
- the **subsets** represent the DU locations, such that each element of the universe will be contained by the subset, if the corresponding demand point is connected to the DU location:
  - $u_j \in S_i$ if $s_j$ is connected to $\omega_i$
- the **cost** of each subset is equal to the DU cost:
  - $c_i \equiv C_{DU}$

Optimum of the Set Cover problem gives a minimal set of subsets, i.e. minimal set of DU locations in the original NTD problem; therefore an “oracle” capable to solve the Set Cover problem also solves the NTD problem special case for DSL networks.

$Set\ Cover \rightarrow NTD_{DSL}$: An $NTD_{DSL}$ problem instance will be constructed, based on the given Set Cover problem.

The construction will follow Figure 13. The **universe** of the Set Cover problem will be represented by demand points in the NTD problem, and one $\omega_i$, DU location will be added for each subset $S_i$ of the Set Cover problem. For each elements $u_j \in S_i$, the corresponding demand point $s_j$ will be connected to DU location $\omega_i$.

The $NTD_{DSL}$ problem is constructed in the following steps:

- regarding the graph $G = (V, E)$ of the $NTD$ problem:
  - the set of **nodes** $V$ in the $NTD$ graph will contain the following elements:
    - one **Central Office** node (CO)
Available DU locations \( \Omega \): one \( \omega_i \) DU location for each subset \( S_i \) of the Set Cover problem

One demand point for each element \( u_j \) in the universe of the Set Cover problem

- The set of edges \( E \) in the NTD graph connects each \( \omega_i \) DU location to the CO, and each demand point to those DU locations, for which the corresponding element of the universe belongs the respective subset, i.e.
  \[ E = \{(DU_i,u_j) | u_j \in S_i \} \cup \{(CO,DU_i) | i \in I \} \]

- The edge costs will be assigned as follows:
  - Installation costs for edges connecting DUs to the CO will be equal to the cost of the corresponding subset, installation costs on the demand point – DU edges, as well as all fiber costs will be set to zero:
    - \( C_0(DU_i,DU_j) = c_i \)
    - \( C_0(DU_i,u_j) = 0 \)
    - \( C_0(e) = 0 \)

- DU costs will also be set to zero, therefore the cost minimization will tend to minimize the feeder installation costs, i.e. \( \sum_i c_i \):
  - \( C_{du} \equiv 0 \)

- The distribution loop length constraint will be set sufficiently long, in order to allow all demand points to be assigned to any of its connected DU locations (i.e. the length constraints may be neglected):
  - \( L_{dist}^{max} > \max \left( \text{dist}(s_i,u_j) \right) \)

In the optimal topology for this specially constructed NTD problem, a set of DU locations minimizing the feeder network installation costs will be selected to connect all the demand points – exactly the set which minimizes \( \sum_i c_i \), and covers all demand points, i.e. elements of the universe in the Set Cover problem.

Provided that the graph transformations are made in polynomial time, and it maintains optimality and validity of the solution, the two problems are equivalent under linear-reductions.

**Corollary 3a**: Since the Set Cover problem is known to be NP-hard [54], according to the Linear reduction, this special case of the NTD problem is also NP-hard.

### 3.1.4. Special case #4 (Point to Point Fiber Access Networks, P2P)

**Theorem 4**: The special case of NGA Topology Design (NTD) problem for P2P networks is NP-hard.

The NTD_P2P problem is more directly related to the Steiner-tree problem than the earlier discussed NTD_DSL problem: in a point-to-point access network, several nodes (the demand points) have to be connected to the CO. If we take the CO and the demand points as terminal nodes of the graph, we see the relationship with the Steiner-tree problem. Therefore, the Steiner-tree problem will be used as the base problem to prove NP-hardness of the NTD_P2P problem, but now it requires graph transformation different from the one used earlier for the NTD_DSL problem.
LINEAR REDUCTION

**Lemma 4:** The NGA Topology Design (NTD) problem in its special case for P2P networks, i.e. when the cable deployment cost dominates all other costs, and the Steiner tree problems are equivalent under linear reductions (L-reductions).

**Proof 4:** A polynomial time reduction scheme is given here for both directions.

*Steiner-tree → $NTD_{P2P}$:* Given a Steiner-tree problem instance, an $NTD_{P2P}$ problem will be constructed, which has the same optimum. The construction follows the above described relationship between the two problems:

- the graph $G = (V, E)$ of the $NTD$ problem will be identical to the graph of the Steiner-tree problem, i.e. it has the same set of nodes and edges
- the Central Office (CO) of the $NTD$ problem will be located an arbitrarily chosen node which was a terminal node ($r_i$) in the Steiner-tree problem:
  - $CO \equiv r_i$,
- the set of demand points in the $NTD$ problem will be identical to the set of terminal nodes in the Steiner-tree problem:
  - $S = R$,
- the edge cost values of the $NTD$ problem will be assigned as follows:
  - $c_0(e) = w(e)$, i.e. the deployment costs are equal to the edge costs of the Steiner-tree problem
  - $c_u(e) \equiv 0 \ \forall e \in E$, i.e. the fiber costs are negligible
- finally, the distance limitation are relaxed, i.e. set to infinity:
  - $L_{\text{max}} = \infty$.

This $NTD$ problem instance is a (simplified) $NTD_{P2P}$ problem, which is equivalent to the original Steiner-tree problem. Therefore its optimal solution gives a solution of the original Steiner-tree problem at the same time. The construction has polynomial complexity.

*$NTD_{P2P} \rightarrow$ Steiner-tree:* the construction is simple and straightforward. The two graphs will be the same, the Steiner-tree will be “spanning” a subgraph of the CO and the demand points. Formally:

- graph $G$ of the Steiner-tree should be identical to the graph $G = (V, E)$ of the $NTD$ problem
- terminal nodes of the Steiner-tree will be the nodes representing the Central Office (CO) and the demand points (S) in the original $NTD$ problem:
  - $R = S \cup \{CO\}$
- edge weights of the Steiner-tree will be equal to cable deployment costs of the $NTD$ problem:
  - $w(e) = c_0(e)$

The transformation has linear complexity, and maintains optimality features. Moreover, the optimal Steiner-tree itself is identical to the optimal P2P topology, with minimal deployment cost for the original $NTD_{P2P}$ problem.

**Corollary 4a:** Since the Steiner-tree problem is known to be NP-complete, even for planar or bipartite ($R, V\setminus R$) graphs, or with equal edge weights [61], the reduction makes this special case of the NTD problem also NP-complete, even with these more restricted scenarios.

**Remark:** The feeder network segment of AON and DSL networks may be interpreted as a P2P network: the DUs are connected directly to the CO with dedicated optical fibers.
I have to note here, that even if we have good reasons to neglect the length constraints for the P2P special case (as explained in Section 2.6.4), it does not affect the NP-hardness of the problem. With the length constraints kept we arrive to the problem which is known as the “Delay/Diameter Constrained Steiner Tree (DCST)” problem [65]. The problem was raised with multicast trees, and it is known to be NP-hard. The NP-hardness holds even if the “topology” of the Steiner-tree is fixed, which assumption does not hold in our case [64]-[66].

3.1.5. NTD PROBLEM IN GENERAL

In the previous four subsections, several special cases of the NTD problem were discussed. These were based on the concept of narrowing the set of loose constraints or excluding parts of the objective (cost function). The described special cases lead to “almost valid”, “almost optimal” solutions for a subset of NTD problems, with special input settings given in section 2.6.

However, the problem even in its simplified forms remains NP-complete.

**Theorem 5**: What follows is that the general case of the problem is clearly NP-complete, since any of its relaxations has been proven to be NP-complete.

**Proof 5**: A general NTD problem, without any restrictions on input values, is composed of the two above discussed optimization problems: For the feeder network part, a Steiner-tree problem has to be solved, for the distribution network part, a capacitated facility location problem is given. Both are NP-complete, what makes the NTD problem also NP-complete. ■

3.2. APPROXIMABILITY STUDIES

Even though the NTD problem and its special cases were proven to be NP-hard, my goal is still to find a solution for them, even for large scale scenarios. When facing such highly complex problems, two fundamental approaches are considered:

- **Relaxation**, when a subset of the constraints is relaxed, eliminated, and the problem is optimized in a wider solution space. The solution may be suboptimal or even invalid for the original problem, but at least it is achieved in a reasonable (typically polynomial) time.

- **Approximation**, in contrary leads to valid solutions. In this case the constraints are kept, but exact optimality is not forced, an approximation is required instead. The running time (complexity) has to be decreased at the expense of losing optimality.

The fundamental requirement of practical applicability does not allow oversimplification. The special cases themselves are acceptable relaxations. Unfortunately, as it was proven, that extent of relaxation still does not allow exact optimization in polynomial time: the special cases are still NP-hard. However, the realistic approach would be lost with a higher level of relaxation, which is contradictory with my fundamental goals.

These altogether imply that a combination of these complexity reduction principles is in demand: the special cases as relaxations have to be solved by approximation algorithms (heuristics). Such a combination is necessary for the highest quality solution achieved in a reasonable time.

What follows in this subsection is the approximability analysis of these special cases. I will use the term **theoretic bounds** for lower bounds on the best approximation factors obtainable in polynomial time. Typically indirect proofs are known for these, stating “no better approximation factor exists, unless P = NP”. The best known approximation algorithms for an equivalent or analogous
mathematical problem will be treated as **practical bounds**. These show what is surely possible, since these related algorithms can be interpreted as constructive proofs for some approximation ratio that is obtainable.

*It is impossible to construct any better heuristic algorithm than the theoretic bounds, and not reasonable to expect better than the practical bounds.*

The presented approximability results are typically corollaries of the linear reductions presented and proven in section 3.1 about complexity analysis, therefore the numbering is consequent to that section.

### 3.2.1. SPECIAL CASE #1 (PASSIVE OPTICAL NETWORKS, PON)

This special case was proven to be equivalent under linear reductions with the Capacitated Facility Location (CFL) problem (**Lemma 1**).

**Corollary 1b**: The linear reduction maintains validity of the approximability results of the facility location problem for the $NTD_{PON}$ problem.

The facility location problem is a well-known mathematical problem, Guha et al. prove [73] that it is NP-hard to approximate the facility location problem with a better constant factor than 1.463, which gives the *theoretic lower bound* also for the $NTD_{PON}$ problem approximation.

The best known approximation of the capacitated facility location problem is from Mahdian et al. providing 2-approximation with a constructive proof, which sets the *practical lower bound* [69].

### 3.2.2. SPECIAL CASE #2 (ACTIVE OPTICAL NETWORKS, AON)

This special case was proven to be equivalent under linear reductions with the capacitated p-median problem (**Lemma 2**).

**Corollary 2b**: The linear reduction maintains validity of the approximability results of the k-median problem for the $NTD_{AON}$ problem.

Meyerson et al. [72] prove that it is NP-hard to approximate the k-median problem with better approximation factor than $1 + \frac{2}{e} \approx 1.736$, which gives the *theoretic lower bound* also for the FTD problem special case.

The best known approximation algorithm for the k-median problem however has significantly higher approximation ratio: Arya et al. [71] give a constructive proof for $3 + 2/p$ constant factor approximation, where the value of $p$ is a variable describing complexity of an interior iterative step of the algorithm. It makes the approximation factor scalable, but not lower than 3.

### 3.2.3. SPECIAL CASE #3 (DIGITAL SUBSCRIBER LINE NETWORKS, DSL)

Partial equivalence under linear reduction with the Set Cover problem was proven (**Lemma 3**).

**Corollary 3b**: Through a graph transformation step, equivalence of the $NTD_{DSL}$ and the Set Cover problem was proven, therefore not only their complexity, but also approximability features are identical.

For the Set Cover problem, without any further restrictions, the greedy algorithm gives $H(n)$ approximation, where $H(n)$ is the $n$th harmonic number ($H(n) \leq 1 + \ln n$) - $H(n)$ being the *practical lower bound* in this case. On the other hand, no constant factor polynomial time algorithms may be
constructed for the Set Cover problem, the best polynomial time approximations can give \((1 - \varepsilon) \cdot \ln n\) approximation of the optimum [74] – being the \textit{theoretic lower bound} here.

However, with a reasonable restriction of the Set Cover problem, we get better approximation algorithms. Where each set has a size at most \(B\), (i.e. no demand points are within reach from more than \(B\) DU locations), the \textit{theoretic lower bound} becomes as low as \(B^{1/19}\), while the best known solutions deliver \((1 + \ln B)\) approximation, being the \textit{practical lower bound} in this case [75].

In Chapter 4, I will introduce the notion of criticality for DU locations and demand points, which also addresses that bounding value of \(B\) – here I just shortly mention that the existence of tight length constraints (i.e. short reach of DUs) makes an upper bound on set sizes a reasonable assumption.

### 3.2.4. Special case #4 (Point to Point Fiber Access Networks, P2P)

This special case was proven to be equivalent under linear reductions with the Steiner-tree problem (Lemma 4), therefore it has similar complexity.

**Corollary 4b**: The linear reduction maintains validity of the approximability results of the Steiner-tree problem for the \(N\)\(TD\)\(_{P2P}\) problem.

Based on published results for the Steiner-tree problem, the theoretic lower bound is 1.0062 [61], while the practical lower bound 1.55 [62] for approximation of the \(N\)\(TD\)\(_{P2P}\) problem.

### 3.2.1. NTD Problem in General

**Theorem 5b**: The inherited \textit{practical lower bound} for constant factor approximation of the general NTD problem is not less than 3, and the \textit{theoretical lower bound} is not less than 1.736, i.e. it is not possible to approximate the NTD problem within a factor of 1.736 unless \(P = NP\), however the best known heuristic algorithm for the underlying problems provides \(3 + 2/\rho\) approximation.

**Proof 5b**: Investigation the listed special cases reveals the approximability of the NTD problem in general. The cost function of the general problem consists of three main elements: \(C_0\), the deployment costs; \(C_v\), the fiber costs; and \(C_{DU}\), the DU costs.

The cost function of the \(N\)\(TD\)\(_{P2P}\) special then covers the \(C_0\) deployment costs, and for those, we have 1.0062 or 1.55 theoretic and practical approximability bounds. The cost function of the \(N\)\(TD\)\(_{P2P}\) special then covers the \(C_v\) fiber costs, and for those, we have 1.736 or \(3 + 2/\rho\) theoretic and practical approximability bounds. Finally, the cost function of the \(N\)\(TD\)\(_{DSL}\) special case covers the \(C_{DU}\) distribution unit costs, with the most complicated approximability bounds – let them be \(B^{1/19}\) and \(1 + \ln B\), respectively.

Since the \(N\)\(TD\)\(_{DGN}\), \(N\)\(TD\)\(_{DSL}\) and \(N\)\(TD\)\(_{P2P}\) special cases, as well as the original NTD problems are solved over the same solution space, the NTD problem not only contains them as special cases, but its objective is the sum of these special cases. What follows is that the lowest best possible constant approximation factor for the general NTD problem cannot be lower than the maximum of the weighted average of approximation factors for these subproblems, since a constant \(\alpha\)-factor approximation for such a composition of the problems should lead to an \(\alpha\)-factor approximation for any of its composers as well.
The practical lower bound on the approximation factor (3, 1 + \ln B and 1.55 for the capacitated p-median, bounded set cover and Steiner-tree problems, respectively):

\[
\max_{\alpha, \beta}(\alpha \cdot 3 + \beta \cdot (1 + \ln B) + (1 - \alpha - \beta) \cdot 1.55) = \max(3, 1 + \ln B)
\]

Regarding the theoretical lower bound (1.736, \(B^{1/19}\) and 1.0062 for the capacitated p-median, set cover and Steiner-tree problems, respectively):

\[
\max_{\alpha, \beta}(\alpha \cdot 1.736 + \beta \cdot B^{1/19} + (1 - \alpha - \beta) \cdot 1.0062) = \max(1.736, B^{1/19})
\]

3.3. Conclusion

Formulation of the NTD problem, definition of its important special cases, and the exhaustive complexity and approximability analysis completes the theoretic background of the problem addressed in my dissertation. The following table concludes the results achieved regarding the theoretical background of the NGA Topology Design problem. Complexity and approximability analysis were carried out not only for the general form of the problem, but also for its special cases, motivated and defined by current and future NGA technology and architecture classification.

The proven complexity, i.e. NP-completeness of problems shows necessity for heuristic approximation of the optimum. The approximability results show possibilities and limits of guaranteed approximation: no better approximation can be carried out than the theoretic lower bound, and to my knowledge no better approximation algorithm was published over the last decades for the underlying mathematical problems than the practical lower bound.

<table>
<thead>
<tr>
<th>ALGORITHMIC ANALYSIS SUMMARY</th>
<th>(NTD) in general</th>
<th>(NTD_{PON}) special case</th>
<th>(NTD_{AON}) special case</th>
<th>(NTD_{DSL}) special case</th>
<th>(NTD_{P2P}) special case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear reduction “base problem”</td>
<td>-</td>
<td>Capacitated facility location</td>
<td>Capacitated p-median</td>
<td>Set Cover</td>
<td>Steiner-tree</td>
</tr>
<tr>
<td>Complexity</td>
<td>NP</td>
<td>NP</td>
<td>NP</td>
<td>NP</td>
<td>NP</td>
</tr>
<tr>
<td>Approximability (theoretic)</td>
<td>(\geq 1.736)</td>
<td>1.463</td>
<td>(1 + 2/e = 1.736)</td>
<td>(B^{1/19}) (*) or ((1 - \varepsilon) \cdot \ln n)</td>
<td>1.0062</td>
</tr>
<tr>
<td>Approximability (practical)</td>
<td>(\geq 3)</td>
<td>2</td>
<td>(3 + 2/p &gt; 3)</td>
<td>(1 + \ln B) (*) or (1 + \ln n)</td>
<td>1.55</td>
</tr>
</tbody>
</table>

TABLE 4 SUMMARY ON THEORETICAL BACKGROUND

(*) For the role and definition of B, please refer to section 3.2.3.
4. CRITICALITY: A GRAPH THEORETIC APPROACH

The NGA Topology Design Problem, as it was outlined in Section 2, was translated to the language of graph theory. In this section, I will review the problem from a graph theoretic point of view. Some features of the modeling graph itself may bring us closer to either the optimum or at least upper and lower bounds. In order to find the relevant features, we have to take a look at the constraints and cost factors of the NTD problem, and make an effort to describe what effect these may have on the optimal solution.

