



Technical University of Munich  
Department of Electrical and Computer Engineering  
Chair of Renewable and Sustainable Energy Systems

## Open Source Modelling and Optimisation of Energy Infrastructure at Urban Scale

**Dipl.-Ing. Johannes Florian Franz Clemens Dorfner**

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik  
der Technischen Universität München zur Erlangung des akademischen Grades eines

**Doktor-Ingenieurs (Dr.-Ing.)**

genehmigten Dissertation.

**Vorsitzender:**

Univ.-Prof. Dr.-Ing. Wolfgang Utschick

**Prüfer der Dissertation:**

1. Univ.-Prof. Dr. rer. nat. Thomas Hamacher
2. Univ.-Prof. Dr.-Ing. Ulf Schlichtmann

Die Dissertation wurde am 21.12.2015 bei der Technischen Universität München eingereicht  
und durch die Fakultät für Elektrotechnik und Informationstechnik am 18.10.2016 angenommen.



## **Abstract**

The transition towards more renewable energy use and higher energy efficiency requires a rethinking of urban energy supply. One of the major design decisions that energy system modellers need to address is the suitability of urban areas for central or decentralised power and heat generation. Another impact that current models have to handle is assessing the impact of storage technologies on the overall energy system. In order to handle these complexities, continuous development of energy system models with different focal points is necessary.

This thesis presents a suite of mathematical optimisation models adapted to energy systems. These models allow to explore the design space of energy infrastructure on an urban scale. The heat and electricity models put emphasis on capacity expansion planning, unit commitment, and design of minimum cost energy distribution networks. In addition to exhaustive mathematical descriptions, two of the models are also published online under open source licenses. Four case studies demonstrate the use of these models in real-world scenarios. For two of the four presented case studies, all input data and processing steps are also shared to allow the reproduction of results and facilitate continued research. The case studies show that the combined planning of the whole energy system can uncover still untapped potential synergies.

## **Zusammenfassung**

Die verstärkte Nutzung von erneuerbaren Energien sowie die steigende Energieeffizienz machen ein Umdenken in der Planung städtischer Energieversorgung nötig. Welche Stadtgebiete für zentrale oder aber dezentrale Strom- und Wärmeversorgung geeignet sind, ist eine der großen Fragen, auf die Energiesystemmodelle Antworten finden können. Ein weiterer Aspekt, der zunehmend an Bedeutung gewinnt, ist die Auswirkung von Energiespeichern auf das gesamte Energiesystem. Um diese Komplexität beherrschen zu können, ist eine kontinuierliche Entwicklung von Energiesystemmodellen mit unterschiedlichen Schwerpunkten erforderlich.

Diese Arbeit präsentiert eine Folge von mathematischen Optimierungsmodellen für Energiesysteme. Diese Modelle erlauben es, den Raum möglicher Entwürfe für städtische Energieinfrastruktur zu analysieren. Die Modelle minimieren die Gesamtkosten zur Ausbau- und Einsatzplanung sowie für den Entwurf von Verteilnetzen für Strom und Wärme. Neben kompletter mathematischer Dokumentationen sind zwei der Modelle auch online unter Open Source Lizenzen veröffentlicht. Vier Fallstudien demonstrieren die Anwendung der Modelle auf Städte und Stadtteile. Für zwei der Fallstudien sind zusätzlich alle Eingabedaten und Analyseschritte verfügbar gemacht, um die Nachvollziehbarkeit und Nutzung der Ergebnisse für weitere Forschung zu erleichtern. Die Fallstudien zeigen, dass die kombinierte Planung städtischer Energiesysteme bislang ungehobene Synergiepotenziale erschließen kann.

## **Acknowledgements**

This thesis is the result of work as a researcher at Technical University of Munich in the time from 2010 to 2016. The first years I spent at the Chair of Energy Economy and Application Technology. The last years I worked at the Chair of Renewable and Sustainable Energy Systems. In between I spent 9 months at TUM CREATE in Singapore.

I am especially thankful to my supervisor, Prof. Thomas Hamacher, for his support and advice during all those years. By bringing me into contact with diverse topics, I got enough time to find my main research interest.

I would like to express my gratitude towards all colleagues I had the pleasure to work with during those years, both in Munich and abroad. I especially thank Peter Böhme for hours of good discussions; Karl Janker and Rita Dornmair for being the best roommates one could wish for during one's final year of writing – always there with good suggestions, delightful conversation and enlightening suggestions. Philipp Kuhn gave helpful advice in modelling questions, and I could always rely on Simon Herzog for trivia and not missing deadlines in our common project. Special thanks go to Dennis Atabay and Thomas Zipperle, whose early adoption of and contribution to my work gave me confidence in being on the right way.

For thorough proof reading and editorial suggestions, I am very thankful to Peggy Bergmann, Peter Böhme, Magdalena Dorfner, Patrick Krystallas, and Konrad Schönleber.

Finally, I am most thankful to my family for their love and support, and for accepting my more and more English demeanour.



# Contents

<b>Abstract</b>	<b>1</b>
<b>Acknowledgements</b>	<b>2</b>
<b>Contents</b>	<b>3</b>
<b>List of Figures</b>	<b>6</b>
<b>List of Tables</b>	<b>8</b>
<b>List of Algorithms</b>	<b>9</b>
<b>List of Listings</b>	<b>10</b>
<b>1 Introduction</b>	<b>11</b>
1.1 Motivation . . . . .	12
1.2 State of the art . . . . .	12
1.3 Contribution . . . . .	15
1.4 Legend . . . . .	17
<b>2 Method</b>	<b>19</b>
2.1 Urban infrastructure . . . . .	20
2.1.1 On the term infrastructure . . . . .	20
2.1.2 Overview on urban infrastructure networks . . . . .	21
2.2 Mathematical optimisation . . . . .	25
2.2.1 A short primer on mathematical modelling . . . . .	25
2.2.2 Mathematical formulations of energy model components . . . . .	29
2.3 Literature review on energy system models . . . . .	36
2.3.1 Reference energy system . . . . .	36
2.3.2 Network flow models . . . . .	39
2.3.3 Urban energy system models . . . . .	40
2.3.4 Closed source models . . . . .	41
2.3.5 Open source models . . . . .	41
2.4 Algorithms . . . . .	43
2.4.1 Optimisation . . . . .	43
2.4.2 Load duration curve clustering . . . . .	44
2.4.3 Geographic operations . . . . .	46
2.5 Software . . . . .	51
2.5.1 Modelling languages . . . . .	51

2.5.2	Data management . . . . .	55
2.5.3	Solvers . . . . .	56
2.5.4	Final tool chain . . . . .	57
2.6	Data . . . . .	59
2.6.1	Technical . . . . .	59
2.6.2	Economic . . . . .	59
2.6.3	Time series . . . . .	60
2.6.4	Geographic information . . . . .	61
<b>3</b>	<b>Models</b>	<b>63</b>
3.1	URBS (SISO processes) . . . . .	65
3.1.1	Origin . . . . .	65
3.1.2	Conceptual overview . . . . .	65
3.1.3	Example model result . . . . .	66
3.1.4	Mathematical description . . . . .	66
3.2	urbs (MIMO processes) . . . . .	77
3.2.1	Motivation for development . . . . .	77
3.2.2	Design goal . . . . .	77
3.2.3	Example model result . . . . .	77
3.2.4	Mathematical description . . . . .	78
3.3	DHMIN . . . . .	91
3.3.1	Origin . . . . .	91
3.3.2	Design goal . . . . .	92
3.3.3	Mathematical description . . . . .	92
3.4	dhmin (seasonal) . . . . .	98
3.4.1	Motivation for model development . . . . .	98
3.4.2	Main design idea . . . . .	98
3.4.3	Example result . . . . .	98
3.4.4	Mathematical description . . . . .	100
3.5	rivus . . . . .	107
3.5.1	Motivation for model development . . . . .	107
3.5.2	The main idea . . . . .	107
3.5.3	Mathematical description . . . . .	108
3.5.4	Possible model extensions . . . . .	116
<b>4</b>	<b>Case studies</b>	<b>117</b>
4.1	District heating network (Frankfurt) . . . . .	119
4.1.1	Input data . . . . .	119
4.1.2	Scenario development . . . . .	120
4.1.3	Results . . . . .	120
4.2	District heating network (Munich west) . . . . .	122
4.2.1	Data preparation . . . . .	123
4.2.2	Scenarios . . . . .	125
4.2.3	Results . . . . .	126
4.2.4	Discussion . . . . .	127