Here I focus on the graph, therefore the solution space narrowing effect of constraints, especially the range (length) constraints are important and discussed in this section. Namely the total network range, plus the feeder and distribution network segment limitations will be investigated.

Total network range has an obvious and substantive consequence: any demand point further than $L_{\text{max}}$ from the CO will remain unconnected. The feeder network range also has a relatively straightforward effect: it constraints the DUs to be close to the CO; in the extreme case, in the CO itself, which degenerates the point-multipoint network to a point-to-point architecture.

The distribution network range limitations have the most complicated effect on valid topologies, i.e. the solution space. These have the most interesting interaction with the graph itself, since short distribution network segments emphasize local features of the graph. Low $L_{\text{max}}^\text{dist}$ values have key importance regarding the clustering subproblem: the topology design problem tends to find coverage of the service area with low diameter groups, and in this case, a properly chosen set of DU locations is fundamental. Several critical nodes of the graph may be identified that determine the location of DUs.

Identification of those critical nodes, the notion of criticality and its applications is the subject of the remainder of this section.

4.1. DEFINITION OF CRITICALITY METRIC AND ORDERING

Some demand points in the graph may have a special position, especially with short distribution network segment constraints. Outlier nodes are the most obvious examples of nodes with significant effect on topology: these may require a dedicated DU in their outlier position. On the other hand, nodes in densely populated, central areas typically do not affect the topology seriously: a DU position may be chosen from a wide set of available DU locations to connect them.

These examples are trivial for the human eye, but need some formal explanation for an algorithm. Such formal representation also helps deeper investigation, beyond trivial observations. In order to identify the necessary DU locations for complete coverage of demand points, I have introduced a metric to measure such significance of a node in the graph, namely node criticality.

Definition 1: Node criticality

Measures the number of DU locations that can serve the demand point, i.e. from which the node is within $L_{\text{max}}^\text{dist}$ (distribution network reach). Formally, if $DU_i$ denotes the $i^{th}$ element within the set of available DU locations, $s_j$ denotes the $j^{th}$ demand point, $d(x, y)$ is the distance of $x$ and $y$, $L_{\text{max}}^\text{dist}$ is the maximum DU-demand point distance:

$$\text{crit}(s_j) = |\{DU_i|d(DU_i, s_j) < L_{\text{max}}^\text{dist}\}|$$
Intuitively, this metric stands for the number of DU locations, of which at least one must have a DU in the final topology to connect the demand point. The lower this value is, the more critical a demand point is. On Figure 14, the bold numbers top-right from the nodes indicates their criticality value. Clearly, nodes with criticality value of 1 determine that a DU has to be located in their corresponding DU location.

Here I recall the approximability results on the $NTD_{DSL}$ problem (section 3.2.3): if the size of the sets is bounded by $B$, i.e. there is no demand point that may be covered by more than $B$ DU locations, there exist much better approximations for the set cover problem than in the general case. Using the notion of criticality, $B = \max_j \text{crit}(s_j)$.

**Definition 2**: Criticality code

For every node, the DUs within distance $D_{\text{dist}}$ are enumerated, in ascending order of their IDs. Such a list is referred to as the criticality code of the node, enlisting the available DU locations for covering the node:

$$\text{criticality\_code}(s_j) = \#\text{ASC} \{ DU_i | d(DU_i, s_j) < D_{\text{dist}} \}$$

On Figure 14, small numbers within parentheses, e.g. {1,2,3} indicate the criticality code of the node.

**Definition 3**: Component

A group of demand points that are reachable from exactly the same set of DU locations, i.e. a group of (neighboring) nodes having the same criticality code form a component.

On Figure 14, such components are indicated with dashed lines, and these components are substituted by single dots on the following schematic figure (Figure 15). Such simplified form of the graph will be used in the following sections, since a group of nodes having the same features regarding criticality will be handled together.

**Definition 4**: Criticality list

DUs have a criticality list. Criticality values of the demand points within reach of a given DU, in ascending order serve as the criticality list of the DU.

$$\text{List}(DU_i) = \#\text{ASC} \{ \text{crit}(s_j) | d(DU_i, s_j) < D_{\text{dist}} \}$$

The underlined numbers within parentheses, under the DU locations on the figure give an example that helps better understanding of the criticality lists.

Let us also define an ordering relation on DU locations with the help of a lexicographical ordering on their criticality lists:

**Definition 5**: Criticality ordering

This relation is defined on DUs, based on their criticality lists. By notation: $DU_i \leq_c DU_j$, if the criticality list of $DU_i$ precedes that of $DU_j$ in a lexicographical ordering. In the graph it means $DU_i$ has not less nodes with lower criticality value. The criticality relation reflects an ordering driven by importance or necessity of DU locations.

In the given example, the DU locations have the following criticality values:
• $DU_1: (1,2,2,2,2,3)$
• $DU_2: (1,1,1,2,2,3)$
• $DU_3: (1,1,2,2,3)$

This implies the ordering $DU_2 \leq_c DU_3 \leq_c DU_1$.

**Lemma**: The criticality relation is an ordering on the set of DUs ($\Omega$).

**Proof**: Criticality relation is defined as a lexicographic ordering. $lacksquare$

**Corollary**: The criticality ordering has the following features:

• $DU_i \leq_c DU_i$ (reflexive)
• $DU_i \leq DU_j \land DU_j \leq DU_i \Rightarrow DU_i = DU_j$ (antisymmetric)
• $DU_i \leq DU_j \land DU_j \leq DU_k \Rightarrow DU_i \leq DU_k$ (transitive)
• $\forall_i, j: DU_i \leq DU_j \lor DU_j \leq DU_i$ (total)
4.2. Matching and criticality: an upper bound on DU locations

Based on the notion of components, a simplifying graph transformation will be made: every set of nodes having identical criticality codes will be substituted by a single node (according to the definition of criticality code, these nodes are connected to the same DUs). The transformation is shown on the simplified graph above (Figure 15), and re-drawn on Figure 16 to highlight the bipartite property of the simplified graph: the components are connected only to DU locations, but no connections exist within the subset of components or DUs. The upper (red) dots represent the components (groups of demand points with same DU connectivity), and the numbers in the dots stand for their criticality value. The black dots represent the DU locations.

![Figure 17 BIPARTITE SAMPLE GRAPH](image)

Considering the original topology design problem, a valid solution is a set of DUs “covering” all the components. A DU location will be called “active” if a DU is located there in the access network topology. In this section, we will focus on the problem of finding the minimal necessary set of DU locations to cover all components.

**Definition 1:** Minimal necessary set of DU locations

We are given a bipartite graph \( G = (V, E) \). The set of nodes is composed of components and DU locations \( V = C \cup \Omega \), and all of its edges are connecting a component to a DU location:

\[
E = \{(c_i, \omega_j) | c_i \in C, \omega_j \in \Omega\}
\]

A “valid solution” is a subset of DUs \( \Omega^* \), such that for all component nodes there is a neighboring DU location in \( \Omega^* \), while the optimal solution (i.e. the minimal necessary set of DU locations) is the \( \Omega^* \) set with the minimal cardinality:

\[
\Omega^* = \{\omega | \forall c_i \in C : \exists \omega_j \in \Omega^* | (c_i, \omega_j) \in E\}
\]

Such set of DU locations is a Minimum Independent Dominating Set (MIDS) in the bipartite graph, with the restriction that the independent set should be a subset of DU locations. A linear reduction between the MIDS and Set Cover problems is known [79]. Since we are addressing a bipartite graph now, in the following, I will show the relation between the minimal necessary set of DU locations and matching in bipartite graphs.

**Definition 2:** In a graph \( G = (V, E) \) a matching \( M \) is a set of pairwise non-adjacent edges, i.e. none of the edges in \( M \) share a single common vertex.

**Definition 3:** a maximal matching \( M \) is a matching that cannot be extended by adding any more edges to \( M \), since any additional edges would violate its matching property.

**Lemma 1:** Cardinality of any \( M \) maximal matching is an upper bound on the minimal number of necessary active DU locations to cover all components, i.e. \( |M| \geq |DU|_{\text{min}} \).
**Proof 1**: (Indirect)

Any $M$ matching can be translated to a set of active DU locations, by making active all DU locations which are adjacent to any edge of $M$. This set of active DU locations covers all components in $M$.

Suppose that there exists a component $c_j$, which remains uncovered by the above described DU location set, i.e. none of its neighboring DU locations (e.g. $DU_i$) are adjacent to the edges of $M$. In the bipartite graph, all edges adjacent to $c_j$ are connected to one of its neighboring DU locations, i.e. $M$ could be extended with an edge $(DU_i, c_j)$, since none of its terminating nodes have adjacent edges in $M$. Which in itself is contradictory to the initial assumption about $M$ being a maximal matching – and it proves the fact that the translated active DU location set covers all components.

On the other hand, this DU set has exactly $|M|$ elements, since in the bipartite graph every edge is adjacent to one DU location (and one component). These altogether imply that the minimal number of necessary DU locations is not more than $|M|$.

I also note that the inequality is not tight, e.g. in the following example, a maximal size matching contains 2 edges (A1, B2), while a single active DU location (B) may cover all the components:

```
A   B   C
1   2
```

**Figure 18 TIGHT INEQUALITY EXAMPLE**

This first lemma ensures that the number of necessary active DU locations is bounded from above – by the size of any maximal matching. The next necessary step is to find the strongest upper bound, i.e. the minimal size of a maximal matching.

**Lemma 2**: Finding the minimal size maximal matching is NP-complete.

**Proof 2**: This was proven by M. Yannakakis and F. Gavril in [78]

It unfortunately blocks our attempts to decrease the upper bound and tighten the limits, therefore the best upper bound on the number of necessary active DU locations is achieved by the size of an arbitrary maximal matching.

This fact also raises a question about the deviation in size of maximal matchings: may we accidentally get an arbitrary maximal matching which is 100 times bigger than the minimal size? It would significantly weaken the upper bound. The next lemma shows that an arbitrary maximal matching is a 2-factor approximation of the minimum size maximal matching (in any graphs):

**Lemma 3**: The size of an arbitrary maximal matching gives a 2-approximation on the size of the minimum size maximal matching (proven by Z. Gotthilf et al., [77]).

**Corollary 3**: Finding an arbitrary maximal matching leads to an easy-to-find upper bound on the minimum size maximal matching, i.e. also on the number of necessary DUs, even though this upper bound is not tight, i.e. in specific cases may be arbitrarily far from the minimum.
This last implication shows the reason to deal with matching: it can be applied to the NTD problem. A fast greedy algorithm will be proposed for calculating an upper bound on the necessary number of DUs, based on the notion of node criticality measure:

**Algorithm 1** Upper bound approximation for active DU locations:

- **Step 1:** Create the bipartite graph $G$ of components and DU locations as described above.
- **Step 2:** Select the components with criticality value of 1, and connect them to their single DU; add these DUs to the solution. These undoubtedly are necessary for full coverage of components.
- **Step 3:** Let $i = 2$, and then take the unconnected components with criticality value of $i$. Assign them to an arbitrary adjacent DU location until all of them are covered. If there are any more unconnected components, $i++$ and goto Step 3. Otherwise goto Step 4.
- **Step 4:** The selected subset of DU locations is sufficient for connecting all the components, a valid solution is found. STOP.

**Theorem 4:** Complexity of Algorithm 1 is $O(V^2 \cdot \log(V))$

**Proof 4:** Individual steps of the algorithm are analyzed regarding their complexity

The first, initial step for creating the bipartite graph is composed of three elements:

- Criticality values are set in $O(V^2 \cdot \log(V))$ steps: the Bellman-Ford algorithm is applied to calculate the distance between any pair of demand points and DUs, and it is compared to $L_{max}$. The ordered lists of DUs at every node (i.e. the criticality code) are made in $O(V^2 \cdot \log(V))$ steps, since at every node we have at most $O(V)$ DUs within reach.
- Components are created in $O(V^2)$ steps, comparing all nodes pairwise, and contracting the ones with the same criticality code.
- Once the components and DU locations exist, edges of the bipartite graph are added, based on the adjacency information in the component criticality codes, with at most $O(V^2)$ steps (since we have at most $O(V)$ components and DU locations)

These altogether result in $O(V^2 \cdot \log(V))$ complexity for Step 1. Step 2 has a complexity equal to the number of components: $O(V)$. Step 3 is executed at most $O(V)$ times, since the number of DU locations bound criticality (therefore $i$ is not increased further), and each step includes at most $O(V)$ assignments: $O(V^2)$ steps for all executions. If we sum up these, we get $O(V^2 \cdot \log(V))$ complexity for Algorithm 1.■

Summarizing this subsection: we have a polynomial time greedy algorithm for computing an upper bound on the number of necessary DU locations for connecting all demand points, subject to distribution loop length constraints.

### 4.3. LOWER BOUNDS OF DU SET

Finding a lower bound on the number of minimal necessary DU locations for connecting all demand points, with respect to distribution loop length constraints could also be a valuable addition.

An upper bound may be used as an obvious measure of quality (if a solution is provided with more DU locations, it definitely may be improved). On the other hand, a lower bound is more difficult to interpret: a solution far from the achieved lower bound still may be of high quality if the lower bound is not tight. However, any solution close to a known lower bound is immediately proven to be a high...
quality result. Accordingly, in this section an attempt is made for a theoretic lower bound computation, and in a following chapter I will present a linear programming based lower bound calculation method, which is more of practical interest, and also shows the real power of the proposed heuristics, having a relatively small gap between the achieved results and the known lower bound.

At first, a trivial lower bound is described, based on the notion of components and criticality:

**Theorem 5**: The number of components with criticality value of 1 ($n_{c=1}$) makes a lower bound on the number of necessary DUs ($\min_{DUs}$).

**Proof 5**: A set of demand points forming a component with criticality value of 1, have to be connected to only one DU location (all of them to the same). That requires that DU location to be active. However, no other components with criticality value of 1 can be connected to the same DU location, otherwise nodes of these two components would have the same criticality code, and the two components should be the same. ■

Similarly, a lower bound is derived from the number of components with criticality value of 2 ($n_{c=2}$):

**Theorem 6**: The number of components with criticality value of 2 ($n_{c=2}$) provides another lower bound on the number of necessary DUs ($\min_{DUs}$):

$$\min_{DUs} \geq 1 + \sqrt{1 + 8 \cdot n_{c=2}}$$

**Proof 6**: Supposed we are given $\omega$ different DU locations, the maximal number of different components (with different criticality codes!) is equal to $\binom{\omega}{2}$, since no more mutually different pairs of DU locations exist. Therefore:

$$\binom{\omega}{2} \geq n_{c=2}$$
$$\omega \cdot (\omega - 1) \geq 2 \cdot n_{c=2}$$
$$\omega^2 - \omega - 2 \cdot n_{c=2} \geq 0$$

And then, by solving the quadratic inequality, we get:

$$\omega \geq \frac{1 + \sqrt{1 + 8 \cdot n_{c=2}}}{2}$$

Considering non-negativity of $\omega$, we get $\omega \geq \frac{1 + \sqrt{1 + 8 \cdot n_{c=2}}}{2}$. ■

The computed minimal DU locations for different $n_{c=2}$ values are as follows ($\omega$ rounded up to the closest integer):

<table>
<thead>
<tr>
<th>$n_{c=2}$</th>
<th>$\omega$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
</tr>
</tbody>
</table>

According to the fact, that typically the $n_{c=2}$ values do not exceed significantly (if at all) the $n_{c=1}$ values (components with criticality 1), lower bound is typically tighter with $\omega \geq n_{c=1}$, except for small networks and just a few components. Similarly, additional lower bounds are available based on components with 3, 4, ... criticality, but for the same reason, these have low practical value. Increasing criticality values and bounds is more about number theory than the graph itself.
5. **Specialized, Highly Efficient Heuristics**

Beyond clarifying the theoretical background of algorithmic topology design for NGA networks, another fundamental goal of the research was to propose applicable solutions: algorithms that solve not only relatively small sample problem instances under “lab-conditions”, but also realistic, large-scale scenarios, which have practical interest.

Obviously the theoretical studies were necessary before developing algorithms, in order to decide whether the problem can be solved in an algorithmic way at all. The preceding sections support the observation that the problem is NP-complete, exact optimization expectedly cannot be carried out in polynomial time; moreover really tight guaranteed constant-factor approximation is beyond possibilities. The algorithmic analysis led to reasonable requirements regarding time complexity (scalability) and accuracy of the algorithms.

According to the introductory section, effective (likely polynomial) algorithms are required for sufficient scalability: an NTD problem instance solved at once is typically the service area of a single CO, which is in the magnitude of 1.000s to 10.000s of demand points. On the other hand, topology design is an offline problem, therefore several hours or even days of computation are tolerable. As we have seen, exact optimization has exponential complexity, which excludes practical applications, even with these permissive requirements.

Regarding approximation accuracy of heuristic methods, practical applications raise the bar: deployment of NGA networks requires very high investment; cost-minimization planning and accurate cost estimation is of high importance. Therefore a high extent of suboptimality is not acceptable: even the theoretically existent constant factor approximations with 46% or 73% gap are too loose, if we consider the magnitude of investment at these network deployments. However, by terms of guaranteed approximation, these cannot be outperformed.

Hence, for practical reasons, the approximability requirements will not be set for the theoretic “worst case” scenario, but for the average expected value of suboptimality, tested against various real-world scenarios and case studies. Extent of the allowed optimality gap is difficult to define, and even more difficult to test/check, since the exact optimum is typically not known.

As a conclusion, the requirements were set up for the topology design heuristics:

- **scalability**: large-scale problems, even for graphs with 10.000+ nodes
- **accuracy**: suboptimality gap within 10-20% in average

In order to achieve such high performance approximation, highly specialized heuristic algorithms are necessary. According to the well-known *No Free Lunch* theorem [56], specialization allows effective approximation of well-defined problems – this was driving the efforts for identifying special cases. The heuristics should be adapted to characteristics of the problem, and typical input parameters of the various special cases as far as possible.

Earlier in this dissertation, in chapter 2.5 there is an analysis of the various cost components and constraints, regarding their impact on the optimal solution, i.e. network topology. The findings outlined there are crucial for understanding how the heuristics are working.

In this chapter, various heuristic algorithms are proposed for special cases of the NTD problem. As a common feature, all of the presented heuristics are based on a decomposition of the topology design problem, which is introduced in the first subsection. I note here, that this section focuses on the three presented point-multipoint technologies, namely PON, AON and DSL. The point-to-point networks
were important for the theoretic studies, complexity and approximability analysis, but for topology design, the already published algorithms for solving the Steiner-tree problem may be used (e.g. the Distance Network Heuristics [59], or the algorithm of Zelikovsky [62]).

5.1. DECOMPOSITION INTO SUBPROBLEMS

Complexity analysis not only led to the recognition of NP-completeness, but the efforts to identify the included NP-complete subproblems also highlighted the most critical points of the NGA Topology Design (NTD) problem itself.

On an abstract level, optimization for both the general case and special cases of the problem can be seen as a minimum cost, point-multipoint architecture, considering various constraints, connecting demand points to the central office through distribution units. During the design of such topology, three basic questions should be answered:

- In the point-multipoint structure, which demand points should be in the same group (cluster), assigned to a common DU?
- Where should be the distribution units of these groups located?
- Which path should the connections follow between the demand points and their assigned DUs, and also between the DUs and the CO?

These subproblems will be referred to as Group formulation, DU allocation and Connection establishment, respectively.

In general, group formulation belongs to the class of clustering problems, with several specific attributes, e.g. the clustering is carried out on a graph and not in the Euclidean space, diameter of the clusters is bounded by network range constraints, and cardinality of the clusters is fixed due to DU capacity requirements. DU allocation is obviously a location problem, more specifically a capacitated location problem defined on a graph. Connection establishment belongs to minimum cost path and flow problems, complicated with a non-linear, stepwise cost function of parallel connections.

The NTD problem will be decomposed into these subproblems; however the strong cross-dependence of them has to be taken into consideration. Moreover, that cross-dependence is what makes the problem as complete as it is:

- Group formulation strongly affects DU allocation: once the group of demand points is defined, DU allocation reduces to the problem of finding a “centroid” of that group. Therefore a confuse clustering with scattered groups by itself makes “nice” DU allocation impossible, not to mention its effect on the connection establishment, which is intended to minimize the sum of demand point-DU distances.
- DU allocation, together with the DU-demand point assignment (i.e. group formulation) almost completely determines the optimal connection establishment in the distribution network segment. Hence the optimal distribution network is treated rather as the objective function for the DU allocation, than as an independent optimization step later.

Therefore these problems cannot be solved independently, one after another. In every distinct phase and subproblem, the further steps have to be reflected in the objective. Including later phases into the optimality conditions for any of the subproblems lies in the heart of the heuristics presented in this section.
5.2. PASSIVE OPTICAL NETWORKS (PON)

In the case of passive optical networks, we have relatively permissive distance limitations: typically (far) more than K demand points are within $L_{\text{max}}$ distance from the DU locations. The DU capacity constraints are also not decisive: due to the reasonable price of passive equipment, capacity of a splitting point may be easily increased by adding DU equipment at the same location. Therefore none of the constraints has dominating effect on the solution space.

On the other hand, this special case has a complex objective function: both the DU and the distribution network costs are considered, resulting in a “duplex” function, where the weights assigned to these cost factors ($\alpha, \beta$) determine optimal solutions:

$$\min\{\alpha \cdot C_{\text{cableplant}} + \beta \cdot C_{\text{equipment}}\} \approx \min \left\{ \alpha \cdot \sum_{j \in S} \text{dist}[DU(s_j), s_j] + \beta \cdot N \right\}$$

The cable plant cost of the feeder network is significantly lower than that of the distribution network, since the number demand points exceeds the number of distribution units. However, the feeder network costs are not neglected, but the higher “freedom” during the optimization process is in the distribution segment. The developed heuristics therefore aims at a joint optimization of distribution fiber and distribution unit costs, avoiding two extremities, namely (1) when the DU costs are minimized hence the DUs are completely filled, even if distant demand points are assigned to the DU, significantly increasing distribution network costs, and (2) when the distribution fiber lengths are minimized, even if it requires a distinct DU allocated for every single demand point.

The reason for using a point-multipoint architecture (e.g. PON networks) is the fiber saving achieved in the feeder network segment, where a single fiber carries the traffic from the $K$ demand points connected to the DU. My proposed Branch Contracting Algorithm (BCA) in the group formulation phase aims at maximizing these savings. It is achieved by collecting “neighboring” demand points that share a large portion of their path towards the CO (Figure 20).

5.2.1. BRANCH CONTRACTING ALGORITHM (BCA)

All of the proposed heuristic algorithms will be described with a step-by-step description and a flowchart. Moreover, a sequence of figures extends the description of the branch contracting algorithm for PON network, showing the concept of group formulation by path sharing maximization (starting with Figure 20).
**Algorithm**: Branch Contracting Algorithm (BCA)

- **Step 1 (Initialization)**: Construct a tree $T$ in the graph based on the shortest paths between the demand points and the CO. It may be easily seen that this subgraph is a tree [63].

- **Step 2 (Group formulation)**: $U$ denotes the number of not connected demand points in the tree. Start from the farthest demand point on this tree $T$ from the CO, and move from node to node up the tree, towards the CO as its root. If the number of demand points in the current subtree exceeds a predefined threshold $Q$, stop moving up. A new demand point group is formed of the demand points in the subtree, i.e. a branch is cut from the tree, contracting its nodes to a group (this step is reflected in the name of the heuristics). Remove these nodes from the set of unconnected demand points ($U$ decreased). The threshold $Q$ is typically a function of DU capacity $E$, controlling the DU utilization rate. Repeat Step 2 until any unconnected demand point exists ($U > 0$). Figure 21 illustrates the process: the red curved arrow points to the root of the subtree, while the bright blue areas indicate members of the subtree. The upward move process terminates when the subtree is large enough, and the dark blue area shows the final group borders.
• **Step 3 (DU allocation):** For every group formulated as described in Step 2, find the best DU location, minimizing the sum of distribution network cost. The available DU locations within the boundary of group are evaluated by summarizing lengths of paths to all demand points of the group, by using the Bellman-Ford algorithm. The location with minimal summarized distance is chosen as the DU location for the group.

• **Step 4 (Connection establishment):** The demand points should be connected to the assigned DUs via their shortest paths, while connecting the DUs to the CO is a more complicated process. Due to the stepwise behavior of the cable plant costs in the feeder network, minimization of the sum of $C_0$ costs plays an important role. In order to use the minimal necessary set of links for the feeder network, the problem becomes similar to the Steiner-tree problem. With small modifications, I have used the well-known Distance Network Heuristics (DNH, [59]) to solve the Steiner-tree problem, i.e. to design the feeder network.
5.2.2. COMPLEXITY

**Theorem 5.1:** The presented BCA algorithm works with time complexity of $O(V^2 \cdot \log(V))$

**Proof 5.1:**

Preliminary assumptions (these stand for all other algorithms in this section):

- The graph models a map, therefore the maximal degree of a node is bounded from above: a crossing of two streets is a node with degree of 4. Therefore, the number of network links (undirected graph edges) is bounded by $\left(\frac{\text{maxdegree}}{2}\right)$ times the number of nodes, i.e. $O(E) \sim O(V)$
- The number of DU locations is also proportional to the number of nodes (i.e. to the population / graph size): $O(\Omega) \sim O(V)$
- For ease of notation, during the complexity analysis here, I will use $V$ instead of $|V|$

Based on these assumptions, steps 1-4 of the BCA algorithm will be analyzed, in order to identify their complexity.

- **Step 1:** During the initialization phase, the shortest path tree has to be built and the subtree information has to be stored
  - Shortest path tree: the Bellman-Ford algorithm runs in $O(V \cdot E)$ time. After that, along all the demand point-CO shortest paths, the edges have to be added to the tree, in at most $O(V \cdot V)$ steps. While adding the edges to the tree, and proceeding on the paths towards the CO, the demand points are stored at all intermediate nodes in the subtree lists.
  - The subtree lists have to be ordered by the distance of demand points from the root node of the subtree, that needs at most $O(V \cdot \log(V))$ steps.
- **Step 2:** The group formulation is composed of two basic operations, in an iterative manner:
  - In order to find the next branch to cut, the furthest leaf has to be found, which needs $O(V)$ steps due to the Bellman-Ford algorithm already executed at the initialization phase. It is followed by upward moves, in at most $O(V)$ steps, towards the CO, and finally, once the root of the subtree is found, the “deepest” $K$ demand points are selected, in linear time, since the subtree lists are ordered.
The branch cut operation requires a refresh operation at all the nodes along the path from the subtree root to the CO. Deletion of the nodes from the ordered subtree lists is done in linear time at every node, resulting in $O(V \cdot K)$ operations.

- **Step 3:** The DU allocation runs in $O(\Omega)$ steps: the sum of distance values have to be compared among all DU locations, while the distance values are already known due to the Bellman-Ford algorithm in the initialization phase.

- **Step 4:** Finally, the feeder network is constructed as a Steiner-tree either with the basic Distance Network Heuristics that runs in $O(V^2 \cdot \log(V))$ time, or with the slightly more accurate algorithm of Zelikovsky that runs in $O(V^3)$ time, providing the best known approximation with a constant factor of 1.55.

Steps 1, 3 and 4 run only once, while the Step 2 has to be executed for every group, $O(V)$ times. Therefore, the overall complexity of the BCA algorithm, if the DNH algorithm is used to build the Steiner-tree is:

$$O(V^2) + O(V \cdot (V + V)) + O(V) + O(V^2 \cdot \log(V)) = O(V^2 \cdot \log(V))$$

If the algorithm of Zelikovsky is chosen to build the Steiner-tree of the feeder network, the BCA algorithm inherits its $O(V^3)$ complexity.

5.2.3. APPROXIMATION PERFORMANCE

Factor of guaranteed approximation for the PON topology design problem is really complex to derive, since the problem itself is a composition of different problems, with varying weights of different cost factors.

The weights themselves are parameters of the proposed heuristic algorithm as well: the predefined threshold $Q$ for the minimal number of demand points served by a DU has to reflect the significance of DU costs versus the cable plant costs. The capacitated facility location (CFL) problem as an underlying problem has a worst-case approximation factor of 2.

Since this worst-case approximation factor is far from the expected quality of results for practical scenarios, the approximation performance evaluation is focused on average/expected approximation, presented in the numerical results section, via evaluation of case studies, comparing heuristics results to Linear Programming lower bounds.

5.3. ACTIVE OPTICAL NETWORKS (AON)

Active Optical Networks (AON) have their own specific features, in some aspects similar, but from other aspects, contradictory to the already discussed Passive Optical Networks (PON) and Digital Subscriber Line (DSL) networks. Distance limitations are fairly permissive, similarly to PONs, and the DUs are active, complex and expensive equipment, similar to DSL networks. The high DU costs make the AON problem primarily a clustering problem with permissive capacity and length constraints.

A minimal set of groups/clusters is in demand, minimizing the necessary number of DU equipment. At the same time, a clear structure of DU-demand point assignments ensure low cabling costs. Therefore the resulting topologies are expected to be more “regular”, regarding the size and shape of clusters, in order to avoid overlapping of neighboring groups.

The concept of Voronoi-diagrams has to be mentioned here. In general, a Voronoi diagram is a decomposition of points in a given space, according to a set of “sites”. Every point in the space is assigned to the closest one of these sites, and the points assigned to the same site form a Voronoi cell.
In our case, the Voronoi-diagram is interpreted on the graph, with the DUs as its “sites”, while the cells are the demand point clusters (groups) around the DUs, even though the DU capacity constraints will slightly distort this ideal structure.

My proposed Iterative Neighbor Contracting Algorithm (INCA) tries to create such a balanced structure of demand point clusters and DU locations, using a bottom-up clustering technique: as an initial step, all demand points are assigned to their closest DU location. In the subsequent steps, these clusters are contracted with their neighbors, in order to match DU capacities, and achieve high utilization of DU equipment.

5.3.1. Iterative Neighbor Contracting Algorithm (INCA)

The flowchart of the algorithm is depicted on Figure 25, while the step-by-step description of the algorithm is given below.

Algorithm: Iterative Neighbor Contracting Algorithm (INCA)

- **Step 1 (Initialization):** Given a list of available DU locations, assign every demand point to the closest one (resulting in a Voronoi-diagram around the DU locations). The groups of demand points assigned to these locations form a non-overlapping coverage, however the groups may be smaller/larger than the desired DU utilization.

- **Step 2a (Pre-filtering):** The desired minimal utilization of a DU equipment is $Q$, which is typically a function of the DU capacity $K$. For every DU location with more than $Q$ demand points assigned, allocate a DU, and assign the closest $K$ demand points to it. Remove these from the subsequent calculations, but the residing demand points will be handled later. Step 2a is repeated until no DU locations exist with more than $Q$ demand points.

- **Step 2b (Aggregation):** Let $\mathcal{A}$ denote the set of clusters containing less demand points than the threshold $Q$. Find a pair of neighboring groups in $\mathcal{A}$. Merge these groups, and as the cluster center, use the DU location with the smaller summarized distance from the contracted group members (closer to the “centroid” of the group). If the merged group has more than $K$ members, $K$ demand points closer to the new DU location will stay in the group, the rest will be moved to neighboring clusters. Repeat this step until any “undersized” groups may be merged, i.e. stop and go Step3 if $|\mathcal{A}| = 0$, or there are no neighboring undersized groups left.

- **Step 3 (Connection establishment):** The distribution network between the DU and the assigned demand points follows their shortest paths. The feeder network is relatively sparse, therefore designed as a Steiner-tree (with the CO and the DUs as its terminals). This Steiner-tree is constructed using the DNH algorithm, similar to the feeder segment of PON networks.
5.3.2. **Complexity**

**Theorem 5.2**: The presented INCA algorithm works with time complexity of $O(V^2 \cdot \log(V))$.

**Proof 5.2**: Based on the preliminary assumptions described at PON networks, Steps 1-3 of the INCA algorithm will be analyzed, in order to calculate their complexity:

- For Step 1, the Voronoi-diagrams have to be created. To assign demand points to their closest DU location, a single execution of the Bellman-Ford algorithm is necessary that runs in $O(V \cdot E)$ time, and then, for every node, the DU location has to be selected with minimal distance value, that requires $O(V \cdot \Omega)$ steps. In total, Step 1 needs $O(V^2)$ operations.

- By a more detailed resolution of Step 2:
  - The DU locations with more than $Q$ demand points have to be found and realized. Finding these needs $O(\Omega)$, while in order to assign exactly the closest $K$ to the new DU, these have to be ordered ($O(V \cdot \log(V))$), and then the deletion of the connected nodes can be done in linear time. In total, this happens in $O(V \cdot \log(V))$ time.
  - For aggregation of DUs:
    - The closest neighbor of the chosen DU can be found by a DFS search on $O(V + E)$ time, and for choosing the centroid of the new group, distance values from the Bellman-Ford algorithm have to summed up and compared, in linear time.
    - The actual DU and group contract operation runs in $O(K \cdot \Omega)$ time, since all the affected nodes have to be assigned to their closest available DU, that means a simple comparison of all DUs for every node. And there are no more than $2K$ affected nodes, since the two contracted groups had less than $K$ members.

- Finally, the Steiner-tree problem of the feeder network can be solved by any of the following two algorithms: the basic Distance Network Heuristics runs in $O(V^2 \cdot \log(V))$ time, while the slightly more complex algorithm of Zelikovsky runs in $O(V^3)$ time.
Steps 1 and 3 are to be run only once, while the operations of Step 2 are used in an iterative manner, in a worst case scenario $O(V)$ times. Therefore, the overall complexity of the INCA heuristics, if the DNH heuristics is used to build the Steiner-tree is:

$$O(V^2) + O(V \cdot [V \cdot \log(V) + V + V \cdot \log(V)]) + O(V^2 \cdot \log(V)) = O(V^2 \cdot \log(V))$$

If the algorithm of Zelikovsky is chosen, the INCA heuristics inherits its $O(V^3)$ complexity.

5.3.3. APPROXIMATION PERFORMANCE

Any guaranteed approximation factor is again very difficult to prove, similar to PON networks. The clustering problem itself contains the capacitated p-median problem (CPMP), supposed that the number of DUs is fixed ($p = \frac{N}{K}$). For the CPMP problem, $3 + \frac{2}{p}$ is the best known approximation factor in the literature, and anything better than $1 + \frac{2}{\epsilon} \approx 1.736$ is impossible. Both of these worst-case approximation factors are far from the expected quality of results for practical scenarios.

The observed approximation performance for practical applications and realistic case studies, I refer Chapter 7 about the numerical results and evaluation of case studies. There the heuristics are compared to Linear Programming results.

5.4. DIGITAL SUBSCRIBER LINE (DSL) NETWORKS

DSL networks with fiber feeder segment have a very special feature, which makes them different from the completely optical access network types addressed in the dissertation. The re-use of the legacy copper network makes the distribution network cost negligible, at least in comparison with the fiber deployment required by the other technologies. Therefore the cable plant cost reduces to the feeder network segment. At the same time, the physical constraints on the distribution network segment play fundamental role: the copper network is strongly limited by the attenuation characteristics of the copper itself.

The Distribution Unit (DU) costs are significant: the network equipment, on the boundary of the optical feeder and the copper distribution network are expensive. On the other hand, DU capacity constraints do not have significant effect: typically the DU capacity itself is higher than the number of demand points within the range of the DU due to the limited copper loop length, i.e. not the DU capacities themselves; rather the copper loop length constraints play an important role. Moreover, the incurring DU costs are more related to the number of active DU locations and not the equipment itself, since the capacity of a distribution unit is typically expandable by additional cards with moderate increase in price.

Therefore, the optimization problem formulation allows the minimization of the DU equipment or the active DU locations either: in the latter case, the DU capacity constraints ($K') should be relaxed (i.e. set to infinity). These altogether define a special “coverage” problem, in which a minimal number of DU locations are sought, “covering” all the demand points, i.e. every demand point has a DU located within $\frac{d_{\text{dist}}}{\text{max}}$ distance.