4.3	District cooling network (Munich centre) . . . . .	129
4.3.1	Input data . . . . .	129
4.3.2	Results . . . . .	131
4.3.3	Discussion . . . . .	133
4.4	Multi-commodity network (Haag) . . . . .	135
4.4.1	Data preparation for rivus . . . . .	136
4.4.2	Results model rivus . . . . .	139
4.4.3	Data preparation for urbs . . . . .	141
4.4.4	Results model urbs . . . . .	148
4.4.5	Closing the loop . . . . .	150
4.4.6	Sensitivity analysis: battery investment costs . . . . .	150
4.4.7	Discussion . . . . .	152
<b>5</b>	<b>Conclusion</b>	<b>153</b>
5.1	Summary . . . . .	153
5.2	Limitations . . . . .	154
5.3	Outlook . . . . .	154
<b>A</b>	<b>Common model structure</b>	<b>155</b>
<b>B</b>	<b>MathProg storage sizing model</b>	<b>161</b>
<b>C</b>	<b>Time series plots for Haag i. OB</b>	<b>165</b>
<b>D</b>	<b>Symbols</b>	<b>185</b>
<b>E</b>	<b>Acronyms</b>	<b>189</b>
<b>F</b>	<b>Bibliography</b>	<b>193</b>

# List of Figures

1.1	World population in rural and urban areas from 1950 to 2050 . . . . .	12
1.2	Systems diagram of a city and supporting region . . . . .	13
1.3	Opposing directions from which to tackle the MES design problem . . . . .	14
1.4	Thesis models on spatial and temporal resolution . . . . .	16
2.1	Frequency of keywords in published books between 1900 and 2008 . . . . .	20
2.2	Gas network simulation software . . . . .	22
2.3	LP, IP and MILP in comparison . . . . .	27
2.4	Model element: time . . . . .	30
2.5	Derivation of a discretised electricity load curve for a day . . . . .	31
2.6	Discretisation of multiple load curves . . . . .	32
2.7	Model element: conversion processes . . . . .	32
2.8	Model element: energy transmission . . . . .	34
2.9	Model element: storage . . . . .	35
2.10	MARKAL model RES diagram . . . . .	37
2.11	Graphical overview on the original method URBS . . . . .	39
2.12	Annual load duration curve discretised by 15 samples . . . . .	44
2.13	Steps of the near operation for finding closest edges . . . . .	48
2.14	Steps of the skeletonization process for deriving a street graph . . . . .	50
2.15	Modelling tool chain with stages and exemplary software packages . . . . .	51
2.16	Modelling tool chain with software used in this thesis . . . . .	57
2.17	Clusters of cholera cases in the London epidemic of 1854 . . . . .	61
3.1	<b>URBS</b> model RES diagram . . . . .	65
3.2	<b>URBS</b> example plot, base scenario . . . . .	68
3.3	<b>URBS</b> example plot, cooperative scenario . . . . .	69
3.4	Process capacity constraints . . . . .	74
3.5	Input/output flow chart of model <b>urbs</b> . . . . .	77
3.6	<b>urbs</b> model RES diagram . . . . .	78
3.7	Example model site locations, with storage position . . . . .	78
3.8	<b>urbs</b> example plot, base scenario . . . . .	80
3.9	<b>urbs</b> example plot, CO <sub>2</sub> limit scenario . . . . .	81
3.10	Comparison of discrete and continuous pipe capacity costs . . . . .	91
3.11	Input/output flow chart of model <b>DHMIN</b> . . . . .	92
3.12	Example street graph with 8 vertices, 14 edges, and 2 source vertices . . . . .	93

3.13	Investment and O&M costs for a single edge $e_{ij}$ over $\pi_{ij}^{\text{in}}$ . . . . .	96
3.14	Sankey diagram of equation <i>demand satisfaction</i> (3.73) . . . . .	96
3.15	Input/output flow chart of model <b>dhmin</b> (seasonal) . . . . .	98
3.16	<b>dhmin</b> (seasonal) example result . . . . .	99
3.17	Investment and O&M costs for a single arc $a_{ij}$ over pipe capacity $\bar{\pi}_{ij}$ . . . . .	104
3.18	Sankey diagram of equation <i>demand satisfaction</i> (3.85) . . . . .	105
3.19	Input/output flow chart of model <b>rivus</b> . . . . .	107
3.20	Stacked graphs of street networks for demand and supply . . . . .	108
4.1	Location of case studies in Germany . . . . .	118
4.2	Target district heating network in Frankfurt . . . . .	120
4.3	Relative cost error in case study Frankfurt . . . . .	121
4.4	Area of investigation for case study Munich west . . . . .	122
4.5	Input data for case study in Munich West . . . . .	124
4.6	Optimal district heating network topologies $\bar{\pi}_{ij}$ in Munich west . . . . .	128
4.7	Total supplied heat in Munich west for different connect quotas. . . . .	128
4.8	Possible cooling stations and cooling demands in case study Munich centre . . . . .	130
4.9	Optimal district cooling pipe capacities in Munich centre . . . . .	132
4.10	Optimal DC flow in time step <i>All sources working</i> . . . . .	133
4.11	Optimal DC flow in time step <i>outage of an Isar source</i> . . . . .	134
4.12	Model coupling of <b>rivus</b> (network sizing) and <b>urbs</b> (storage scheduling) . . . . .	135
4.13	Map of buildings in Haag i. OB . . . . .	137
4.14	Aggregated building areas in Haag i. OB . . . . .	138
4.15	Map of peak demands for electricity and heat in Haag i. OB . . . . .	140
4.16	Map of peak demands for electricity and heat in Moosham . . . . .	141
4.17	Transmission and conversion capacities in Haag for three scenarios . . . . .	142
4.18	Map of 12 clusters for Haag i. OB . . . . .	143
4.19	Best matching TMY3 station locations for Haag i. OB . . . . .	144
4.20	Normalised load curves for TMY3 station Olympia Airport . . . . .	146
4.21	Capacity factor time series of solar and wind power for Haag i. OB . . . . .	147
4.22	Installed battery storage capacity by cluster in scenario CHEAP BATTERY . . . . .	148
4.23	Comparison of cumulated installed capacities with and without peak reduction . . . . .	150
4.24	Installed battery storage capacity by investment costs in Haag i. OB . . . . .	151

# List of Tables

2.1	Examples of infrastructure in different domains . . . . .	21
2.2	Pipe diameter for different flow speeds and temperature differences . . . . .	24
2.3	Comparison of optimisation techniques . . . . .	27
3.1	Model comparison of <b>URBS</b> , <b>urbs</b> , <b>DHMIN</b> , <b>dhmin</b> (seasonal), <b>rivus</b> . . . . .	63
3.2	<b>URBS</b> model sets . . . . .	67
3.3	<b>URBS</b> technical model parameters . . . . .	70
3.4	<b>URBS</b> economic model parameters . . . . .	71
3.5	<b>URBS</b> model variables . . . . .	72
3.6	<b>urbs</b> model sets . . . . .	79
3.7	<b>urbs</b> technical model parameters . . . . .	84
3.8	<b>urbs</b> economic model parameters . . . . .	85
3.9	<b>urbs</b> model variables . . . . .	86
3.10	<b>DHMIN</b> Model Parameters . . . . .	94
3.11	<b>DHMIN</b> Model Variables . . . . .	94
3.12	<b>dhmin</b> (seasonal) model parameters . . . . .	103
3.13	<b>dhmin</b> (seasonal) model variables . . . . .	103
3.14	<b>rivus</b> model sets . . . . .	109
3.15	<b>rivus</b> technical model parameters . . . . .	111
3.16	<b>rivus</b> economic model parameters . . . . .	111
3.17	<b>rivus</b> model variables . . . . .	113
4.1	Building Types and Heat Demand Parameters . . . . .	125
4.2	Parameter values for the case study in Munich west . . . . .	126
4.3	Summarised optimisation results in Munich west . . . . .	127
4.4	Time steps in case study Munich centre . . . . .	131
4.5	Summary of optimisation results in case study Munich centre . . . . .	131
4.6	Parameters for demand estimation in Haag i. OB . . . . .	139
4.7	Matching of OSM building types to DOE load types . . . . .	145
4.8	Maximum allowed fluctuating renewable capacities for Haag i. OB . . . . .	147
4.9	<b>urbs</b> result energy sums for Haag i. OB, scenarios BASE and BATTERY CHEAP . . . . .	149
A.1	Attribute-value system example . . . . .	156