5.4.1. STEPWISE ALLOCATION OF CRITICAL DUS (SACD) ALGORITHM

Flowchart of the algorithm may be followed on Figure 26, and the step-by-step description is given below.

**Algorithm**: Stepwise Allocation of Critical DUs (SACD)
• **Step 1 (Initialization)**: Place a virtual DU to all available locations. Assign all demand points within $L_{dist}^{\text{max}}$ distance to the virtual DU (one demand point may be assigned to multiple virtual DUs). Compute criticality values, and perform the lexicographical criticality ordering of DUs.

• **Step 2 (DU allocation #1)**: If any virtual DU exists with more assigned demand points then a predefined threshold $T$, choose the most critical, with respect to the ordering defined above. The value of $T$ is a function of the DU capacity ($K$), controlling the desired initial utilization rate. Allocate a single DU at the chosen position, covering the $T$ most critical demand points in the respective demand point list, and remove these $T$ demand points from the graph, and from all other virtual DUs. Repeat this step until virtual DUs exist with at least $T$ demand points, otherwise go to the next step.

• **Step 3 (DU allocation #2)**: If no more virtual DUs exist with at least $T$ demand points, choose the one with the highest utilization, and allocate a DU there. Repeat this step until unconnected demand points exist, or until a predefined utilization lower bound is reached. I note that if the number of active DU locations is minimized instead of DU equipment ($K = \infty$), Step 3 will be skipped.

• **Step 4 (Connection establishment)**: The copper network between the DU and the demand points is shortest path based (and a priori defined by the existing and re-used copper network). As I have described it for PON networks, the DNH algorithm will be used to construct the Steiner-tree of the feeder network.

![Flowchart of the Stepwise Allocation of Critical DUs (SACD) Algorithm](image)

**FIGURE 26 FLOWCHART OF THE STEPWISE ALLOCATION OF CRITICAL DUS (SACD) ALGORITHM**

### 5.4.2. Complexity

**Theorem 5.3:** The presented SACD algorithm works with time complexity of $O(V^2 \cdot \log(V))$

**Proof 5.3:** Based on the preliminary assumptions described at PON networks, steps 1-4 of the SACD algorithms will be analyzed, regarding their complexity:

- By summing up the above described tasks, Step 1 needs $O(V^2 \cdot \log(V))$ steps:
  - Assigning demand points to all DU locations within reach: in order to make this assignment, an execution of the Bellman-Ford algorithm is necessary that runs in $O(V \cdot E)$ time. With respect to my initial assumptions, it is equal to $O(V^2)$.
  - Compute node criticality values: for every node-DU location pair a length-check has to be carried out, based on the distance values obtained from the Bellman-Ford algorithm: $O(\Omega \cdot V) \sim O(V^2)$. 

---

| Specialized, highly efficient heuristics | 53 |
Lexicographical ordering of DU: at first, the nodes assigned to a DU location have to be ordered in increasing order by their criticality values, that needs $O(V \cdot \log(V))$ steps per DU, $O(\Omega \cdot V \cdot \log(V)) \sim O(V^2 \cdot \log(V))$ in total. Finally, the DUs themselves require a lexicographical ordering, based on their (already ordered) node-lists, that needs $O(\Omega \cdot \log(\Omega))$ steps, i.e. $O(V \cdot \log(V))$ according to my initial assumptions.

- Choosing the next position for DU allocation (Step 2) runs in $O(1)$ time, due to the lexicographical ordering.
- The DU allocation itself (Step 3) needs at the deletion of at most $K$ points from all other virtual DUs in $O(K \cdot \Omega)$ time, and a reordering of the DUs in $O(\Omega \cdot \log(\Omega))$ steps. The demand point lists of DUs remain ordered after the deletion. By summarizing these, Step 3 needs $O(K \cdot \Omega + \Omega \cdot \log(\Omega)) = O(V + V \cdot \log(V)) = O(V \cdot \log(V))$ steps.
- Step 4: For building the Steiner-tree of the feeder network, two heuristics can be used. The simpler Distance Network Heuristics runs in $O(V^2 \cdot \log(V))$ time, while the Zelikovsky algorithm runs in $O(V^3)$ time, providing the best known approximation factor of 1.55.

Steps 1 and 4 are to be run only once, while Steps 2 and 3 are used in an iterative manner, at most $O(V)$ times. Therefore the overall complexity of the SACD algorithm, if the DNH algorithm is used to build the Steiner-tree is:

$$O(V^2 \cdot \log(V)) + O(V \cdot [1 + V \cdot \log(V)]) + O(V^2 \cdot \log(V)) = O(V^2 \cdot \log(V))$$

If the algorithm of Zelikovsky is chosen, the SACD algorithm inherits its $O(V^3)$ complexity.

### 5.4.3. APPROXIMATION PERFORMANCE

In Section 4.2 it was proven that the criticality based greedy algorithm gives an upper bound for the minimal number of necessary DU locations, i.e. the SACD algorithm also provides an upper bound in this sense. Even though the upper bound is not tight, in most of the practical situations, it may lead to a 2-approximation, according section 4.2 on matching. However, we can take a little step forward, and evaluate the approximation of the minimal amount of DU equipment, with respect to the DU capacity ($K$) constraints.

**Theorem 5.4:** Provided that the given maximal matching is of minimal size, the above described algorithm yields to a maximal “additional” 2-approximation inaccuracy of the problem, regarding the number of DU units used.

**Proof 5.4:** This suboptimality occurs when the number of demand points assigned to the DU location leads to an “underutilization” of the DU equipment. The worst case scenario is then the lowest possible utilization of DU equipment on the necessary DU locations, i.e. the case when the $k_i$ distribution units are just not sufficient to connect the demand points assigned to the respective DU location. More formally, the situation when $k_i \cdot K + 1$ demand points are connected to every active DU location, requiring $k_i + 1$ DUs there. It is important to see that the since the DU location is active in the optimal solution, at least one DU has to be assigned there ($k_i \geq 1$). Suboptimality occurs when multiple DUs are located there.

Assuming $\omega$ active DU locations, the total number of demand points is as follows:

$$S = \sum_{i=1}^{\omega} (k_i \cdot K + 1) = \omega + K \cdot \sum_{i=1}^{\omega} k_i \geq K \cdot \sum_{i=1}^{\omega} k_i$$ (1)
These demand points require a minimum of $\sum k_i$ DUs, regardless of their actual position. However, due to the $k_i \cdot K + 1$ demand points assigned to each location $i$, the number of DUs in total will be:

$$\#DU_s = \sum_{i=1}^{\omega} (k_i + 1) = \omega + \sum_{i=1}^{\omega} k_i \quad (2)$$

The proportion of the minimum necessary and actually deployed DUs, according to (1) and (2):

$$\frac{\#DU_s}{S/K} = \frac{\omega + \sum_{i=1}^{\omega} k_i}{\sum_{i=1}^{\omega} k_i} = 1 + \frac{\omega}{\sum_{i=1}^{\omega} k_i}$$

Since $k_i \geq 1$ for any location:

$$\sum_{i=1}^{\omega} k_i \leq \sum_{i=1}^{\omega} 1 = \omega$$

This implies:

$$\frac{\#DU_s}{S/K} = 1 + \frac{\omega}{\sum_{i=1}^{\omega} k_i} \leq 1 + \frac{\omega}{\omega} = 1 + 1 = 2$$

Therefore in a worst case scenario two times the minimal necessary number of DU equipment are deployed for the given set of DU locations. ■

**Corollary 5.4:** The number of active DU locations is a 2-approximation of the necessary DU equipment. If the criticality based greedy algorithm gave an $\alpha$-approximation of the necessary active DU locations, the greedy SACD algorithm provides a $2 \cdot \alpha$ approximation on the minimal amount of DU equipment, and an $\alpha$ -approximation of the DU locations.

I have to note here then is a “worst case” scenario, with specifically constructed bipartite graphs, the $\alpha$ approximation factor of the greedy algorithm may be arbitrarily high. However, this is not the typical case for generic graphs, derived from street system graphs (the origin of the problem addressed in this dissertation).
6. Exact Optimization and Metaheuristics

The heuristic solutions proposed in the previous section are highly specialized algorithms, intended to provide high quality approximate results, within reasonable time constraints. In this section, two totally different reference methods are presented: (1) a mathematical programming approach, used to find the optimality gap, i.e. the distance between the heuristic solutions and the exact optimum, and (2) Simulated Annealing, a general (meta)heuristic approach, which will be used to evaluate the scalability of the highly specialized heuristics.

The Optimality Gap

The already presented algorithms are heuristics, so we need to know how far they are from the optimum: approximation quality is a primary question. In the first subsection, I will make an effort towards exact optimization, in order to evaluate the gap between the heuristic solutions and the optimum. Through several improvements steps, including linearization and relaxation/decomposition techniques, a linear programming formulation will be derived, which provides a lower bound of the optimum, even for relatively large problem instances.

Such a lower bound then contributes to the initial question: the gap between the heuristic solutions and the optimum. Supposed that we have a lower bound, i.e. a cost value even lower than the optimum \( (OPT - \alpha) \). The heuristic solution is higher than the optimum \( (OPT + \beta) \), i.e. the optimality gap is \( \beta \). In such a case, the \( \alpha + \beta \) distance bounds the optimality gap from above, and this \( \alpha + \beta \) distance will be available even if the optimum itself remains unknown.

Specialization Gain

Estimating the optimality gap tells the price of using heuristic approaches instead of exact optimization. Unfortunately, exact optimization is beyond possibilities, as the complexity results have shown. Supposed that an acceptable optimality gap is obtainable, heuristic approaches provide a reasonable solution. However, finding the right heuristic solution is not straightforward. The proposed solutions rely on the specific problem characteristics, i.e. the special cases presented in Section 2.6. As it was discussed in the respective sections, accurate identification of these special cases determines the efficiency of the heuristics (see No Free Lunch Theorem [56]).

A large set of generally adopted metaheuristic approaches exist in the literature, e.g. Random Optimization [84], Genetic Algorithm [85], Simulated Annealing [86], Tabu Search [87], and several nature inspired algorithms: Ant Colony Optimization [88] or the Firefly Algorithm [89] are nice examples for learning from the nature. Many of them were investigated regarding their applicability for the NGA Topology Design (NTD) problem [40], and I found the Simulated Annealing (SA) [86] a well suitable approximation scheme. In the second subsection, a Simulated Annealing scheme is developed and adapted to the NTD problem.

These will serve as a “benchmark” general heuristic algorithm for performance evaluation of all presented, highly specialized heuristics, considering both approximation quality and scalability, while the numerical evaluation and the case studies will be presented in the next section.
6.1. **MATHEMATICAL PROGRAMMING**

In this section, the mathematical programming formulation is introduced, beginning with the initial Quadratic Programming (QP) formulation, and then improved through various steps towards a computationally tractable, Linear Programming (LP) formula.

6.1.1. **QUADRATIC PROGRAMMING (QP) FORMULATION**

Let me recall Section 2.4, where the formal model of the NGA Topology Design problem was introduced, and the optimization problem formulated. The problem in its most straightforward representation is a quadratic programming (QP) problem.

Let me introduce some basic assumptions for calculating the \(|\text{Variables} \times \text{Constraints}|\) dimensions of the formulation:

- for the sake of simplicity, in this section I will use \(E\) instead of \(|E|\), \(V\) instead of \(|V|\), etc. during the problem dimension calculations
- the number of edges \((E)\) and nodes \((V)\) is in the same order of magnitude: \(E = O(V)\). This initial assumption is supported by the fact that our graph represents the map (street system) of a given service area, at which the nodes are degree-constrained: most of them have exactly two adjacent edges, and street crossings typically have a degree of four
- the number of DU locations and demand points is proportional to the number of nodes: \(\Omega = O(V)\)

Without repeating the whole QP formulation and its explanations, let us simply repeat the definition of variables and the set of constraints (while the reader is kindly asked to have a look at the QP formulation in Chapter 2.4). Here we proceed with calculating the dimensions of the Quadratic Programming (QP) formulation (using the above assumptions).

**Variables**

- \(\forall i \in S, \forall j \in \Omega \quad \omega^i_j \in (0,1)\) Indicator of the demand point-DU location assignment, value is 1 only if the demand point \(i\) is connected to DU location \(j\).
- \(\forall j \in \Omega \quad n_j \in \mathbb{N}\) The number of DUs at location \(j\).
- \(\forall i \in S, \forall e \in E \quad x^i_e \in (0,1)\) Indicator of edge \(e\) on the path between demand point \(i\) and its assigned DU location (Distribution network)
- \(\forall j \in \Omega, \forall e \in E \quad y^j_e \in (0,1)\) Indicator of edge \(e\) on the path between \(DU_j\) and the CO (Feeder network)
- \(\forall e \in E \quad F(e)\) Number of connections over edge \(e\) in the feeder and in the distribution networks altogether
- \(\forall e \in E \quad I_e \in \{0,1\}\) Indicator variable for existence of edge \(e\) either in the feeder or in the distribution network segments
- \(N\) Total number of DUs deployed in the network

**Variable set dimensions**

\[
\text{Var}(QP) = S \cdot \Omega + \Omega + S \cdot E + \Omega \cdot E + E + E + 1 = S \cdot E + \Omega \cdot (1 + S + E) + 2 \cdot E + 1 - O(V) \cdot O(V) + O(V) \cdot [1 + O(V) + O(V)] + O(V) = 2 \cdot O(V^2) + O(V) = O(V^2)
\]
CONSTRAINTS

(1) \[ \forall i \in S \quad \sum_{j \in \Omega} \omega_j^i = 1 \]

(2) \[ \forall j \in \Omega \quad \sum_{i \in S} \omega_j^i \leq n_j \cdot K \]

(3) \[ \forall v \in V, \forall i \in S \quad \sum_{e: v \to} x_e^i - \sum_{e: v \to} x_e^i = \begin{cases} \omega_j^i & \text{if } v \in \Omega \\ -1 & \text{if } v = i \\ 0 & \text{otherwise} \end{cases} \]

(4) \[ \forall v \in V, \forall j \in \Omega \quad \sum_{e: v \to} y_e^j - \sum_{e: v \to} y_e^j = \begin{cases} -n_j & \text{if } v = DU_j \\ 0 & \text{otherwise} \\ +n_j & \text{if } v = CO \end{cases} \]

(5) \[ \forall i \in S \quad \sum_{e \in E} l(e) \cdot x_e^i + \sum_{j \in \Omega} \left( \omega_j^i \cdot \sum_{e \in E} l(e) \cdot y_e^j \right) \leq L_{\text{max}} \]

(5a) \[ \forall i \in S \quad \sum_{e \in E} l(e) \cdot x_e^i \leq L_{\text{dist max}} \]

(5b) \[ \forall i \in S \quad \sum_{j \in \Omega} \left( \omega_j^i \cdot \sum_{e \in E} l(e) \cdot y_e^j \right) \leq L_{\text{feed max}} \]

(6) \[ \forall e \in E \quad F(e) = \sum_{i \in S} x_e^i + \sum_{j \in \Omega} y_e^j \]

(7) \[ \forall e \in E \quad F(e) \leq I_e \cdot (S + \Omega) \]

(8) \[ N = \sum_{j \in \Omega} n_j \]

CONSTRAINT SET DIMENSIONS

\[ \text{Con}(QP) = S + \Omega + V \cdot S + V \cdot \Omega + S + S + E + E + 1 = S \cdot (V + 4) + \Omega \cdot (1 + V) + 2 \cdot E + 1 \sim O(V) \cdot O(V) + O(V) \cdot O(V) + 2 \cdot O(V) + 1 = 2 \cdot O(V^2) + O(V) = O(V^2) \]

In summary, dimensions of QP:

\[ \text{Dim}(QP) = \text{Var}(QP) \times \text{Con}(QP) = O(V^2) \times O(V^2) \]
6.1.2. MIXED INTEGER PROGRAMMING (MIP) FORMULATION

Linearization of the quadratic length constraints (5.5b) of QP is possible: both \( \omega \) and \( y \) are binary variables, therefore their product is the AND operation of Boolean algebra. A new variable \( z \) is introduced, and by two constraints, \( z = \omega \cdot y \) is enforced, as follows:

\[
\begin{align*}
z &= \omega \cdot y \\
\Rightarrow & \begin{cases} 
\omega + y \geq 2z \\
-1 + \omega + y \leq z
\end{cases}
\]

In our case, the new variable will be:

\( \forall i \in \mathbb{S}, \forall j \in \Omega, \forall e \in \mathbb{E} \quad z_{i,j}^{l/e} \in (0,1) \) Indicator of edge \( e \) being on the path between \( DU_i \) and the \( CO \) if demand point \( i \) is assigned to \( DU_j \) (Feeder network!)

Constraints (5) and (5b) are substituted by (A1)-(A4) as follows:

(A1) \( \forall i \in \mathbb{S} \quad \sum_{e \in \mathbb{E}} x_e^i + \sum_{j \in \Omega} \sum_{e \in \mathbb{E}} z_{i,j}^{l/e} \leq L_{max} \)

(A2) \( \forall i \in \mathbb{S} \quad \sum_{j \in \Omega} \sum_{e \in \mathbb{E}} z_{i,j}^{l/e} \leq t_{feed}^{max} \)

(A3) \( \forall i \in \mathbb{S}, \forall j \in \Omega, \forall e \in \mathbb{E} \quad \omega_j^e + y_e^i \geq 2 \cdot z_{i,j}^{l/e} \)

(A4) \( \forall i \in \mathbb{S}, \forall j \in \Omega, \forall e \in \mathbb{E} \quad -1 + \omega_j^e + y_e^i \leq z_{i,j}^{l/e} \)

(A3)-(A4) ensures that \( z_{i,j}^{l/e} = \omega_j^e \cdot y_e^i \), and then (5) is transformed to (A1):

\[
\sum_{e \in \mathbb{E}} x_e^i + \sum_{j \in \Omega} \left( \omega_j^e \cdot \sum_{e \in \mathbb{E}} y_e^i \right) = \sum_{e \in \mathbb{E}} x_e^i + \sum_{j \in \Omega} \sum_{e \in \mathbb{E}} \omega_j^e \cdot y_e^i = \sum_{e \in \mathbb{E}} x_e^i + \sum_{j \in \Omega} \sum_{e \in \mathbb{E}} z_{i,j}^{l/e} \leq L_{max}
\]

Similarly, (A3)-(A4) ensures that \( z_{i,j}^{l/e} = \omega_j^e \cdot y_e^i \), and then (5b) is transformed to (A2):

\[
\sum_{j \in \Omega} \left( \omega_j^e \cdot \sum_{e \in \mathbb{E}} y_e^i \right) = \sum_{j \in \Omega} \sum_{e \in \mathbb{E}} \omega_j^e \cdot y_e^i = \sum_{j \in \Omega} \sum_{e \in \mathbb{E}} z_{i,j}^{l/e} \leq t_{feed}^{max}
\]

Unfortunately, the problem dimensions are increased with this linearization. Both the variable and the constraint sets became \( V \) times larger, i.e. the problem dimensions are multiplied by \( V^2 \):

**variable set dimensions**

\[
Var(ILP) = Var(QP) + S \cdot \Omega \cdot E = O(V^2) + O(V^3) = O(V^3)
\]

**constraint set dimensions**

\[
Con(ILP) = Con(QP) + (S + S + 2 \cdot S \cdot \Omega \cdot E) - (S + S) = Con(QP) + 2 \cdot S \cdot \Omega \cdot E = O(V^2) + O(V^3) = O(V^3)
\]

In summary, dimensions of LP:

\[
Dim(ILP) = Var(ILP) \times Con(ILP) = O(V^3) \times O(V^3)
\]
6.1.3. AGGREGATED FLOWS: A REDUCED LINEAR PROGRAMMING FORMULATION

Linearization of the originally quadratic problem was a significant step forward; even if the further increased dimensions, namely the $O(V^2)$ variables and $O(V^3)$ constraints results in a really challenging problem size: an $O(1000) \times O(1000)$ size matrix representation of the problem even for a graph of 10 nodes - an enormous matrix with approximately one million elements, for a pretty small graph.

Therefore, the problem has to be reduced in size, without losing its linearity.

In this section, a new LP formulation is introduced, which is based on the aggregation of the distinct network segments into a single feeder and distribution network flow problem, respectively:

- the feeder network flow originates from the CO as its source, and terminates in the DU locations as its sinks, serving every DU location with a flow of $n_j$ units, i.e. one unit per distribution unit
- the distribution network flow originates from the DU locations as its sources, and terminates at the demand points as its sinks, serving every demand point with a flow of 1 unit

These flow problems are not independent: the DUs are treated as sinks of the feeder flow and sources of the distribution flow, while the traffic terminating in the feeder flow gives a bound on the traffic originating in the distribution flow, according to DU capacity constraints.

Such an aggregated flow problem hides the individual connections, hence the access network topology cannot be derived from it, but it gives a lower bound on the optimal value of the cost function: if the original topology design problem has a solution with cost $C$, it is also a solution for the flow problem with the same cost $C$, even though the flow problem may have a solution with even lower cost.

![FIGURE 28 AGGREGATED MIP: THE FLOW PROBLEM](image)

The length constraints are relaxed in this $MIP_{agg}$ formulation: the individual connections are not traceable, therefore their lengths are not considered. Problem complexity is further reduced by taking the linear relaxation of the problem: the flow integrality constraints are relaxed. In the aggregated flow formulation, integrality does not prevent flow splitting anyway: the variables stand only for the sum of distinct flows, not for the individual flows themselves.

According to the above described concept, the linear programming formulation for aggregated flows is as follows:
Variables

∀e ∈ E \quad x_e ∈ \mathbb{N} \quad \text{Feeder network flow over edge } e

∀e ∈ E \quad y_e ∈ \mathbb{N} \quad \text{Distribution network flow over edge } e

∀j ∈ \Omega \quad n_j ∈ \mathbb{N} \quad \text{The number of DUs at location } j

∀e ∈ E \quad I_e ∈ \{0,1\} \quad \text{Indicator variable for edge } e \text{ either in the feeder or the distribution network flows}

N \quad \text{Total number of DUs}

Objective

\text{minimize } C_{\text{flow}} = N \cdot C_{\text{DU}} + \sum_e [I_e \cdot C_{\text{u}}(e) + (x_e + y_e) \cdot C_v(e)]

Constraints

\begin{align*}
(1) & \quad \forall v ∈ V \quad \sum_{e ∈ v} x_e - \sum_{e ∈ v} x_e = \left\{ \begin{array}{ll}
N & v = CO \\
-n_v & v \neq CO
\end{array} \right. \\
(2) & \quad \forall v ∈ V \quad \sum_{e ∈ v} y_e - \sum_{e ∈ v} y_e = \left\{ \begin{array}{ll}
-1 & v \in S \\
1 & v \in \Omega
\end{array} \right. \\
(3) & \quad \forall e ∈ E \quad x_e + y_e \leq I_e \cdot (S + \Omega) \\
(4) & \quad N = \sum_{v ∈ V} n_v
\end{align*}

Constraints (1) and (2) are the flow conservation constraints for the feeder and distribution network flows, respectively, while (3) and (4) sets the auxiliary variables for the cost function: the indicator values $I_e$, and the number of DUs. I note that the value of $n_j$, i.e. the number of DUs deployed at location $j$ is implicitly constrained by (2): the DUs must “absorb” the flow originated from the demand points in $S$, then these $n_j$ values are acting as a lower bound on the feeder network flow, which is forced towards its minimum by the cost functions.

This aggregated linear programming formulation has the following dimensions:

Variable set dimensions

\text{Var}(MIP_{\text{aggr}}) = E + E + \Omega + E = 3 \cdot E + \Omega \sim O(V)

Constraint set dimensions

\text{Con}(MIP_{\text{aggr}}) = V + V + E + 1 = 2 \cdot V + E + 1 \sim O(V)

In summary, dimensions of $MIP_{\text{aggr}}$:

\text{Dim}(MIP_{\text{aggr}}) = O(V) \times O(V)

This way I have constructed a mixed integer, linear formulation with significantly lower dimensions. Obviously, this formulation does not solve the original problem directly, it rather gives a lower bound. However, even a lower bound was extremely difficult to find, which makes this $MIP_{\text{aggr}}$ formulation a valuable addition – as we will see at the validation and evaluation section later.
6.1.4. Overview of Mathematical Programming Formulations

The formulations presented in this section are compared and summarized in Table 5. The initial Quadratic Programming (QP) formulation was the direct formal interpretation of the topology design problem, as it was described in Section 2. The following Integer Linear Programming formulation (ILP) was the linearization of QP, with the same conditions, solution space and optimum. The complexity challenges were then addressed by an aggregated MIP formulation ($MIP_{agg}$), which has significantly lower dimensions.

Even if the latter, aggregated MIP formulation gives only a lower bound on the optimum instead of an exact solution, it was implemented and applied for evaluation, due to the heavy complexity challenge the topology design problem poses. The significantly lower dimension, linearization and the linear relaxation of the majority of its variables altogether made the $MIP_{agg}$ mathematical programming approach scalable enough, at least for mid-size problem instances (see Section 7 for numerical results).

<table>
<thead>
<tr>
<th>Formulation</th>
<th>Problem type</th>
<th>Dimensions</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>QP</td>
<td>Quadratic, Integer</td>
<td>$O(V^2) \times O(V^2)$</td>
<td>Exact optimization</td>
</tr>
<tr>
<td>ILP</td>
<td>Linear, Integer</td>
<td>$O(V^3) \times O(V^3)$</td>
<td>Exact optimization with linearization</td>
</tr>
<tr>
<td>$MIP_{agg}$</td>
<td>Linear, Mixed Integer</td>
<td>$O(V) \times O(V)$</td>
<td>Reduced complexity, linearization, linear relaxation</td>
</tr>
</tbody>
</table>

6.2. Metaheuristics

Several metaheuristic approaches and optimization strategies are known in the literature, e.g. genetic algorithms, tabu search, branch and bound methods, hill climbing or even the greedy algorithm. Many of these methods have been investigated, considering their applicability for the NGA Topology Design problem [40]. Simulated Annealing turned out to be the most promising alternative, among others due to its scalability, the ability to avoid local optima, and the possibility to influence the convergence speed by the temperature function [40]. It does not require numerous problem/solution instances (like e.g. Genetic Algorithm), neither enormous amount of visited previous states (like e.g. Tabu Search) to store.

In contrary to mathematical programming, Simulated Annealing does not provide exact optimum. On one hand, it will be used as a “benchmark” for the proposed heuristic algorithms. However, on the other hand, it is not used only as a benchmark, but the really flexible parameter-setting features of SA make it a promising alternative for general applications, with future technologies, significantly different costs and/or constraints, where the highly specialized heuristics suffer performance degradation. Therefore the Simulated Annealing approach is a significant part of the proposed algorithm set.

6.2.1. Simulated Annealing

The concept of Simulated Annealing comes from metallurgy: the metal is heated at first, and then the molten metal material goes through a controlled cooling process. The initial high temperature causes the atoms to move dynamically around their initial position, letting them to reach a minimum energy state later. With decreasing temperature this motion is also reduced, and finally stopped: the atoms
get stuck to their position in the crystal structure. The cooling process ensures convergence of the simulated annealing, while the high temperature range helps to avoid local minima, by allowing state transitions through lower and higher energy states. Allowing moves toward higher energy states (which is a “backward” step in a minimization process) is the most important feature of SA. The flowchart of a general Simulated Annealing process is given on Figure 29.

![Flowchart of the Simulated Annealing (SA) Process](image)

6.2.2. SIMULATED ANNEALING FOR THE NGA TOPOLOGY DESIGN PROBLEM

Applying a SA scheme is not a straightforward process. SA itself just gives a strategy, but it needs customization and adaptation to the addressed optimization problem. Such adaptation itself makes an SA implementation effective. Namely, the following subroutines have to be defined:

- cooling (temperature control) strategy
- neighbor state generation
- state evaluation
- decision function
- initial state generation
- termination

**STATE EVALUATION**

During the SA process, I will mostly concentrate on the set of Distribution Units (DUs). Assuming an a priori given DU allocation (including the number and location of DUs), the network topology is relatively easily derived, in the following steps:

- **Step 1**: Let me recall the notion of Voronoi diagrams. As an initial step, we assign all demand points to their closest DU.
- **Step 2**: In order to fulfill capacity constraints, this assignment is refined. For DUs with more than \( K \) demand points (“overloaded” DUs), the closest \( K \) demand points will be assigned to the DU, while the rest remain unconnected. Obviously, if \( n \) DUs are located at the same location, the closest \( n \cdot K \) demand points will be assigned to them.
- **Step 3**: After filling these “overloaded” DUs, the remaining demand points will be connected to the remaining DUs having spare capacity.
Based on the assignment procedure a network topology is created with respect to the given DU allocation, for which the cable/fiber costs will be calculated, and added to the cost of the DU units. The resulting overall cost will be used for state evaluation and comparison.

**Neighbor state selection**

Convergence and scalability is mostly influenced by neighbor state selection, therefore it is probably the most important part of the SA adaptation. It should support the process to walk effectively across the solution space: the number of necessary steps between the two most distant states defines the “diameter” of the solution space. I propose a neighbor state selection method based on the number and location of DUs. A neighbor state is a network topology, which is created via modification of the current topology by any of the following operations:

- **ADD**: Add a new DU to the current solution by assigning it to one of the available DU locations. The new DU location may be randomly selected, or in order to speed up convergence, it may be influenced by distribution of demand points: the new DU should be located with higher probability in a region where overloaded DUs exist.

- **DELETE**: Remove a DU from the current solution. A DU may be randomly chosen for deletion, or it may be again influenced by the distribution of demand points: a DU should be removed with higher probability from a region where underutilized DUs exist (with significantly less than \( \sqrt{\text{demand points assigned}} \)).

- **MOVE**: Move a DU from its location to a neighboring DU place, in geometric sense: the closest available DU location in the graph is selected. I note that this operation can be interpreted as a combination of an ADD and a DELETE operation of DUs.

The neighbor state is the result of a random choice among these operations. Combined with the demand point assignment process described at the state evaluation, adding a DU splits oversized groups, deleting a DU contracts undersized groups, while movement slightly rearranges the affected groups. At initial, high temperatures, multiple DUs may be involved in these operations, i.e. multiple DUs may be added, removed or moved in a single step, in order to speed up the convergence.

The number of necessary DUs in an optimal solution is typically close to \( \frac{S}{K} \), i.e. the population per DU capacity ratio. The diameter of the solution space is the maximum of necessary steps to move from one solution to another one. Removing all DUs of solution \( Y_1 \), and adding all DUs of solution \( Y_2 \) requires approximately 2 \( \times \frac{S}{K} \) steps, therefore the diameter of the solution space:

\[
\hat{d} \sim 2 \cdot \frac{S}{K}
\]

In a typical PON network scenario, with a few thousand demand points (\( S \sim 3,000 \)), and 1:64 splitting ratio (\( K = 64 \)), diameter of the solution space is in the magnitude of 100 steps.

**Temperature control**

The temperature controls the probability of a “backward” step during the minimization process, which increases overall cost. It lies in the heart of Simulated Annealing, prevents it from being stuck in a local minimum. The schedule and rate of temperature reductions give a tradeoff between quality of the solution and scalability, i.e. the convergence speed. Typically processes with faster convergence may stop earlier, with higher probability in a local minimum, while slower processes have a higher probability to reach a global optimum (or at least a “better” local minimum). Figure 30 shows an example for this phenomenon (“Kecel” case study, VDSL network design).
The temperature should decrease monotonically. Here I have chosen an exponentially decreasing function from $T = 1000$, multiplying it by an constant $\alpha < 1.0$ in every loop, which leads to a fast decrease in the beginning, and slower process at the end, around the presumed optimum (Figure 31).

**DECISION FUNCTION**

If the evaluated neighbor state ($N_1$) is better (has lower cost) than the actual state ($N_0$), the SA accepts it, and takes the step towards $N_1$. However, even if $N_1$ has higher cost (i.e. it is a *worse state*), a probabilistic decision is made, which allows move to $N_1$. This probabilistic decision is controlled by the temperature. For every decision, a random number ($R \in [0,1]$) is chosen, and compared to a predefined decision function $d$ of the temperature ($T$) and the difference between the cost of $N_0$ and $N_1$, i.e. $\Delta c = c(N_1) - c(N_0)$:

$$d(T, \Delta c) < R \Rightarrow \text{accept even a worse state}$$

Therefore, the temperature decreasing strategy, and also the decision function has to be determined. Decision function $d$ is monotonic in $T$: with decreasing temperature, the probability of a “backward” step is also decreased. I have used the following decision function:

$$\frac{\Delta c}{e^{\mu_{\text{price}}}} > R$$

Here $\mu_{\text{price}}$ is a scaling factor for compensating the difference between the price and temperature values, which can differ by several orders of magnitude.

**INITIAL STATE GENERATION**

For any iterative heuristic approach, choosing the right initial state has significant impact on convergence speed and quality of results. In the case of the NTD problem, and the above discussed realization of SA, the DUs play central role. Namely, the number and the location of distribution units must be determined at the initial state.

As a “naive” approach, the initial DU allocation follows the distribution of demand points. An optimal DU allocation that allows maximal utilization of DUs requires $|S|/K$ distribution units; therefore the algorithm will start with 20% more, i.e. $1.2 \cdot |S|/K$ DUs, in order to allow a “random walk” in the solution space at the beginning. For the initial allocation of these DUs, a weighted random selection is applied among the available DU locations ($\Omega$), where the weights of the DU locations reflect the local demand point density, i.e. the number of demand points within an $r$ radius of the given DU location.
**Termination**

The iterative process needs an “exit condition”, which stops the process. For this realization of SA, I have added two terminating conditions. The Simulated Annealing process stops:

- when $T$ falls below a certain threshold (in the current implementation, below 1.0)
- if the cost remains unchanged for a sufficiently long period of time, i.e. no neighbor states are accepted for a number of iterations

**Parameter setup**

The decision function and the temperature control together define the dynamics of the SA process; therefore, such “fine tuning” has key importance. I have set its parameters to keep the decision “alive” throughout the cooling process. Figure 31 shows the temperature decrease, the probability of an ADD/DELETE and a MOVE state transition together. These state transition probabilities have a “slow start” curve, which supports the initial steps across the solution space, in order to avoid local optima. The convergence speed is increased during the SA process. The ADD/DELETE transitions have a more “disruptive” effect, while a MOVE operation causes just moderate changes (“fine tuning”); therefore the acceptance probability decrease is slower for MOVE. The higher cost change caused by an ADD/DELETE operation is reflected in its assigned $\mu_{price}$ scaling factor. The curves are controlled via the above described parameters, namely $T$, $\alpha$ and $\mu_{price}$:

- with a lower initial $T$ values, the “slow start” phase gets shorter, while higher initial temperature makes it longer
- the $\mu_{price}$ scaling factor affects the height of the curves: higher $\mu_{price}$ values result in higher acceptance probabilities
- the $\alpha$ temperature decrease coefficient affects the number of iterations: the closer it is to 1.0, the more iterations we get

![Acceptance probability vs. temperature (AETH)](image_url)

**FIGURE 31 SIMULATED ANNEALING PROCESS DYNAMICS**

In this subsection I have proposed a metaheuristic solution, which meets the requirements and specifications of the addressed NTD problem; the distinct “building blocks” were designed to provide a reasonable search for the optimum over the solution space. The resulting universal heuristic approach tries to find an optimal solution within the domain of valid topologies “independently”: in contrary to the earlier presented, highly efficient heuristics, it is not intended to follow the network designers’ way of thinking. For this reason, SA will be a valuable “benchmark” for evaluation, and a universal approach to solve any future NTD problems which do not fit any of the presented special cases.
7. NUMERICAL RESULTS: VALIDATION AND EVALUATION

In this section, the proposed methods and algorithms are evaluated thoroughly. In the first section, the algorithms are validated with problem instances which have a regular structure. Therefore, the optimum can be calculated analytically. Results of the heuristics are then compared to that optimum.