# List of Algorithms

2.1	Original K-Means algorithm . . . . .	45
2.2	Modified K-Means algorithm for load curve discretisation . . . . .	46
2.3	Area aggregation . . . . .	48
2.4	Skeletonisation of road networks . . . . .	49

# List of Listings

2.1	GAMS example model <a href="#">fuelstation.gms</a> . . . . .	53
2.2	MathProg example model (full model in listing B.1) . . . . .	54
2.3	YALMIP example model . . . . .	55
2.4	A street segment encoded using OpenStreetMap XML . . . . .	62
A.1	Common structure of model files <a href="#">dhmin.py</a> , <a href="#">urbs.py</a> , <a href="#">rivus.py</a> . . . . .	158
A.2	Exemplary run script <a href="#">runexample.py</a> . . . . .	159
B.1	MathProg example model for storage sizing <a href="#">soforsg.mod</a> . . . . .	162



# Chapter 1

## Introduction

*You can never plan the future by the past.*

---

Edmund Burke (1729–1797)

Cities first emerged around the year 4000 BCE in Mesopotamia. The process of forming cities, called urbanisation, was enabled by the development of agriculture during the Neolithic revolution. It is still not a settled question whether agriculture has developed under the pressure of growth of an already settled population, or whether the domestication of plants gradually evolved from intensified use of wild crops [105] [110, ch. 2]. Either way, once food productivity had increased, social specialisation and division of labour evolved as a natural consequence. But the aggregation on population in dense space also created negative side effects like epidemics, always limiting the maximum size a city could reach.

Among other factors, developments in urban growth were historically enabled through technological advances. Sanitation and hygiene dramatically increased life expectancy [126, pp. 9–11]. The industrial revolution allowed automating manufacturing and agriculture. Increased food production requiring less workers enabled us to feed a growing world population, while freeing up labour from field work for other occupations. Division of labour lead to a specialisation of professions in craft, industry, services, research and arts.

The growth of cities has always depended on the capability to support its population's basic needs. The required support is met by so-called infrastructure, a term whose meaning is discussed in section 2.1.1. In brief, a city, when seen as an entity, requires its demands for nutrition, goods, services and energy to be satisfied either from within the system or its so-called support region, a conceptual area around the city. On the output side, it requires waste disposal and transport services to export its production output [59]. In this thesis, the focus is put on the energy sector for stationary uses.

Energy supply in cities comprises of meeting the demand for electricity and heat of households, commerce and industry. Until the late 1990s, electricity demand was primarily satisfied by central power plants, while heat demand was met by local fuel combustion. The energy transition that is currently taking place in Europe and around the world is characterised by a shift towards renewable energy resources. Higher efficiency is beginning to link electricity and heat sectors and questions the established structures of both systems. In parallel, electrification is still an ongoing process in developing countries.

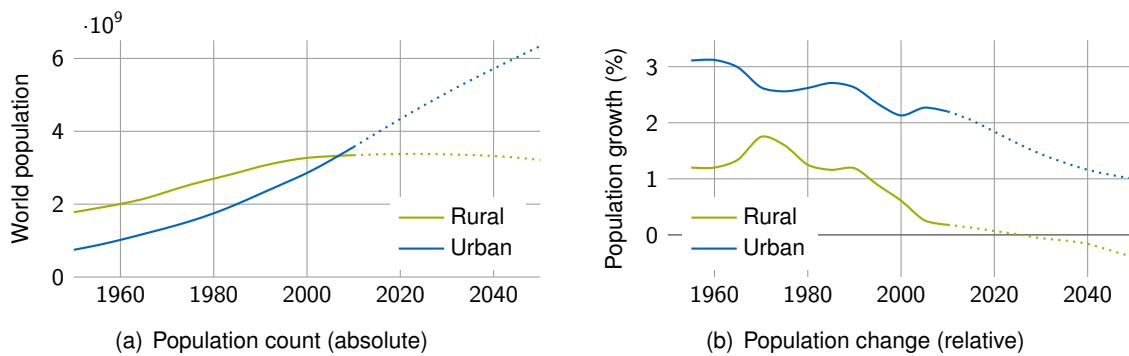


Figure 1.1: World population in rural and urban areas from 1950 to 2050 [118]

Although a completely sustainable way of living cannot be accomplished solely with technical measures, changes in the technical energy infrastructure might play a key role in achieving this goal [117, pp. 45–51]. Only a concerted approach in all domains [73, p. 114] has a chance of shifting the energy supply towards sustainability [73, pp. 103–106] in the long run.

## 1.1 Motivation

The world's urban population overtook rural population for the first time during the year 2007, as shown in figure 1.1. Efforts targeted towards urban planning receive higher leverage. The press release from the latest *Urbanization Prospects* [118] forecast for urban development states that

[...] a successful urban planning agenda will require that attention be given to urban settlements of all sizes. If well managed, cities offer important opportunities for economic development and for expanding access to basic services, including health care and education, for large numbers of people. Providing public transportation, as well as housing, electricity, water and sanitation for a densely settled urban population is typically cheaper and less environmentally damaging than providing a similar level of services to a dispersed rural population. UNO, 2014 [119]

Thus, finding sustainable solutions to satisfy the basic needs of a growing urban population allows to yield yet untapped potential synergies, as for example shown in a recent case study for the Shanghai region [88]. Especially improvements in urban energy infrastructure might help to improve the efficiency and sustainability of the whole system [5].

This thesis hopes to make a contribution towards exploring innovative ways for planning urban energy infrastructure. It aims to take into account synergies by coupling the energy sectors heating, cooling, and electricity.

## 1.2 State of the art

A wide range of scientific disciplines describe the development of cities. The broadest view on the subject is arguably offered by the field of **ecology**, which emphasises modelling a city as a

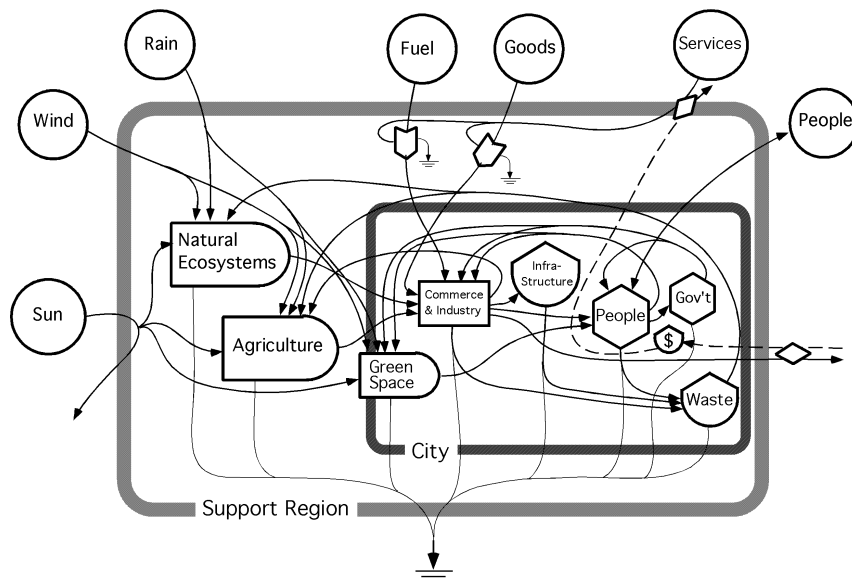


Figure 1.2: Systems diagram of a city and its support region as a flow network. Image: M. T. Brown [13]

living organism, embedded in an environment which must support its metabolism, as visualised in figure 1.2. This view is accompanied by a range of modelling approaches that draw from the theory of natural ecosystems. One widely cited model is Puget Sound Regional Integrated Synthesis Model (PRISM) [4]. It bundles an urban economy and a landscape ecology model. It covers land use, water use, emission output, and infrastructure. Land use in this model is based on demographic and economic submodules, which lead to resource consumption and land development. The aim of such a framework is to characterise or simulate the observed behaviour of a city at a large scale.