7.1. VALIDATION: CALCULATIONS ON GRID TOPOLOGY

For validation, a regular graph structure is needed which allows analytic calculation of the optimal solution. Therefore grid topologies were used, since these are able to model a regular street system. Figure 32 is an example grid topology. Such grid topologies will be defined by four parameters:

- \( N \): number of squares along one side of the grid, i.e. the grid has \( N \times N \) squares, \((N - 1) \times (N - 1)\) street crossings
- \( L \): length of the edges of each square, i.e. the complete grid is \( L \times N \) wide
- \( k \): the number of demand points on a single side of an edge in the grid, with equal distance between their respective drop points and the street crossings
- \( d \): length of the drop cable connecting the end points to the streets, i.e. edges of the grid

The number of demand points \( S \) is expressed with these parameters. The “neighboring zone” of a street crossing is shown on Figure 32 by the colored circle: these are the demand points which have the center of the circle as their closest street crossing. There are \( 2k \) demand points along a street segment \( k \) on both sides), half of which belong to the same “neighboring zone”. Summarizing the four directions, \( 4k \) demand points fall into every zone, and there are \((N - 1)^2\) street crossings:

**8x8 Grid**

\[ N = 8 \]
\[ L = 100 \text{ m} \]
\[ k = 4 \]
\[ d = 10 \text{ m} \]
\[ S = (N - 1)^2 \cdot 4k \]  

\textbf{7.1.1. Optimal Solution: Analytic Calculations}

My intention was to define a scenario at which we can easily calculate the optimal solution. Assuming that the street crossings are the available DU locations (\( \Omega \)), a proper setting of the DU capacity allows all demand points to be connected to their closest DU location (street crossing), minimizing the distribution fiber (DF) usage. The \( 4k \) demand points within the neighboring zone have to be served by one or multiple DUs, therefore the DU capacity has to be a divisor of \( 4k \):

\[ K = \frac{4k}{\alpha} \]  

With this setup, in an optimal topology the DUs will be located at the street crossings, exactly \( \alpha \) at each location. Components of the network deployment cost for the optimal topology will be calculated in the following paragraphs.

\textbf{Distribution Units}

The minimal necessary number of DUs is given by dividing the number of demand points by the DU capacity, according to (A) and (B):

\[ \#DU_s = \frac{S}{K} = \frac{(N - 1)^2 \cdot 4k}{4k/\alpha} = \alpha \cdot (N - 1)^2 \]  

\textbf{Distribution Fiber (DF)}

Figure 33 shows the “neighboring zone” of a DU location: the fiber usage connecting all these demand points to the DU location at the street crossing will be calculated now. Assuming \( k \) demand points on a single edge of the grid, the distance between neighboring drop points (indicated by \( s \) on Figure 33) is expressed by the parameters of the grid:

\[ s = \frac{L}{k + 1} \]  

\textbf{FIGURE 33 DISTRIBUTION FIBER IN THE GRID}
Then it is easy to see that we have to connect \( k/2 \) demand points on each side of every edge adjacent to the street crossing. These drop points have \((s, 2 \cdot s, 3 \cdot s, \ldots, k \cdot s)\) distance from the DU location. The distribution fiber for a single DU region is the sum of the four directions, and two sides per direction, therefore:

\[
DF = 8 \cdot (s + 2s + 3s + \cdots + k \cdot s) = 8 \cdot \sum_{i=1}^{k/2} i \cdot s = 8 \cdot s \cdot \sum_{i=1}^{k/2} i = 8 \cdot s \cdot \frac{k}{2} \cdot \frac{k+1}{2}
\]

\[
= 8 \cdot \frac{L \cdot \frac{k}{2} \cdot \frac{k+2}{2}}{k + 1} = L \cdot s \cdot \frac{k \cdot (k + 2)}{k + 1}
\]

The drop fiber is ignored here, since the drop fiber usage is independent of the topology optimization; it would only add a constant \( d \) for all demand points.

In total, we are given \((N - 1)^2\) street crossings, therefore the total distribution fiber length in the network with this setup is:

\[
\sum_{i=1}^{k/2} DF = (N - 1)^2 \cdot L \cdot \frac{k \cdot (k + 2)}{k + 1} \quad (2)
\]

**Feeder Fiber (FF)**

In this regular setup, \( n \) DUs are located at every street crossing. The minimum necessary amount of feeder fiber will be then \( \alpha \) times the sum of distances between DU locations and the CO, which should be in a central position (see Figure 32 again). The distance between every DU location and the CO is equal to \( L \) times the sum of the absolute value of its “coordinates”, taking the CO as the origin \((0,0)\).

The \( x \) and \( y \) coordinates are then easy to read from the figure. In the middle line, the \( y \) coordinates are equally 0. In the first lines below and above, we have \( y \) coordinates equally 1, and so on – until the bottom and top of the grid, i.e. the \((\frac{N}{2} - 1)\)th lines, and in every line we have \( N - 1 \) points. If we sum up just the \( x \) coordinates at first, we get:

\[
\Sigma^x = 2 \cdot (N - 1) \cdot \left( 1 + 2 + \cdots + \frac{N}{2} - 1 \right) = 2 \cdot (N - 1) \cdot \sum_{i=1}^{N/2 - 1} i
\]

\[
= 2 \cdot (N - 1) \cdot \frac{\left( \frac{N}{2} - 1 \right) \cdot \left( \frac{N}{2} - 1 + 1 \right)}{2} = 2 \cdot (N - 1) \cdot \frac{N - 2 \cdot N}{2} \cdot \frac{N}{2}
\]

\[
= \frac{N \cdot (N - 1) \cdot (N - 2)}{4}
\]

Obviously \( \Sigma^x = \Sigma^y \). If we have \( \alpha \) DUs at each DU location, the feeder fiber usage is \( \alpha \) times the sum of \( x \) and \( y \) coordinates, i.e. \( \sum FF = \alpha \cdot (\Sigma^x + \Sigma^y) = 2 \alpha \cdot \Sigma^x \):

\[
\sum FF = \alpha \cdot \frac{N \cdot (N - 1) \cdot (N - 2)}{2} \quad (3)
\]
**Deployment Costs**

The minimal necessary cable deployment defines a subgraph, i.e. a subset of network links that preserves connectivity. It has to be a minimal set: removing any more links from the network makes it unconnected. A minimal, connected graph is a tree, by definition.

Clearly, the drop cables connecting the demand points to the street are necessary, otherwise the demand points would be disconnected from the network, i.e. there is no reason to minimize the drop cables. Therefore, here I will concentrate on links of the street system, in order to find a minimal connected set of them.

On every edge of the street system grid between two street crossings, at most one small segment may be removed from the graph. Otherwise, there will be an endpoint between the two removed segments, being disconnected from both neighboring street crossings, therefore, unreachable from any DU locations (e.g. the four central demand points on Figure 34):

![Figure 34 Connectivity of Demand Points](image)

The feeder network should connect all DUs to the CO, i.e. connectivity of the street system grid is necessary. Now if we take a look at the original graph (Figure 35), we can divide it in two different regions. The *external region* is on the boundary of the grid. Here, outside of the last demand point, \( \frac{k}{2} + 1 \) small segments are unnecessary on every external edge (bold, red segments on the figure). In the *internal region*, the street system should remain connected. In an \( N \times N \) grid we have \( (N - 1) \times (N - 1) \) street crossings. A tree connecting \( (N - 1)^2 \) nodes contains \( (N - 1)^2 - 1 \) edges, therefore \( (N - 1)^2 - 1 \) edges of the feeder network must remain intact. The remaining edges of the street system are not required for the feeder network, however due to the distribution network connectivity, not more than one segment \( s \) may be removed from each of them, as we have seen earlier.

![Figure 35 Minimal Deployment Calculations](image)
We get minimal network deployment (DP) by subtracting these two “savings” from the complete street system length, i.e. segments removed in the internal and external regions:

\[ \sum DP = DP_{total} - DP_{saving\, internal} - DP_{saving\, external} \]

The complete street system length is easy to sum up: the graph contains \((N - 1)\) “streets” both horizontally and vertically with length of \(N \cdot L\):

\[ DP_{total} = 2 \cdot L \cdot N \cdot (N - 1) \]

The external region savings are the bold red segments on Figure 35. According to the above findings, there are \(\frac{k}{2} + 1\) unnecessary small segments at the “external end” of those \(N - 1\) street system edges in all four directions:

\[ DP_{saving\, external} = 4 \cdot (N - 1) \cdot s \cdot \left(\frac{k}{2} + 1\right) = 4 \cdot (N - 1) \cdot \frac{L}{k + 1} \cdot \frac{k + 2}{2} = 2 \cdot L \cdot (N - 1) \cdot \frac{k + 2}{k + 1} \]

In the internal region, there are in total \(2 \cdot (N - 1) \cdot (N - 2)\) edges of the street system. In order to maintain connectivity of the feeder tree, \((N - 1)^2 - 1\) of them have to remain intact (as it was explained above). On the rest of the street edges, one small segment \(s\) per each edge is unnecessary. Therefore, the internal region savings are:

\[ DP_{saving\, internal} = s \cdot \left[2 \cdot (N - 1) \cdot (N - 2) - (N - 1)^2 - 1\right] \]

\[ = s \cdot (2N^2 - 6N + 4 - N^2 + 2N - 1 + 1) = \frac{L}{k + 1} \cdot (N^2 - 4N + 4) \]

\[ = \frac{L}{k + 1} \cdot (N^2 - 4N + 4) \]

In summary, the minimal cable deployment:

\[ \sum DP = DP_{total} - DP_{saving\, internal} - DP_{saving\, external} \]

\[ = 2 \cdot L \cdot N \cdot (N - 1) - \frac{L}{k + 1} \cdot (N - 2)^2 - 2 \cdot L \cdot (N - 1) \cdot \frac{k + 2}{k + 1} \]

\[ = L \cdot \left\{2 \cdot N \cdot (N - 1) - \frac{(N - 2)^2}{k + 1} - 2 \cdot (N - 1) \cdot \frac{k + 2}{k + 1}\right\} \]

\[ = L \cdot \frac{2 \cdot N \cdot (N - 1) \cdot (k + 1) - (N - 2)^2 - 2 \cdot (N - 1) \cdot (k + 2)}{k + 1} \]

\[ = \frac{L}{k + 1} \cdot (2N^2k + 2N^2 - 2Nk - 2N - N^2 + 4N - 4 - 2Nk - 4N + 2k + 4) \]

\[ = \frac{L}{k + 1} \cdot (2N^2k + N^2 - 4Nk - 2N + 2k) = \frac{L}{k + 1} \cdot (2N^2k + N^2 - 4Nk - 2N + 2k) \]

\[ = \frac{L}{k + 1} \cdot [(N^2 - 2N + 1) \cdot (2k + 1) - 1] \]

Finally we get for the minimal deployment:

\[ \sum DP = \frac{L}{k + 1} \cdot [(N - 1)^2(2k + 1) - 1] \quad (4) \]

Equations (1)-(4) give the optimal value for the number of DUs, feeder and distribution fiber usage, and cable deployment, respectively.
7.1.2. EXAMPLE GRIDS

A set of different regular grid structures were chosen for the validation of the heuristic algorithms. The main parameters of these grid networks are concluded in Table 6. The table also contains the calculated dimensions of the optimal network topology, according to the formulas of the previous section.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Grid types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>4x4</td>
</tr>
<tr>
<td>N</td>
<td>4</td>
</tr>
<tr>
<td>L</td>
<td>100</td>
</tr>
<tr>
<td>k</td>
<td>4</td>
</tr>
<tr>
<td>d</td>
<td>10</td>
</tr>
<tr>
<td>K</td>
<td>16</td>
</tr>
<tr>
<td>α</td>
<td>1</td>
</tr>
<tr>
<td>Demand points</td>
<td>144</td>
</tr>
</tbody>
</table>

Optimal solution

| Deployment          | 1600 | 8800 | 19012 | 97231 | 40480 | 87341 | 446523 |
| Feeder Fiber        | 1200 | 16800| 134400| 2688000| 168000| 1344000| 26880000|
| Distribution Fiber  | 4320 | 23520| 166024| 3184246| 108000| 762353 | 14621538|
| Distribution Units  | 9    | 49   | 196   | 784    | 225   | 900   | 3600   |

7.1.3. NUMERICAL RESULTS

Three completely different heuristic approaches were presented for three different types of Next Generation Access network technologies, namely Passive Optical Networks (PON), Active Optical Networks (AON) and Digital Subscriber Line (DSL) networks. First of all, I recall Section 2.5 about the specialization of the heuristics. For these three network types, the various cost factors are represented in the total cost with different weights, and the given physical constraints are also significantly different for each network type.

During the validation, these three heuristics, namely the Branch Contracting Algorithm (BCA), the Iterative Neighbor Contracting Algorithm (INCA) and the algorithm for Stepwise Allocation of Critical DUs (SACD) were applied for optimizing PON, AON and DSL network topologies on the grids, respectively. The calculated total costs are normalized: the analytically calculated optimum is used as 100.0%, the heuristics results are compared to it, showing the cost surplus, i.e. the difference from the optimum.

Starting with PON networks, Figure 36 shows a comparison of total cost: for the “densely populated” grids the heuristics has almost optimal results, but even the “worst case” 4x4 grid leads to only 3.6% higher cost than the optimum, which is surprisingly good from a really fast heuristic algorithm.
The total cost figures may hide the details, therefore on the following figures we look behind the curtain. Figure 37 shows the weight (contribution) of distinct cost factors to the total cost: the fiber costs are clearly dominated by the build and equipment costs. Figure 38 shows a comparison of distinct cost components of the heuristic solution to that of the optimal topology.

On Figure 39, these differences are weighted by their contribution to the total cost. We can observe that in the lowest density 8x8 grid, there is a somewhat significant difference in distribution fiber costs, while at the two other 8x8 grids with higher density of demand points, the distribution network is optimal (the same phenomenon was observed with 16x16 grids).

The reason is a tricky artifact or “side-effect” of the highly regular grid structure (Figure 40): if the heuristics creates a group which is not identical to the optimal “neighboring zone” (see Section 7.1.1), it automatically creates a malformed neighboring group. An example is the bottom-right group on the figure: four of its demand points should have been connected to its left neighbor street crossing. Because of these four points, the neighboring group will be similarly malformed, and this irregularity
propagates from group to group, all across the grid. Fortunately, with real-life scenarios, or even with more demand points and DUS within the same “neighboring zone”, this side-effect becomes negligible (see “8x8 dense”, and “8x8 dense+” grid results on Figure 38).

**Relative "material" surplus of BCA heuristics vs. optimum**

<table>
<thead>
<tr>
<th></th>
<th>Build</th>
<th>Feeder Fiber</th>
<th>Distribution Fiber</th>
<th>Distribution Units</th>
<th>Total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>8x8</td>
<td>0,0%</td>
<td>2,4%</td>
<td>48,3%</td>
<td>8,2%</td>
<td>2,8%</td>
</tr>
<tr>
<td>8x8 dense</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
</tr>
<tr>
<td>8x8 dense+</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
</tr>
</tbody>
</table>

**Cost surplus of BCA heuristics vs. Optimum total cost**

<table>
<thead>
<tr>
<th></th>
<th>Build</th>
<th>Feeder Fiber</th>
<th>Distribution Fiber</th>
<th>Distribution Units</th>
<th>Total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>8x8</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,4%</td>
<td>2,4%</td>
<td>2,8%</td>
</tr>
<tr>
<td>8x8 dense</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
</tr>
<tr>
<td>8x8 dense+</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
<td>0,0%</td>
</tr>
</tbody>
</table>
If we investigate the INCA heuristics for AON (Figure 41 & Figure 42) and SACD heuristics for DSL networks, we get even better results: the heuristics produce network topologies almost identical to the optimal solution, both in terms of total cost and every component of the total cost.

As a conclusion, on these regular grid structures the proposed heuristics performed well, having a very moderate or even no difference from the optimum at all. In the next section, the heuristics will be evaluated on large scale real world scenarios.
7.2. EVALUATION

The proposed heuristic algorithms will be evaluated against realistic scenarios in terms of approximation performance and scalability. Now we are not facing artificial problems with regular structures; hence we immediately lose the ability to calculate the optimum. At this point, we turn to the reference methods presented in Section 6. Here I recall the figure from Section 6:

![Image of reference points for evaluation]

FIGURE 43 REFERENCE POINTS FOR EVALUATION

The optimal solution is the minimal cost network that serves all the demand points with the given constraints. The Integer Programming solution provides a lower bound on it, with a cost lower by $\alpha$ than the optimum (see Section 6.1.3 for details). The Simulated Annealing provides an approximation of the optimum, with $\gamma$ distance from it, while the proposed, specialized heuristics will approximate the optimum by $\beta$. What makes evaluation difficult is the fact that we do not know the optimum itself. Therefore, we can only prove that the heuristic solution does not exceed the optimum higher than $\alpha + \beta$ – this will be the way to evaluate accuracy of the heuristics.

The case studies will be presented in Section 7.2.1. Section 7.2.2 presents the evaluation of “solution quality”, i.e. the ability of the heuristics to approximate the optimum. Section 7.2.3 is devoted to scalability aspects: resource requirements, namely running time and memory consumption of the presented methods.

7.2.1. CASE STUDIES

A set of reference areas were chosen for evaluation purposes; these will be presented in the first subsection. The proposed methodology for topology design of Next Generation Access (NGA) networks has four main input data: (1) the map, i.e. graph of the street system, (2) location of demand points, i.e. buildings and demand points, and (3) existing infrastructure information defines the reference area itself, while (4) the cost database and the physical constraints derived from technology specifications complete the scenario (Figure 44).

![Image of input data for a case study]

FIGURE 44 INPUT DATA FOR A CASE STUDY
For the presented case studies, the digital maps were derived from the publicly available OpenStreetMap database [90], and the demand point data from the Hungarian Statistical Institute. Table 7 concludes the main characteristics of the chosen scenarios. Our AccessPlan Framework (see later) applies a detailed and well-structured cost database. However, the cost data is confidential information, therefore not published in details, but we can reveal the “aggregate” cost parameters used for the topology optimization process (Table 8). The technology specifications are based on international standards for Gigabit Passive Optical Network (GPON, [11]), Active Ethernet (AETH, [17]) and Very High Bitrate DSL (VDSL, [20]) networks (Table 9). The infrastructure information and the cost databases are from joint projects with industrial partners, therefore these are not publicly available.