Naturally, the field **urban planning** has a more prescriptive approach to the topic of urban development. A recent example of a simulation approach comes from Zhang et al. in 2011 [130], simulating future growth in Shanghai with cellular automata and Markov chains, based on raster satellite land use data for the status quo. Development scenarios were derived from calibrating Markov chains from expert interviews. Similar to ecological approaches, the method is more simulative than optimising. Other approaches also include the perspective of resource management to model material flows into and out of a city [3].

**Engineering** methodologies have been applied to a wide range of planning issues. A notable historic example is the application of systems analysis by Forrester [27] to urban planning, bringing quantitative methods of control theory to the planning table. His methods were to be employed later by the study *Limits to Growth* [79] for the Club of Rome in 1972 using the model World3<sup>1</sup> [78].

A review paper by Mancarella [75] surveyed numerous publications on multi-energy system (MES), in which “electricity, heat, cooling, fuels, transport, and so on optimally interact with each other at various levels (for instance, within a district, or a city, or at a country level).” He

<sup>1</sup>The model nowadays can now be easily run in a browser, as in this faithful re-implementation: <http://bit-player.org/wp-content/extras/limits/>

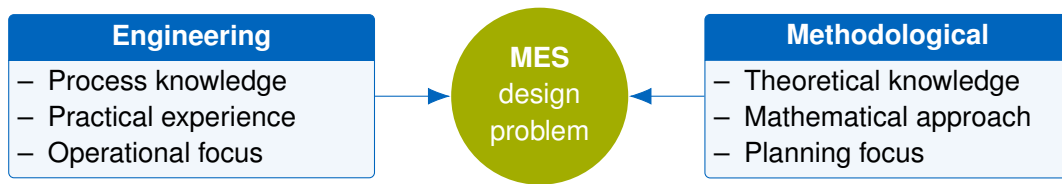


Figure 1.3: Opposing directions from which to tackle the multi-energy system design problem

lists four perspectives that motivate this interaction: first, the *spatial* perspective stresses the hierarchy of scales on which systems interact. These scales typically are: building, district, city, region, country, continent, and global. Secondly, the *multi-service* perspective focuses on system outputs that contribute to multiple contexts of energy use. The classical example here is cogeneration of heat and electricity in combined heat and power (CHP) plants. Thirdly, the *multi-fuel* perspective stresses system inputs that are combined by systems. The classical example here are waste incineration plants. Likewise, biomass and fluctuating renewable energy sources also emphasise a focus on multiple fuel use. Fourthly, the *network* perspective focuses on the interconnections between components of a system, for example in order to size a local district heating network [34]. The review concludes:

From the literature analysis on deterministic models it emerges how thermoecconomics is most suitable for MES plant design and could be used for operational optimisation as well, although energy-based economic models and classical discounted cash flow techniques (NPV assessment, in particular) are much more used for investment appraisal due to their simplicity.

Mancarella, 2014 [75, p.14]

With *thermoecconomics*, the author summarises methods that take into account the level of quality of different forms of energy (e.g. heat: low, electricity: high) in their economic assessment of energy (or rather exergy) streams. Additionally, he identifies (the frequent lack of) internalising externalities as the biggest hurdle towards a multi-criteria focus of research. Stochastic methods, while widely applied, are not yet understood well enough to have become a standard practice with commonly accepted rules.

Figure 1.3 shows two diametrically different directions from which the problem to design future MES can be approached. Each direction has its own strengths and weaknesses, as eloquently described by von Meier in the book *Electric Power Systems* [123, chapter 9.3]. The engineering approach lends itself to a mastership and refinement of existing technologies and system structure. It relies on operational knowledge of the existing physical system; this method empirically leads to working solutions that tend to perform very well within the (only implicitly) considered design space. On the other hand, the methodological approach tends to ignore technical details for the benefit of considering a broader design space. This can lead to a proposal of less orthodox solutions, while the practical implementation of such a solution is left open. This approach is therefore less geared towards practicality, but more to feasibility studies in general.

The optimisation methods employed in this thesis clearly favour the methodological approach by neglecting the question of how to practically operate an interconnected MES. Instead, the methods focus on identifying potentially attractive designs in a broad design space by

considering the main trade-off relationships quantitatively. Some practical considerations for electricity network planning are given by Kaufmann [57]. Concerning the methodological approach, section 2.3 provides an overview of existing models that follow the common reference energy system (RES) representation.

## 1.3 Contribution

This thesis hopes to make the following three contributions to the field of energy system modelling: the first one is a push towards improved **maintainability** of models by using version control, well documented source code and a small, well-documented code base. Secondly, it hopes to improve the standards in **reproducibility** of its results by publishing all source code as open source and making some of the presented case study data sets available together with the model. Thirdly, this thesis presents an example of **co-optimisation** using two independent models by feeding one model's results as input into the other model.

### Maintainability

Two of the models described in this thesis, **URBS** [101] and **DHMIN**, had been in use for several years to perform multiple studies conducted for the German power system [43], the European power system [102, 103, 104], or the district heating networks of Frankfurt [74] and Salzburg [12, 99]. Unfortunately, up to now, no *complete* mathematical description for **DHMIN** or **URBS** had been published to allow for independent review of these models.

This thesis aims to remedy the situation by fully describing those two *pre-existing* models. Additionally, three new models that were derived from those two models are presented, described and published with their implementation. This way, the method is independently documented from the software implementation, making it easier to re-use conceptual parts of the models without having to resort to the source code. Both high level overviews and detailed mathematical descriptions for each model are provided in chapter 3.

### Reproducibility

Many energy system studies published today either omit publishing a full model description or cannot disclose all used input data. The motivation for this can be two-fold: for once, a lot of data is corporate property and is only disclosed to researchers under the condition of confidentiality. In that case, being allowed to publish results of research conducted with closed data serves at least the purpose of making available conclusions derived from otherwise completely inaccessible datasets. This thesis is no exception to this rule: two of the four case studies can only be paraphrased qualitatively or be discussed with modified input parameters. In the other case, data could be published, but is not. Lack of motivation, fear of embarrassment, or sense of competition are possible reasons for this lack of openness. Here, a chance for allowing others to build on one's work is lost.

This thesis encourages reuse of the effort made by publishing all models, input data for two of the four presented case studies, as well as data processing scripts as open source. That way, not only the mathematical descriptions can be scrutinised, but also their implementation

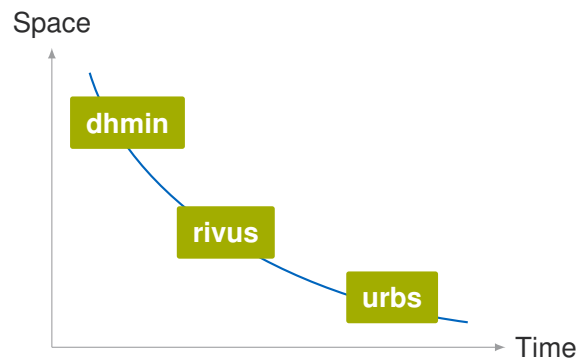


Figure 1.4: Thesis models on line of constant complexity (↘) in model design space of spatial and temporal resolution (arbitrary scales)

in code. By relying on numerous open source software packages, the author has tried to make every step in the whole modelling tool chain as concise, accessible, and readable as possible.

## Co-optimisation

This study applies linear programming (LP) and mixed integer linear programming (MILP) models to urban energy infrastructure. Sector-coupling, i.e. planning infrastructure for both electricity and heat demand, is not a novelty, but this thesis hopes to increase clarity and explicitness for planning of distribution networks at an urban scale. Two classes of models are presented. The first, represented by **URBS** and **urbs**, are generic capacity expansion and unit commitment models. They are the basis of more customised models that are employed by colleagues, e.g. for unit commitment in power system studies. The second class are network models. Their focus is pre-study quality sizing of network capacities, as shown in models **DHMIN** and **dhmin**. Model **rivus** adds sizing of energy conversion processes and thus exhibits some aspects of both model types. This trade-off is summarised graphically in figure 1.4.

In the presented case study in Haag in Oberbayern in section 4.4, the effect of local, distributed battery storage technologies is investigated under different scenarios as part of project EEBatt [116], division *research on future visions*.

The presented models are used to explore the parameter space of stylised economic and technical parameters to find scenarios (i.e. parameter value combinations) in which substitution effects among the different network and energy conversion topologies occur. They then, using a co-optimisation approach, approximately quantify the effect of storage on optimal planning decision using the model **urbs**.

Coined in Mancarella's [75] four perspectives of MES, as discussed in the previous section, **rivus** contributes to the *network* perspective on MES, while **urbs** helps to transform the results into the time domain and interpret them in the *multi-service* perspective, taking local storage of different energy carriers into account.

## 1.4 Legend

The following chapter briefly outlines all used *methods* first. Next, the five optimisation *models* are presented conceptually and mathematically. Then, four different *case studies* are discussed, followed by a final *conclusion* and outlook.

Paragraphs with blue background are **remarks** that might be of interest for model users, but may be skipped without loss of continuity.

Paragraphs with ivory background are **examples** that try to clarify a preceding definition by giving a concrete example.

A full listing of symbols is given in appendix D and a list of all abbreviations in appendix E.





## Chapter 2

# Method

*The urban researcher has limited impact since, as related to the urban decision making process, he comes late, leaves early, and does not get involved in implementation.*

---

Frederick William Heiss (1932–2014)

Having introduced the need for an optimal planning of urban energy infrastructure in a general manner, this chapter now introduces all methods that are employed in the models and case studies presented thereafter.

This chapter begins with a brief literature search on definitions of the term infrastructure. As it turns out, the word is a rather recent invention having its roots – like operations research – in the military sector. After this excursion, urban infrastructure networks are shortly presented.

The second part then contains a primer on mathematical optimisation. It briefly describes the different problem classes from general optimisation problems over convex to linear programming and mixed integer linear programming. Finally, the mathematical building blocks for describing the model elements energy carriers, demand, supply, conversion, transmission, and storage are presented individually.

A literature review on existing energy system models helps to locate the presented work in the landscape of energy system modelling. Both widely used and newly developed models are presented.

The section on algorithms outlines the optimisation algorithms of employed solvers, while glossing over all details that *actually* make modern solvers fast. Of probably higher interest are the detailed descriptions of custom algorithms for clustering load duration curves, and all the geographic operations needed for preparing spatial input data.

The section about software presents the final tool chain, and the alternatives that were considered during its evolution into the current form. Pointers for possible future improvements are also given.

The chapter is concluded by a description of input data. This includes technical (capacity, efficiency) and economic parameters (investment costs, annuity factor). Geographic data and their particularities (e.g. coordinate projections) are described here as well.

## 2.1 Urban infrastructure

### 2.1.1 On the term infrastructure

Starting with a library index search, infrastructure is mentioned in the fields of economics, geosciences and spatial planning. This indicates the cross-sectional significance of the topic. A search on the use of the term in books during the last century in the Google Ngrams database [36] yields the graph shown in figure 2.1(a). Another search for keywords specific to energy infrastructure yields the delightful figure 2.1(b). It can be seen that *infrastructure* is quite a young term, coming into common use only during the 1960's. The second graph documents the occurrence of the three keywords *power grid*, *district heating*, and *gas pipeline* in their literature database. The curves show a sharp increase in frequency of all terms beginning in the 1970's. However, the decline of district heating since the 1980's should not be confused with a drop in interest [87] but could also indicate maturity of the technology.

The latest review publication on the term infrastructure is a paper by Buhr [16] from 2003, which summarises definitions used by economists during the 20th century. He presents a classification of infrastructure into three types: institutional, personal and material. *Institutional infrastructure* comprises rules and procedures, mainly provided and enforced by the state. In that sense, government and legislation provide a legal infrastructure to the public. *Personal infrastructure* means human capital, i.e. people and their education and skills. *Material infrastructure* represents capital goods that provide services to the general public. These include transportation, education, health facilities, provision of energy and water, facilities for sewage, garbage disposal and air purification; building and housing stock, facilities for administrative purposes and for the conservation of natural resources. The task of infrastructure as seen by Buhr lies in "rendering possible the opening and development of the economic agent's activities." Economic agents are households, enterprises and the market. **Infrastructure thus are all those assets which enable economic activities of the participants in an economy.**

In his thesis from 1997, Wolter [129] provides an extensive comparison of colloquial and technical use of the term *infrastructure* through time. The term is considered to have its roots in military language, where it stands for the fundament of an organisation. During the late 1950's, the term became generally adopted in both scientific and political discussion. From there, its meaning extended fast to encompass **transportation systems, energy supply, education**

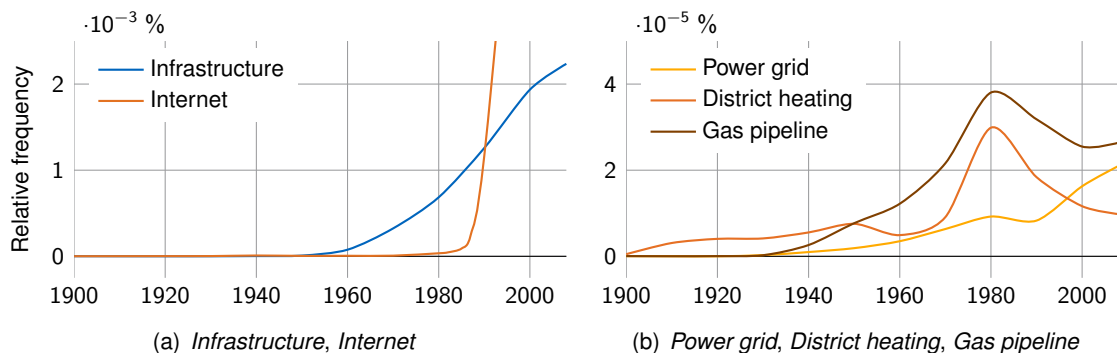


Figure 2.1: Frequency of keywords in published books between 1900 and 2008 [36]

Table 2.1: Examples of infrastructure in different domains. Abridged from [129]

Domain	Examples
Administration	Laws and regulations enabling use of the material infrastructure
Communication	Postal service, telephone networks, internet, undersea cables, satellites
Education	Libraries, schools, universities, research institutes
Energy	Production, generation, transmission and storage
Financial	Money system, financial institutions, stock market, contracts
Health	Doctors, hospitals, pharmacies, emergency rescue services
Monitoring	Meteorological monitoring networks; earth observation satellites, GPS
Security	Police, armed forces, fire service, disaster preparation
Transportation	Public transportation, roads, rails, waterways, tunnels, airports
Waste	Garbage collection, recycling facilities, waste disposal
Water	Water supply, sewage, irrigation, flood control, snow removal

**sector, and communication networks.** To categorise the different uses, he presents three criteria: Infrastructure can be distinguished by its provider (private or public), by its properties (technical, economic) or by the domain. Through cluster analysis, he derives four basic types of infrastructure: *Public base services*, *capital intensive core infrastructure*, *supplemental infrastructure*, and *market-provided services*. Having derived labels for infrastructure types, he presents different approaches to directly define *infrastructure*. Table 2.1 provides examples of infrastructure by domain. However, this approach of definition by exhaustive enumeration is deemed to fail, as it unavoidably becomes obsolete. Wolter concludes that descriptive properties of material infrastructure are hard to find due to the heterogeneous nature of the domains it touches. Especially *institutional* and *personal infrastructure* are too broad to have any chance of finding selective properties at all. The alternative, i.e. limiting the term infrastructure only to material assets, falls short as well. Sometimes the definition **publicly provided goods and services** is used, but it is both too restrictive – as it excludes privately provided services – and too broad at the same time – as not all publicly provided goods are infrastructure. Wolter finally focuses on *material infrastructure*, but leaves the definition open to include services in case they significantly contribute to public welfare.