### Table 7: Scenarios / Case Studies

<table>
<thead>
<tr>
<th>Scenario Name</th>
<th>Köszeg</th>
<th>Kecel</th>
<th>Solymár</th>
<th>Sashegy</th>
<th>Újpalota</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Small city</td>
<td>Central</td>
<td>Countryside</td>
<td>Agglomeration</td>
<td>Suburban</td>
</tr>
<tr>
<td>Area (km²)</td>
<td>0.5</td>
<td>7.21</td>
<td>4.42</td>
<td>5.89</td>
<td>1.18</td>
</tr>
<tr>
<td>Buildings</td>
<td>367</td>
<td>3,071</td>
<td>2,134</td>
<td>1,080</td>
<td>1,080</td>
</tr>
<tr>
<td>Demand points</td>
<td>367</td>
<td>3,165</td>
<td>2,714</td>
<td>4,366</td>
<td>6,994</td>
</tr>
<tr>
<td>Demand points / building</td>
<td>1.00</td>
<td>1.03</td>
<td>1.27</td>
<td>4.04</td>
<td>6.48</td>
</tr>
<tr>
<td>Buildings / km²</td>
<td>730</td>
<td>430</td>
<td>480</td>
<td>180</td>
<td>920</td>
</tr>
<tr>
<td>Demand points / km²</td>
<td>730</td>
<td>440</td>
<td>610</td>
<td>740</td>
<td>5,930</td>
</tr>
<tr>
<td>Graph nodes</td>
<td>1,075</td>
<td>6,712</td>
<td>4,783</td>
<td>11,205</td>
<td>2,265</td>
</tr>
<tr>
<td>Graph edges</td>
<td>1,104</td>
<td>6,850</td>
<td>4,846</td>
<td>8,040</td>
<td>2,319</td>
</tr>
<tr>
<td>Street system length</td>
<td>18,040</td>
<td>97,858</td>
<td>55,703</td>
<td>152,556</td>
<td>26,660</td>
</tr>
</tbody>
</table>

### Table 8: Cost Parameters

<table>
<thead>
<tr>
<th>Access Network Technology</th>
<th>GPON</th>
<th>AETH</th>
<th>VDSL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distribution Unit Cost</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New trenching</td>
<td>C₀</td>
<td>150,000</td>
<td>4,000,000</td>
</tr>
<tr>
<td>Re-use of existing substructure</td>
<td>Cᵥ</td>
<td>800</td>
<td>10</td>
</tr>
<tr>
<td>New aerial cable</td>
<td>C₀</td>
<td>400,000</td>
<td>400,000</td>
</tr>
<tr>
<td>Re-use of existing aerial cabling</td>
<td>Cᵥ</td>
<td>800</td>
<td>10</td>
</tr>
<tr>
<td>Re-use copper cable</td>
<td>C₀</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Cᵥ</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

### Table 9: Technology Specifications and Physical Constraints

<table>
<thead>
<tr>
<th>Network technology</th>
<th>Network range (segments)</th>
<th>Distribution unit capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPON</td>
<td>L_max = 20,000 m</td>
<td>K = 64</td>
</tr>
<tr>
<td>AETH</td>
<td>L_feed = 20,000 m</td>
<td>L_dist_max = 3,000 m</td>
</tr>
<tr>
<td>VDSL</td>
<td>L_feed = 20,000 m</td>
<td>L_dist_max = 300 m</td>
</tr>
</tbody>
</table>
The first scenario is the central area of a countryside city (“Kőszeg”), which is the smallest case study, used mainly for test and visual verification, assigning exactly one demand point per building (i.e. FTDB architecture). The second case study is a countryside town (“Kecel”), mainly with single family houses (i.e. one demand point per building). The third area (“Solymár”) is located in the agglomeration of Budapest, the capital of Hungary. It has slightly different settlement structure, but the majority of the buildings is still owned by a single family. The last two scenarios are from Budapest: one “suburban” region, with a mix of different building types, containing also “empty” areas of a cemetery and a small nature reservation area, therefore it has a low population density in average (“Sashegy”). The last area is a densely populated urban region of Budapest, typically with large apartment houses (“Újpalota”): this one has the highest number of demand points among the five different reference areas.
7.2.2. APPROXIMATION PERFORMANCE

This section is devoted to evaluation of the solution quality, i.e. ability of the presented methods to minimize cost function of the NGA Topology Design (NTD) problem. Once we leave the “lab conditions”, evaluation and interpretation of the results becomes complex and sometimes not easy to understand. In this subsection, the proposed highly efficient heuristics presented in Section 5 are compared to the Mixed Integer Programming (MIP, Section 6.1) lower bound, in order to evaluate the quality of approximation, i.e. the gap between the heuristic solution and the optimum.

7.2.2.1. MIXED INTEGER PROGRAMMING (MIP) RESULTS

Results of the MIP formulation need some little explanation. Any MIP solver, during its operation, maintains an upper and a lower bound. In a cost minimization problem, the upper bound is the best known feasible, integral solution. The assumed optimal solution is somewhere between the upper and lower bounds, therefore the MIP solver tries to converge them. For large-scale, highly complex problems, the resource constraints may not allow the solver to achieve the exact optimal solution, therefore it returns the best known optimal solution, and the gap between these bounds, which adds some kind of uncertainty to the achieved results.

The MIP solver was used with settings that enforce faster upper bound convergence, i.e. finding the best known feasible integral solution. Two examples are shown here: Figure 50 presents an MIP solver process, in which the lower and upper bounds converge in approximately 15 minutes, therefore the MIP problem was solved successfully. The difference between upper and lower bound convergence speed may be observed also: the upper bound quickly approaches the optimal solution (in approx. 100 seconds), while the lower bound increases slower. Figure 51 shows a more complex optimization process: after 72 hours of computation, there is still more than 20% gap between the upper and lower bounds, even if the best known integral solution was found in the very beginning of the process.

This phenomenon was observed for all of the presented MIP results: whenever the calculations were terminated due to time or memory constraints, the upper bounds had been unchanged for a long time, i.e. the best known feasible solution was found quickly in the beginning of the process, and then the solver spent most of the time with the lower bound improvement.
The MIP results were carried out by using the most powerful MIP solver on the market: ILOG CPLEX, now developed by IBM (version 12.2). Specifications of the computer running the calculations are summarized in Table 10 – I just have to emphasize that CPLEX 12 supports parallelization, therefore all CPU cores were used during the calculations.

In the following subsections, all heuristics will be compared to MIP results. Which MIP results cannot be treated as optimal results for the original topology design problem, but they give a lower bound of it. Due to the simplified, relaxed reformulation of the complex quadratic problem, the MIP formulation may lead to even lower cost results, since the length constraints are relaxed. Therefore the distance between heuristics and MIP results is an upper bound on the distance between heuristics results and the optimum.

**TABLE 10 CPLEX & COMPUTER SPECIFICATIONS**

<table>
<thead>
<tr>
<th>CPU</th>
<th>Intel Core i5 2500K</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU speed</td>
<td>3.3 GHz</td>
</tr>
<tr>
<td>CPU cores</td>
<td>4</td>
</tr>
<tr>
<td>RAM</td>
<td>8 GB</td>
</tr>
<tr>
<td>CPLEX version</td>
<td>12.2</td>
</tr>
<tr>
<td>CPLEX search tree size</td>
<td>500 GB</td>
</tr>
</tbody>
</table>
7.2.2.2. BCA HEURISTICS FOR PASSIVE OPTICAL NETWORKS

The total cost of the network deployment is the first and foremost aspect of evaluation and comparison. On Figure 52, the BCA heuristics results, the MIP solutions and MIP lower bounds are depicted as a pair of columns for all presented case studies. The results are normalized such that MIP solution is 100% in any case. The results show that the Branch Contracting Algorithm (BCA heuristics, Section 5.2.1) provides an approximation of the MIP solution within 11.2% for all scenarios, and since the MIP solution gives a lower bound on the minimum cost, BCA may approximate the optimum even more.

The second pair of columns requires further explanation: in the “Kecel” scenario, the BCA heuristics outperforms MIP, which is normally not possible. The reason is the extreme complexity of even a simplified MIP formulation, and “Kecel” is the scenario with the highest number of buildings, i.e. demand point locations. The MIP solver has found a valid solution after 3 minutes, while it had 60% relative gap. After 24 hours of parallel calculations on four CPU cores this gap was still 57%. In the meantime, BCA has finished in 34 seconds on one CPU core, and provided better results, falling “within the gap” between upper and lower bounds of MIP. This notably scalability gain of BCA heuristics explains the results for the “Kecel” scenario.

In order to avoid confusion, let me shortly explain how the LP solver’s relative gap will be depicted on the figures. The “empty” bar close to the MIP result shows the lower bound maintained by the LP solver (CPLEX) during the calculations. When the LP solver has stopped (ran out of time or memory), the solver had a lower bound value (MIP l.b.), besides the solution it found. No better solution exists than the MIP l.b. value, however the best valid solution found is higher (MIP). The relative gap of the solver is the distance between these two, i.e. lower bound + gap = best solution found.

The aggregated total cost may hide differences in various components of the cost function, especially for components with lower weight. Which is in fact correct: the BCA algorithm was designed in a way that it preferably minimizes cost factors with higher weight. The contribution of the various factors to the total cost in a typical PON scenario is drawn on Figure 53. The distribution unit and distribution...
fiber costs are more or less balanced, both play an important role – this differentiates between PON networks, and AON/DSL networks. Build costs have the highest contribution: this is typical for an access network deployment. On the other hand, build costs are less dependent on network design than fiber or DU costs, since build costs are more predictable: optical cables will be deployed along all the streets where demand points exist. The number of DUs and optical fiber is more dependent on the structure of network, i.e. clustering of demand point groups. Therefore on Figure 54, the more variant cost factors are depicted, i.e. the equipment and fiber cost for the five different scenarios.

Figure 55 and the following figures depict an in-depth comparison of the results, showing only a moderate difference between various components for MIP and BCA. It is not surprising: according to the cost breakdown figures, both fiber and equipment costs have a non-negligible weight, and it does not allow a significant sub-optimality in any cost component, i.e. none of them could be preferred over the other. As we will see, it makes the difference between PON, and the following AON/DSL networks. Concluding these results, the BCA heuristics has proven its ability to approximate the optimum within a reasonable 10% range, both in terms of total cost and significant cost factors.
7.2.2.3. INCA HEURISTICS FOR ACTIVE OPTICAL NETWORKS

The Iterative Neighbor Contracting Algorithm (INCA, Section 5.3.1) is compared to the MIP results, as I did for BCA heuristics earlier. The total cost comparison is depicted on Figure 57. The difference between the INCA heuristics results and the MIP solutions are between 3.5% - 7.4%, while it may be even closer to the exact optimum. It opens promising possibilities for practical applications.
Besides total cost comparison, the different cost components were also investigated: the total cost summary may hide interesting details. First of all we see the weight of various cost factors on Figure 58. As expected, lower population density increases the weight of fiber plant costs, while in urban and suburban scenarios (“Sashegy” and “Újpalota”, the last two columns) Distribution Unit costs are more significant. Within fiber plant costs, build costs are dominating over fiber costs, especially over the feeder fiber costs, which are almost negligible.

![Cost component weights (AETH, case studies)](image)

Figure 59 and Figure 60 shows cost component comparison of INCA and MIP. Fiber costs are dominated by equipment and build costs, in these two cost factors a significant difference has visible effect on the total cost. INCA closely approximates MIP for both of these cost components (and also the fiber costs), which leads to high quality approximation of the total cost as well. I have to note that INCA performs well both with lower and higher population density scenarios, i.e. for “Solymár” and “Kecel” scenarios where build costs dominate over DU costs, and “Sashegy” or “Újpalota” scenarios, where DU costs are more emphasized. The adaptation to different scenarios makes the INCA heuristics attractive: such a multi-faceted optimization algorithm should not only focus on a single cost component (e.g. build costs or DU costs), but also on an arbitrary weighted combination of these.

![Comparison of cost components: MIP vs. INCA (Kőszeg, AETH)](image)
### FIGURE 60 COMPARISON OF VARIOUS COST COMPONENTS: MIP VS. INCA HEURISTICS FOR AETH NETWORKS II.

<table>
<thead>
<tr>
<th>Location</th>
<th>Build</th>
<th>Feeder Fiber</th>
<th>Distribution Fiber</th>
<th>Distribution Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Köszeg</td>
<td>3.3%</td>
<td>0.0%</td>
<td>0.4%</td>
<td>0.0%</td>
</tr>
<tr>
<td>Kecel</td>
<td>3.1%</td>
<td>0.1%</td>
<td>1.9%</td>
<td>2.3%</td>
</tr>
<tr>
<td>Solymár</td>
<td>2.0%</td>
<td>0.0%</td>
<td>1.5%</td>
<td>0.0%</td>
</tr>
<tr>
<td>Sashegy</td>
<td>-0.6%</td>
<td>0.1%</td>
<td>2.1%</td>
<td>4.3%</td>
</tr>
<tr>
<td>Újpalota</td>
<td>1.9%</td>
<td>0.1%</td>
<td>-0.2%</td>
<td>5.4%</td>
</tr>
</tbody>
</table>

#### FIGURE 61 INCA VS. MIP: EXCESS COST BY DISTINCT COMPONENTS
Figure 61 is a more complex figure: the difference between e.g. build cost results of MIP and INCA are weighted by their contribution to the total cost, and this relative addition to the cost is depicted on the figure. For example if the build cost has 25% weight in the total cost, and INCA build costs are 10% higher than MIP build costs, it means 2.5% cost surplus with respect to total cost. Now I recall Figure 57: the last, urban scenario of “Újpalota” had a MIP vs. INCA difference of 7.2%. Now, if we take a look at Figure 61, we find the reason: the equipment costs are responsible for 5.4% out of that 7.2%. However, this 5.4% is the largest difference value in the table, which is still acceptable: the scalable INCA heuristics causes not more than 5.4% excess cost (relative to the total network deploy cost) in any cost components, i.e. build, fiber or equipment costs.

7.2.2.4. SACD HEURISTICS FOR DSL NETWORKS

Evaluation of the Stepwise Allocation of Critical DUs (SACD, Section 5.4.1) heuristics begins with the comparison of total cost figures versus the MIP lower bound (Figure 62). Even if for one scenario, namely for “Solymár” we can observe an “above average” 15% difference between the MIP lower bound and the heuristics results, it is still close in the acceptable range – and for the other four scenarios we got even more promising results (0.2% - 6.8%).

While the total cost values are similar, if we look at the details, namely the distinct cost components of the MIP and SACD solutions (Figure 63 and Figure 64), we can observe a more significant difference (see e.g. the build and DU costs for the “Kecel” scenario). The SACD heuristics, and the MIP approach leads slightly different results: during the minimization process: what is save by SACD on build costs was saved by MIP on the DU costs.

Even if we did not see such difference between heuristics and MIP results for PON and AON networks earlier, the SACD heuristics still provides a high quality approximation in the total cost. The DSL topology design problem had the most complex and tight constraint set due to the copper network reach limitations. However, the simplified MIP formulation for lower bound calculations has relaxed the length constraints, i.e. it may lead to “invalid topologies”, where demand points are assigned to DUs falling outside of the copper network range. It may be the reason for SACD and MIP leading to different topologies and different split of the total costs among the distinct cost factors. The relaxation of length constraints also explains the fact that we did not see that difference for PON/AON networks, where length constraints were not that tight.
Moreover, the “clear topology structure” was not a formal constraint, even if it is a fundamental requirement for practical applications. The topologies designed by the SACD heuristics (Figure 65 for “Kecel”) fulfill this requirement (the distinct colors represent the clustering of demand points).
If we take a look at the cost component weights (Figure 66), the reason for focusing on equipment cost minimization becomes clear: DU costs dominate all other components of the cost function. This is mainly the consequence of the re-use of the existing (copper) infrastructure: cabling costs become almost negligible.

The weighted difference in the distinct cost components between the SACD heuristics and the MIP solution is shown on Figure 67, as it was shown also for AETH networks above. In general, we see moderate differences not only for the total cost, but also for the various cost components. The colored circles indicate the two most interesting case studies again: “Kecel” and “Solymár”, where the build costs (in the feeder network) had somewhat higher weight (see Figure 66). In “Kecel”, the MIP the heuristic solution leads to a different balance between equipment and cable plant build costs.

According to the cost component weights (Figure 66), build costs are more emphasized in “Kecel” than in any other case study, and that is the reason why the heuristics has just 6.5% total cost surplus compared to the MIP solution, despite the 20% higher equipment costs: savings in the build costs pay off. On the other hand, in the “Solymár” scenario, both the equipment and build costs are higher than that for the MIP solution, which leads to 15% difference in total. However, even these differences fall in a range which is still promising for practical applications, especially if we do not forget that we were not comparing the heuristics to the optimum itself, we were using a lower bound instead.
Simulated Annealing is a complex iterative heuristic algorithm, searching for the optimum potentially all over the solution space. If its resource requirements (time and memory) are met, it leads to remarkable results. The specific resource requirements for really large-scale scenarios will be investigated in the next section, but in this subsection I will concentrate on the results for the presented five case studies, for which computation time and memory needs are satisfied.

Figure 68 and the following two figures show the cost comparison for PON, AON and DSL networks respectively, now with Simulated Annealing added. The available computation time for SA may be easily adjusted through the temperature decrease strategy, therefore I have started four independent calculations for every case studies by SA. For the fastest calculation, I have allowed SA to run exactly for the same amount of time, as the BCA/INCA/SACD heuristics for GPON/AETH/VDSL networks, respectively. It is called “SA heuristics time” in the figures: the brightest green column. Then I have allowed SA to run 10 times and 100 times longer the specialized heuristics. Finally SA was used without time limit, i.e. until one if terminating conditions are met: this one is the last, darkest green column.