For the purpose of this thesis, infrastructure is used as a placeholder term for **all technical equipment for conversion, storage, and distribution of energy** within the boundaries of the system under consideration. These range from below of the level of inter-regional transmission networks (for electricity and natural gas) to boundaries of private property.

### 2.1.2 Overview on urban infrastructure networks

The most visible infrastructure network are undoubtedly streets and rails. The hidden champions, however, are underground networks. They provide the most basic services, i.e. water, heat and energy. This section gives a very short primer on the four network types that are investigated in the case studies.

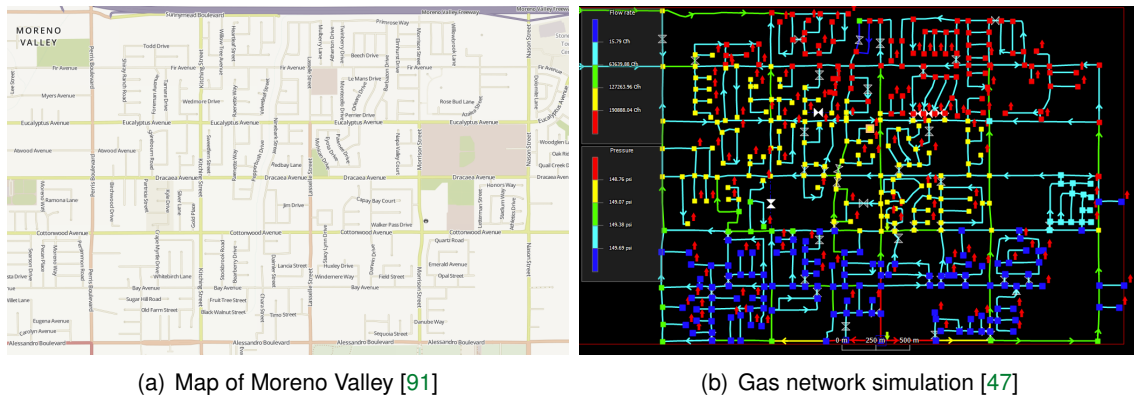


Figure 2.2: Gas network simulation software used on Moreno Valley, CA.

## Gas

Use of natural gas took off with the beginning of the 19th century. Leuschner [67] gives a short history of the German, Utoft and Thomsen [121] of the global history of its usage. First used for public lighting of streets, hundreds of kilometres of gas networks were installed in major cities. At that time, synthesised gas from coal gasification (also called *coal gas*) was primarily used, rather than natural gas. Gasometer buildings, needed for buffering and maintaining a stable pressure at the time, are visible testimonies from that era. By 1824, cooking with gas was first offered. In 1850, the first water heaters for domestic use were presented. However, their market success did not begin before the 20th century due to lack of convenience and substantial safety issues (i.e. explosions). Space heating was the last service to be offered to residential customers. Its use spread during the 20th century, when new materials and advances in radiator design were made.

Today, gas networks are operated typically by utility companies, which oversee construction and operation. Figure 2.2 shows an exemplary city map next to a digital model of its natural gas distribution network. Pressure levels throughout the network can be calculated for the given network topology and thus the consequences of changes in operation or construction can be predicted.

## Electricity

Electricity started off with the introduction of alternating current<sup>1</sup> power lines and affordable generators. A fairly recent meta-review on the (lack of) literature about the history of global electrification was presented in 2002 by Morton [84].

Technically, the major properties of technical infrastructure are *voltage* levels. Higher voltages allow to transmit a given power with lower *electric current*, which reduces *thermal losses*. The capability of a technical component to either emit, transmit or consume a certain *electric power* is summarised in its power rating or – as it referred to in this thesis – its **capacity**. The concept of reactive power, a power flow that oscillates through the network with the grid

<sup>1</sup> *The Empire of Lights* by Jill Jones [56] gives an enlightening account on the format war between alternating current and direct current, also called the “War of Currents”.

frequency, but which principally cannot be extracted for mechanical work, is important in operation planning. For simplicity, only active power flow is regarded in the models presented in this thesis. However, there is rich literature, especially on unit commitment (UC) and optimal power flow models. [19]

### District heating

District heating is the concept of generating heat at a central location instead of the place of consumption. In district heating, thermal energy is transported from a central generation point to consumers. The transport medium is usually hot water, which is pumped through a network of insulated pipes to the place of consumption. Some applications, for example to supply high-rise buildings, also use steam as the energy carrier. At the place of consumption, the energy carrier is cooled in a heat exchanger by the consumer. The cooled water is lead back through a separate return network to close the cycle.

The pipes are made of metal or polyurethane (PU) pipes, insulated with polystyrene foam (“styrofoam”) or glass wool. Thermal expansion and stress are main causes for ageing. Connecting to pipes can be done during operation thanks to drilling techniques that immediately seal the hole tightly. Topology-wise, forming rings among heat sources (i.e. heating stations, usually cogeneration power plants) helps increasing network availability and facilitate operation. Pressure differences between network nodes drive mass flow and thus thermal power transport. The thermal power  $\dot{Q}$  (J/s = W) of a pipe with mass flow  $\dot{m}$  (kg/s), specific heat capacity  $c$  (J/(kg K)) of the medium and the temperature difference  $\Delta T$  (K) between flow and return line is given by the expression

$$\dot{Q} = c \dot{m} \Delta T. \quad (2.1)$$

The most common working fluid, water, has a specific heat capacity  $c$  of roughly 4.2 kJ/(kg K). The required volume flow  $\dot{V}$  then can be derived using the density of water  $\rho$  of about 1000 kg/m<sup>3</sup> and the required mass flow  $\dot{m}$  using the relationship

$$\dot{V} = \frac{\dot{m}}{\rho}. \quad (2.2)$$

The volume flow  $\dot{V}$  is related to the flow velocity  $v$  (m/s) and the pipe diameter  $D$  by the volume of the conceptual cylinder volume that is formed by the pipe’s area cross-section and the distance travelled by the water

$$\dot{V} = \left(\frac{D}{2}\right)^2 \pi v. \quad (2.3)$$

Typical values for  $v$  are up to 2 m/s to 3 m/s, depending on operation. Temperature difference  $\Delta T$  also depends on operation but falls in the range 5 K to 20 K. Using the previous equations, the following expression relates the pipe diameter  $D$  with a desired theoretical thermal power flow  $\dot{Q}$ :

$$D = 2\sqrt{\frac{\dot{Q}}{\rho\pi c v \Delta T}} \quad (2.4)$$

The resulting theoretical diameter for these value ranges and the medium water to transmit a thermal power flow  $\dot{Q}$  of 10 MW is shown in table 2.2. It ranges from over one meter (for low flow speeds and temperature spread) down to 22 cm.

Table 2.2: Theoretical pipe diameter  $D$  (in meters) for the medium water transmitting a thermal power  $\dot{Q}$  of 10 MW for different flow speeds  $v$  and temperature differences  $\Delta T$  between flow line and return line, based on equation (2.4)

Flow speed $v$ (m/s)	Temperature difference $\Delta T$ (K)					
	5	10	15	20	25	30
0.5	1.10	0.78	0.64	0.55	0.49	0.45
1.0	0.78	0.55	0.45	0.39	0.35	0.32
1.5	0.64	0.45	0.37	0.32	0.28	0.26
2.0	0.55	0.39	0.32	0.28	0.25	0.22
2.5	0.49	0.35	0.28	0.25	0.22	0.20
3.0	0.45	0.32	0.26	0.22	0.20	0.18
3.5	0.42	0.29	0.24	0.21	0.19	0.17
4.0	0.39	0.28	0.22	0.19	0.17	0.16

### District cooling

District cooling is the principle of district heating, reversed. In district heating, a consumer retrieves thermal energy by *cooling* pre-heated water. In district cooling, the consumer *heats up* delivered cold water which is returned to the generation point, where it is re-cooled.

The lower temperature difference between cold water and soil of about 10 K to 20 K has two consequences. First, the pipes usually do not need as much insulation as district heating (DH) pipes of comparable capacity, which is an advantage. Secondly, the required diameter for an identical cooling power needs to be larger, as equation (2.1) suggests. Table 2.2 also applies, while the achievable temperature differences are smaller than with district heating.

Historically, the motivation for introducing district cooling (DC) is different from DH. While DH first aimed to increase urban air quality by omitting ubiquitous local combustion of fuels, DC aims to reduce the urban heat island effect that is severed by many local air conditioning units. Thus the urban micro climate is to be improved.

## 2.2 Mathematical optimisation

Mathematical optimisation is a special type of optimisation. Most generally, optimisation means *selecting a best element* from a set of alternatives. In case of mathematical optimisation, the selection criterion is an arbitrary function, called *objective function*. The set of alternatives can be a discrete or continuous, finite or infinite, one- or multi-dimensional set of choices, called *feasible region*.

Mathematical optimisation comprises of two steps: *mathematical modelling* for defining the optimisation problem and *mathematical programming* for finding a solution to a given optimisation problem.

The following sections first give an overview on the basics of mathematical modelling, which are used for formulating optimisation problems. In section 2.4.1, a sketch of algorithms that are employed to solve these problems for optimal or suboptimal solutions are presented.

### 2.2.1 A short primer on mathematical modelling

Mathematical modelling describes the act of translating a conceptual problem into mathematical objects, so that they can be tackled with standard algorithms.

As an example, consider a planner who has to dimension the capacity  $\kappa$  of a power plant to satisfy the electricity demand of a given region. His goal is to minimise the costs that have to be paid for that capacity. The costs are a function of the installed capacity  $c(\kappa)$ . If the electricity demand is given by a function  $d(t)$  over a planning period  $t \in T$ , the capacity of the power plant must be large enough to satisfy that demand for any moment. In other words, the planning decision can be expressed by the following mathematical statement:

$$\begin{aligned} \min_{\kappa \in \mathbb{R}} c(\kappa) \\ \text{s.t. } \kappa \geq \max_{t \in T} d(t) \end{aligned} \quad (2.5)$$

In order to reason about general classes of optimisation problems, mathematicians have defined so-called *standard forms*. The standard form of a general non-linear optimisation problem is:

$$\begin{aligned} \min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) \\ \text{s.t. } \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \\ \mathbf{h}(\mathbf{x}) = \mathbf{0} \end{aligned} \quad (2.6)$$

The objective function  $f$  can readily be identified with the cost function  $c$ , while the decision space is the range of possible capacities, i.e. real positive numbers (with some idealisation). There is only one inequality constraint for the variable  $\kappa$  and no equality constraints. Therefore,

(2.5) in standard form becomes:

$$\begin{aligned} \min_{\kappa \in \mathbb{R}_0^+} c(\kappa) \\ \text{s.t. } \max_{t \in T} d(t) - \kappa \leq 0. \end{aligned} \quad (2.7)$$

Here,  $\mathcal{X} \subseteq \mathbb{R}^n$  is the space of possible solutions.  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is the objective function to be minimised. However, this standard form allows to represent a wide range of optimisation problems because of the following transformations: Maximisation problems can be converted to minimisation by negating the objective function. Function  $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  represents  $m$  inequality constraints, while function  $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^p$  represents  $p$  equality constraints. The set of feasible solutions  $\mathcal{F}$  is the subset of the solution domain  $\mathcal{X}$  that satisfies all constraints.  $\mathcal{F}$  is also called *constraint set*:

$$\mathcal{F} = \{\mathbf{x} \in \mathcal{X} \mid \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \wedge \mathbf{h}(\mathbf{x}) = \mathbf{0}\} \quad (2.8)$$

Using this constraint set definition, equation (2.6) can be equivalently expressed without an explicit constraint:

$$\min_{\mathbf{x} \in \mathcal{F}} f(\mathbf{x}). \quad (2.9)$$

It must be noted that neither objective function nor the constraint need to be explicitly defined. In this case, if one or both are only defined *implicitly*, no general algorithm for solving the general optimisation problem (2.6) can be given. One must instead resort to heuristics. Selecting a good approach is mainly facilitated by a thorough understanding of the problem domain. This path is usually limited to a very small number of variables ( $\dim \mathcal{X}$  small) and constraints. This drawback is disqualifying for the investigations performed in this thesis, as they are concerned with the boundaries of planning decisions in a vast decision space. A comparison of the different kinds of optimisation problems with advantages and disadvantages from specific to generic is given in table 2.3.

Depending on the properties of either the objective function or the constraints, different types of optimisation problems can be distinguished:

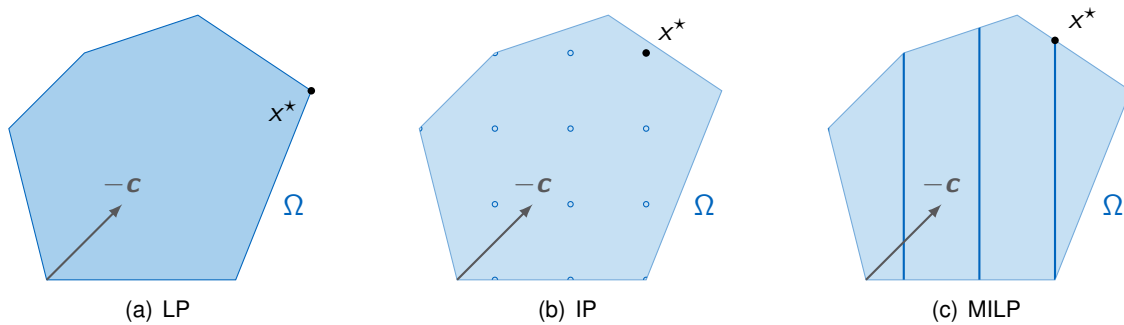
### Convex optimisation

Convex optimisation deals with the special case of a convex objective function and a convex constraint set. From an optimisation perspective, convex problems have a very desirable property: if  $f$  has a local optimum in  $\mathcal{F}$ , then it is also a global minimum. This property enables the use of iterative and search algorithms that yield local minima, without having to deal with the problem of proving that a local minimum is also globally optimal. For the standard form optimisation problem (2.6) to be convex, domain  $\mathcal{X}$  must be a convex set, the inequality constraints  $\mathbf{g}$  must be a vector of convex functions and any equality constraints  $\mathbf{h}$  must be affine. For a set, convexity means that the linear combination  $\lambda x_1 + (1 - \lambda)x_2$ ,  $\lambda \in [0, 1]$  of any two set members  $x_1, x_2 \in \mathcal{X}$  must also be contained in this set. For a function to be convex, any line segment connecting two points on its graph must lie on or above the graph of the function.