In general, we can expect SA to provide lower quality results than the respective BCA/INCA/SACD fast heuristics when it has strict time limits, and comparable or even better results when time limits does not affect it substantially. The achieved results meet these expectations, and also reflects scalability of the algorithms: on Figure 68, we can see the results for GPON networks, namely the comparison of the MIP lower bound, SA and BCA results. As the problem size increases (from left to right on the horizontal axis), the impact of time limits for SA becomes visible: for “Sashegy” and “Újpalota” scenarios, even a time limit which is 100 times longer than BCA computation time was not enough for SA to achieve better results – but without these time limits, SA always (slightly) outperforms BCA. We can also observe how the total cost for SA decreases monotonically when the computation times are increased (see the green bars). I also have to mention, that in a few cases, SA results fall within the MIP “gap” (e.g. for “Újpalota” scenario) – as it was explained in the earlier sections, the “empty” bar next to the MIP solution indicates the LP solver’s lower bound when it exited (at least when it is visible in the depicted range).
On Figure 69, the same comparison is depicted for AETH networks, but it reflects a somewhat lower different scalability gain of INCA vs. SA: if we let SA to work 100 times longer than the INCA heuristics, it provides comparable results even for the bigger scenarios. Finally, on Figure 70 we can compare the MIP, SA and SACD heuristics results for VDSL networks. That scalability gain of the SACD heuristics is clearly seen: apart from the smallest “Kőszeg” scenario, not only a 100 times longer computation time is insufficient for SA match the results of the SACD heuristics, but the SACD heuristics outperforms SA with virtually no time limit as well. The next section gives a deeper investigation of scalability aspects, evaluation the fast heuristics and SA against a really large-scale scenario.

According to the related work (Section 1.3), heuristics capable to handle network scenarios with 1,000+ demand points is a solid result in itself. If problem dimensions are not prohibitive, SA provides even better results than the above presented heuristics for the presented five scenarios, with up to 3,000 buildings or 7,000 demand points. The proposed Simulated Annealing scheme shows its flexibility as it performs well for all three different network types, even if the cost component weights, and also impact of the physical constraints are completely different for them. This flexibility is the most important strength of the SA based heuristics, which makes it a promising candidate for any (yet) unknown, future fixed access network technology. As an element of the algorithm set, it serves as the “Jolly Joker”, if the more specialized heuristics were not suitable for the new technology.
7.2.3. Resource Requirements

Besides accuracy evaluation, scalability requirements for the proposed algorithms add another important aspect. Computational complexity, time consumption, and the limits of applicability regarding problem size have high importance. As I have mentioned in the introductory section, the literature is rich in methods which provide acceptable solutions for small problem instances, i.e. hundreds of demand points. What about realistic scenarios with thousands or tens of thousands of demand points? This will be the last part of the evaluation.

Before presenting the running time comparison of MIP, SA and BCA/INCA/SACD for all case studies, I will try to explain the “dynamic behavior” behind running time comparison of the evaluated methods. On Figure 71 the SA and the MIP solver process is represented as a function of time: on the horizontal axis the time passed, while on the vertical axis the best solution until that time is depicted. The figure corresponds to “Sashegy” scenario with 1,000 buildings and 4,000 demand points; however the
trends are similar for every scenario. The MIP solver still had 39% gap between the lower and upper bounds after 24 hours. The Simulated Annealing approach was converging to its final solution in approximately 1.5 hours, while the highly efficient BCA heuristics provided a solution in 2 minutes of computation. The difference is significant, even in a logarithmic timescale.

The left side of Figure 72 depicts the measured complete running time on a logarithmic scale for MIP, SA and the specialized heuristics for GPON, AETH and VDSL networks, respectively. The specialized BCA/INCA/SACD heuristics are approximately 3 orders of magnitude faster than Simulated Annealing, which is again faster than MIP by approximately 2-3 or even more orders of magnitude. On the right side, scalability of the fast heuristics is shown. The trends clearly show the polynomial running time dependence from the problem size, for all three fast heuristics, according to the analytic analysis.

The resource requirement difference is significant, especially for practical applications. Moreover, from a pure theoretical point of view, the fast BCA, INCA and SACD heuristics are interesting from “l’art pour l’art” algorithmic aspect as well. Constructing fast, polynomial heuristics for the highly complex Network Topology Design (NTD) problem is not straightforward: as we have seen in the related work section, the earlier published methods had much higher resource requirements.
7.3. LARGE-SCALE SCENARIOS AND PRACTICAL APPLICATIONS

From a more practical point of view, network design is an “offline” problem, therefore time constraints are not strict: a couple of hours, or even a few days of computation is still acceptable. However, the difference between the specialized fast heuristics and SA (or even MIP) implies a decisive difference for really large-scale scenarios. Areas with $10,000^+$ demand points are beyond possibilities of MIP, and also SA has strong restrictions, due to its memory usage and computation time requirements.

For these, really large-scale scenarios, the fast and efficient BCA/INCA/SACD heuristics are viable solutions. These large-scale scenarios have practical interest: due to the network node consolidation trends, the Central Offices are serving larger and larger areas currently and in the near future. Our last case study is a complete district of Budapest (“District XII”, Figure 73). This is still not the largest possible service area for a single Central Office, but with $4,000^+$ buildings and $25,000^+$ demand points, it represents a tough algorithmic challenge (Table 11).

Considering practical applications, the MIP approach itself does not result in a valid network topology, it just gives a lower bound on the cost via an aggregated flow problem, without solving the DU-demand point assignment. It was used only for evaluation of the heuristics. For “District XII” scenario, MIP became practically useless: the GPON problem was not solved at all (no valid solution found), AETH was solved, but it still had 38% gap after 24 hours on 4 parallel CPUs. The MIP solver provided acceptable results only for VDSL (13% relative gap on exit with memory limit exceeded).

In contrary, the fast BCA, INCA and SACD heuristics delivered good results without any modification or simplification, even if evaluating solution quality is difficult, since the reference methods were almost useless here. Except for VDSL, where the SACD heuristics total cost was only 3% higher than the best known solution of the MIP problem. However these heuristics were within 10-15% from the MIP optimum for all five earlier presented case studies, therefore these are supposed to have a similar solution quality in somewhat larger scenarios as well.

The total cost comparison of SA and the fast heuristics is depicted on Figure 74, while the respective running time comparison is given on Figure 75. The specialized BCA, INCA and SACD heuristics were
extremely fast (45 seconds to 8 minutes): these clearly scale well, and are capable to solve even larger problems. Simulated Annealing, with a highly efficient and optimized implementation and carefully selected parameters set needs approximately 1-2 days for calculations: SA reached its limits there. The results also illustrate why VDSL may be treated as the most complex special case: SA, even after 2.5 days still provided poor results: approximately 100% higher cost than the SACD heuristics. For GPON and AETH networks, SA was very time consuming, but at least provided acceptable results.

### TABLE 11 SPECIFICATIONS OF “DISTRICT XII” CASE STUDY

<table>
<thead>
<tr>
<th>Scenario: District XII.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Area (km²)</th>
<th>13.56</th>
<th>Buildings</th>
<th>4 151</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graph nodes</td>
<td>19 331</td>
<td>Demand points</td>
<td>26 489</td>
</tr>
<tr>
<td>Graph edges</td>
<td>20 136</td>
<td>Demand points/building</td>
<td>6.38</td>
</tr>
<tr>
<td>Street system length (m)</td>
<td>303 927</td>
<td>Buildings / km²</td>
<td>310</td>
</tr>
<tr>
<td>Demand points / km²</td>
<td>1950</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Numerical results: Validation and Evaluation

**FIGURE 74** TOTAL COST COMPARISON ON A LARGE SCALE SCENARIO (DISTRICT XII)

**FIGURE 75** RUNNING TIME COMPARISON ON A LARGE SCALE SCENARIO (DISTRICT XII)
8. CONCLUSION

8.1. CONTRIBUTIONS

THEORETIC BACKGROUND

The first necessary step towards algorithmic topology design for Next Generation Access (NGA) networks was to clarify its theoretic background. Therefore, I have defined a formal model for the NGA Topology Design (NTD) problem. The parameterized graph model is technology agnostic, i.e. with the adequate parameter settings it models all point-to-multipoint and also point-to-point NGA network technology types. Considering these different concepts or technology types, I have also identified the respective special cases of the NTD problem for Passive Optical Networks (PON), Active Optical Networks (AON) and Digital Subscriber Line (DSL) networks, and as a “degenerate” special case, point-to-point networks also fit in the scope of the model.

The optimization problem was formulated for the NTD problem in general, and also for its special cases. Based on the formal model, the mathematical problem of topology design for NGA networks was analyzed algorithmically. I have identified complexity and approximability of the addressed problems. It was fundamental for defining reasonable requirements for the proposed algorithms. The problem in general, and all of its special cases were proven to be NP-hard, and their approximability features were also clarified. The in-depth investigation of the mathematical problem and its graph representation lead to recognition of key features of the modeling graph. The notion of criticality, the most critical nodes and distribution unit locations highlighted the “pillars” of the topology.

HIGHLY SPECIALIZED HEURISTICS

I have analyzed the effect and significance of various physical constraints and cost components, and then I have proposed a set of highly specialized and effective heuristic algorithms for the identified special cases of the NTD problem, based on the underlying theoretic work.

A common feature of the heuristics is the decomposition of the topology design problem. The subproblems are then solved with respect to the strong cross-dependence among them. The presented BCA, INCA and SACD heuristics use different techniques to solve these underlying problems, according to the specialties of the respective NGA technology types. The BCA heuristics for PON networks utilizes a tree-based segmentation technique, the INCA heuristics for AON networks is built on a bottom-up clustering concept, while the SACD heuristics for DSL networks is using a top-down clustering approach. However, the fact that the clustering is carried out on a graph, with a priori given cluster sizes differentiates this problem from the traditional clustering problems.

The proposed methods have proven their ability to handle large-scale scenarios even with 10,000 demand points with moderate time consumption (within one hour even for the largest scenario), and at the same time, delivered typically 10% or sometimes even better approximation of the optimum. The scalability has key importance, it makes these highly efficient heuristics a valuable contribution to the existing literature, not to mention the practical consequences, i.e. their possible applications.

EXACT OPTIMIZATION & METAHEURISTICS

Obviously, there is “no free lunch”, according to a principle of optimization. The fast heuristics deliver their peak performance for the respective special cases of the topology design problem, but may perform poorly within significantly different circumstances.
An impressive variety of metaheuristic concepts exist in the literature. Some of them were reviewed considering applicability for the addressed access network topology design problem [40]. The concept of Simulated Annealing turned out to be a promising strategy. Its building blocks, i.e. the neighbor state generation routine, the cooling strategy and the decision functions were adapted to the problem. The careful adaptation and a highly effective implementation lead to a complex heuristic solution for the topology design problem. The evaluation has proven that it delivers high quality results with acceptable resource requirements for scenarios with up to thousands of demand points. 

Exact optimization is the ultimate goal for every optimization problem. Unfortunately, as for many interesting problems, exact optimization was beyond possibilities due to the complexity and typical dimensions of the problem: scenarios of practical interest contain thousands of demand points and tens of thousands graph nodes. Mathematical programming as a universal tool for solving optimization problems was applied, however it was not trivial. The original formulation of the problem derived from its formal representation turned out to be quadratic. The linearization further increased complexity: the matrix describing the Integer Linear Programming problem had $V^3 \times V^3$ elements, completely excluding its application even for the smallest test networks. A proper transformation of the problem lead to a linear representation with $V \times V$ dimensions, and this step permitted exact optimization. It was a crucial step for evaluation of the heuristics: even if this MIP formulation does not lead to a complete network topology itself, solving the flow problem defines a lower bound for the minimal network deployment cost, hence it was used as a benchmark for the heuristic solutions.

Validation & evaluation

The proposed heuristics were validated against regular grid structures, where they approximated the analytically proven optimum within the 10% range. Finally, during the evaluation against real-world case the proposed methods have shown approximation performance and scalability that fulfills the requirements, and offers promising applications also in realistic conditions.

8.2. Applications

Even if the work presented in this dissertation is partly theoretical, the original questions, motivating the research are of practical interest. Next Generation Access (NGA) networks are just at the beginning of a rapid growth: the technology is now mature, but the huge investment needs delayed network deployments. Both optimized network planning and techno-economic evaluation supports the process: minimization of network deployment costs is key for profitable investments, while techno-economic evaluation is an integral part of the decision making about the service area and network technology for deployment.

In our recent publications we have proven the significant accuracy gain of a network deployment cost estimation methodology based on my topology design heuristics over the existing geometric modeling approaches [53].

8.2.1. NGA Designer Framework

With my colleagues, I have developed a complex framework was developed during the last few years, based on my the theoretic results in the modeling, and my proposed heuristics and algorithms outlined in this dissertation. The framework supports automatic strategic network design and techno-economic evaluation and comparison of NGA network technologies. The numerical results presented
in the validation and evaluation section of the dissertation were not possible to achieve without the framework.

Geospatial data, technology specific physical constraints, and cost values are brought together within the framework and then the presented heuristics are used for automatic topology design. The resulting network topologies are later analyzed not only for network deployment costs (CAPEX), but also a more detailed business case evaluation module is integrated into the framework.

<table>
<thead>
<tr>
<th>DATA PROCESSING MODULE</th>
</tr>
</thead>
<tbody>
<tr>
<td>• GIS data (various map formats)</td>
</tr>
<tr>
<td>• Infrastructure information</td>
</tr>
<tr>
<td>• Demand point data</td>
</tr>
<tr>
<td>• Cost database</td>
</tr>
<tr>
<td>• Technology specifications</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NETWORK MODELING MODULE</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Graph model</td>
</tr>
<tr>
<td>• Parameters</td>
</tr>
<tr>
<td>• Initialization</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OPTIMIZATION MODULE</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Topology design</td>
</tr>
<tr>
<td>• Clustering of subscribers</td>
</tr>
<tr>
<td>• Efficient, scalable heuristics</td>
</tr>
<tr>
<td>• Provides network data for analysis</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ANALYSIS MODULE</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Visualization of network topology</td>
</tr>
<tr>
<td>• Statistics and reports</td>
</tr>
<tr>
<td>• Deployment cost: Bill of Material</td>
</tr>
<tr>
<td>• Business Case Analysis</td>
</tr>
<tr>
<td>• Techno-economic evaluation and comparison of NGA technologies</td>
</tr>
<tr>
<td>• Feasibility studies, decision support</td>
</tr>
</tbody>
</table>

8.3. TECHNO-ECONOMIC STUDIES

The Assessment Framework, built on the theoretic background, graph model and algorithms presented in this dissertation, was used in the recent years for numerous techno-economic studies, assessing the various next generation optical access (NGA / NGOA) network technologies.

Besides the most straightforward measure, i.e. the total deployment cost of a GPON, AETH, VDSL or P2P network for a selected area, the use of the strategic network design during the cost estimation process gives valuable “insights”. Complete knowledge about the topology of the physical infrastructure not only shows the estimated cost difference between the various technologies (Figure
76), but also reveals the reason why a certain technology performs better than the other – under given circumstances. An example of such insights is to compare the contribution of various cost factors for each distinct technology. A set of cost breakdown figures (Figure 77) for the “Sashegy” scenario is presented below, which highlights which cost components are responsible for the observed difference. Regarding the depicted results, I have to recall here that the technology-independent deployment trenching costs are excluded from the results, i.e. the otherwise dominating build costs only cover the cable installations.

FIGURE 76 DEPLOYMENT COST COMPARISON OF GPON, AETH AND VDSL ON THE “SASHEGY” SCENARIO

I have to note here, however, that the accuracy of such economic comparisons highly relies on the accuracy of input data, which is confidential, and also varies from country to country. That is the reason why such results were not included in the “theoretic” results presented in this dissertation.

8.4. EXPERIENCE & REFERENCE

The methodology and the framework was used within several research and R&D projects in the recent years. We were involved in a Joint Activity of the EU FP7 BONE project, targeting techno-economic evaluation of optical access networks, and we were also collaborating with Magyar Telekom within OASE (Optical Access Seamless Evolution) EU FP7 IP project. Recently, we have been invited in
COMBO, an EU FP7 Integrated Project consortium about convergence of fixed and mobile broadband access/aggregation networks, due to our map-based techno-economic evaluation methodology.

Part of the case study data is from joint R&D projects about optical access network evaluation and comparison with Magyar Telekom, the incumbent network operator in Hungary. Finally, once the methodology was mature, part of the framework development was done within a joint project with NETvisor, a telecommunication software development company.

9. ACKNOWLEDGEMENTS

I am thankful to my supervisor, Tibor Cinkler: first of all I have started my Ph.D. studies because of the wide set of interesting problems and challenges I had to face during my MSc thesis work, which was also supervised by him. He taught me a lot about the vital ability to approach and solve problems, for having fruitful discussions, and supporting me both personally and professionally.

I would like to thank also the invaluable help and support of Géza Paksy, who had guided me in the field of optical access networks, shared all his knowledge with me, supervised, sometimes criticized, sometimes completed my ideas, guided my attention to the most important issues, and most of all, always had his door open for me.

I wish to thank Prof. Michal Pióro for receiving and advising me at Dept. of Electrical and Information Technology, Lund University (Sweden), my studies in the field of optimization and especially linear programming were decisively useful throughout my Ph.D. research activity. The discussions, joint research and publications with Lena Wosinska, Jiajia Chen, Miroslaw Kantor and Bart Lannoo in the frame of the NoE BONE project gave me the inspiring experience of working together with an international group of researchers.

Last but not least, I would like to express my thanks to all my wife, family and friends for their support, and also for the moments which made me smile and relax in the last 5 years spent with the presented research.
10. REFERENCES

[12] ITU-T Recommendation G.983.x family on Broadband passive optical network (BPON), 2005
[13] IEEE 802.3ah Ethernet Passive Optical Networks (EPON), 2004
[17] IEEE 802.ah Ethernet in the First Mile, 2004
[18] ITU-T Rec. G.985 on 100Mbit/s point-to-point Ethernet based optical access system, 2003
[19] ITU-T Recommendation G.986.1 on Gbit/s point-to-point Ethernet based optical access system, 2010
[22] ITU-T Rec. G.992.1 Asymmetric digital subscriber line 2+ (ADSL2+), 2003
[41] FTTH Council Europe, ”FTTH Business Guide”, 2010
[46] B. Van Den Bossche, R. Meersman, J. Vanhaverbeke, A. Schoutteet, “Maximizing the return on investment for FTTX-rollout through the use of GIS street maps and geomarketing data”, CTTE 2010, Ghent, Belgium
References

[74] Raz, Ran; Safra, Shmuel, "A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP", 29th annual ACM symposium on Theory of computing, pp. 475–484, ACM, 1997
[90] OpenStreetMap (OSM) – http://www.openstreetmap.org/