Table 2.3: Comparison of optimisation techniques. Based on [64, Table 3.2]

Technique	Advantages	Disadvantages
Linear programming	<ul style="list-style-type: none"> <li>– Scales well to big systems</li> <li>– Standard solver software available</li> <li>– Globally optimal solution attainable</li> </ul>	<ul style="list-style-type: none"> <li>– Very limited expressions available</li> <li>– Complex interactions difficult or impossible to represent</li> </ul>
Integer programming	<ul style="list-style-type: none"> <li>– Can scale well to big systems</li> <li>– Complex interactions representable</li> <li>– Standard solver software available</li> <li>– Quality of solutions can be assessed</li> </ul>	<ul style="list-style-type: none"> <li>– Bad worst-case complexity</li> <li>– Global optimum often not attainable</li> </ul>
Non-linear programming	<ul style="list-style-type: none"> <li>– Complex interactions fully representable</li> <li>– High freedom of expression</li> </ul>	<ul style="list-style-type: none"> <li>– Scales horribly to big systems</li> <li>– Quality of solutions impossible to assess</li> </ul>
Heuristics	<ul style="list-style-type: none"> <li>– Full freedom of expression</li> <li>– Can scale well to big systems</li> </ul>	<ul style="list-style-type: none"> <li>– Quality of solutions difficult to assess</li> <li>– Algorithms must be tailored by hand</li> </ul>

Figure 2.3: Linear programming, integer programming, and mixed integer linear programming in comparison.  $x^*$  denotes the optimum,  $\Omega$  the linear constraint set.

For the example problem to be convex, cost function  $c(\kappa)$  must be convex. The feasible set ( $\mathbb{R}_0^+$ ) is already convex. So, setting the cost function to a convex function like  $c(\kappa) = 3\kappa^2 + 2\kappa + 1$  would make (2.5) a convex optimisation problem.

## Linear optimisation

Linear optimisation or *linear programming (LP)* is a prominent subtype of convex optimisation. In addition to being convex, both  $f$ ,  $\mathbf{g}$  and  $\mathbf{h}$  must be linear. The *canonical form* of a linear program is given by

$$\begin{aligned}
 & \max_{\mathbf{x} \in \mathcal{X}} \mathbf{c}^T \mathbf{x} \\
 & \text{s.t. } \mathbf{Ax} \leq \mathbf{b} \\
 & \text{and } \mathbf{x} \geq 0.
 \end{aligned} \tag{2.10}$$

Although this form looks as if it limits all variables to being positive, any linear optimisation problem can be converted to this form using the following three techniques:

As mentioned before, a minimisation problem of  $f$  can be converted to a maximisation of the negation  $-f$ . In case of a linear function, minimising  $\mathbf{c}^T \mathbf{x}$  is equivalent to maximising the negative  $-\mathbf{c}^T \mathbf{x}$ . Any equality constraint  $\mathbf{h}^T \mathbf{x} = 0$  can be expressed by two inequality constraints  $\mathbf{h}^T \mathbf{x} \geq 0$  and  $\mathbf{h}^T \mathbf{x} \leq 0$ , which can be added as additional lines to  $\mathbf{A}$  and  $\mathbf{b}$ . Any unbounded variable  $x_j$  in the vector  $\mathbf{x}$  can be split into two components  $x_j^+$ ,  $x_j^-$  and the equality constraint  $x_j = x_j^+ - x_j^-$ , which can both be required to be non-negative by  $x_j^+ \geq 0$  and  $x_j^- \geq 0$ . By converting the equality constraint to two inequality constraints, an unbounded variable is thus converted into two positive variables and two inequalities. In practice, these conversions do not have to be done by hand, but are implemented as pre-processing steps either within the solver, or a modelling framework that forms an interface to the solver.

The planning problem above can be expressed as a linear optimisation problem, if the cost function can (with sufficient accuracy) be approximated to be linear  $c \cdot \kappa$  with constant cost parameter  $c$ . The inequality constraint  $\kappa \geq \max_t d(t)$  can be expressed as a finite set of elementary inequalities, if  $d(t)$  is sampled at sufficiently many time intervals  $d_t$ . Alternatively, one could already pre-process the demand curve and provide the maximum load directly as a parameter:  $d_{\max} = \max_t d(t)$ . With that, the planning problem could be expressed as

$$\begin{aligned} \max_{\kappa} \quad & -c\kappa \\ \text{s.t.} \quad & \kappa \geq 0 \\ & -\kappa \leq -d_{\max}, \end{aligned} \tag{2.11}$$

or without pre-processing the demand curve:

$$\begin{aligned} \max_{\kappa} \quad & -c \cdot \kappa \\ \text{s.t.} \quad & \kappa \geq 0 \\ & \forall t \in T: -\kappa \leq d_t. \end{aligned} \tag{2.12}$$

Why would one want to limit one's options in formulating an optimisation problem to only linear expressions? Simply: as the general form is so specific, several algorithms to solve even large LPs have been developed during the last decade. The historically most important one, the simplex method, is outlined in section 2.5.3.

### Integer programming

Integer programming (IP) is a special form of non-convex optimisation. Integers, often restricted to binary digits  $\{0, 1\}$ , reflect discrete decisions that cannot be done continuously. Scheduling, assignment and routing problems fall into this class. Formally, they can be represented by equation (2.10) with the added constraint that all components  $x_j$  of  $\mathbf{x}$  must be integers. Here, so called mixed integer linear programming (MILP) is of interest. In this problem class only some components  $x_j$  must be integers. This can be done by redefining the domain for  $\mathbf{x} \in \mathcal{S}$ . Without

loss of generality, the first  $r$  entries of  $\mathbf{x}$  have the integer constraint, yielding:  $\mathcal{S} = (\mathbb{Z}^r \times \mathbb{R}^{n-r})$ . With this definition for  $\mathcal{S}$ , the canonical representation for a MILP problem can be expressed by

$$\begin{aligned} \max_{\mathbf{x} \in \mathcal{S}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{Ax} \leq \mathbf{b} \\ & \text{and } \mathbf{x} \geq 0. \end{aligned} \tag{2.13}$$

It is beyond the scope of this thesis to discuss the algorithms for this problem class in detail, as extensive literature exists on the topic. A short discussion of how to approach LP and MILP problems is given nevertheless in section 2.5.3.

If the cost function cannot be approximated by the linear function  $c \cdot \kappa$ , but includes a fixed cost term for any non-zero value of  $\kappa$ , the following approach – called *fixed cost formulation* in operations research literature – can be used. First, a second variable  $\xi \in \{0, 1\}$  is introduced. It represents the binary decision, whether or not  $\kappa$  is greater than zero.

The fixed costs  $c_0$  and the capacity dependent costs  $c_1$  parameter then form the cost function  $c^{\text{fix}}\xi + c^{\text{var}}\kappa$ . The logical connection between  $\xi$  and  $\kappa$  can be expressed by the constraint  $\kappa \leq M\xi$  with a fixed parameter  $M$  that must be chosen bigger than any foreseeable value for the demand. Otherwise, the whole problem would become infeasible. Converted to canonical form (2.13), the problem can thus be stated as

$$\begin{aligned} \max_{\kappa, \xi} \quad & -c_0\xi - c_1\kappa \\ \text{s.t.} \quad & \kappa \geq 0, \xi \in \{0, 1\} \\ & \kappa - M\xi \leq 0 \\ & -\kappa \leq d_t \quad \forall t \in T. \end{aligned} \tag{2.14}$$

For this simple example,  $\xi$  must of course always have value 1 if the demand contains any non-zero value. Real-world applications of this technique must include at least two different technologies and variable costs to create any interesting results. Then, the optimal capacities (and usage) of each technology can depend heavily on the characteristics of the load profile  $d_t$ .

### 2.2.2 Mathematical formulations of energy model components

This section presents conceptual and mathematical building blocks of mathematical optimisation models, which become part of the linear problem (2.10) or linear mixed integer problem (2.13). The models presented in later chapters will be compositions of these blocks. For brevity of exposition, the symbols used are not separately defined in this section. Please refer to Appendix D for a full listing of all used symbols.

#### Space

The models presented in this thesis treat space discretely. Locations are represented as vertices, connected either through *undirected* edges or *directed* arcs to form a graph.



Figure 2.4: Model element: time

In model **urbs**, demand is located in vertices (called *sites*) that represent distinct locations. These locations could be of various scales: buildings, neighbourhoods, districts, cities, regions, countries. Only the labelling and scaling of input data determines the meaning of a vertex.

In models **dhmin** and **rivus**, demand is located in edges. These edges represent street segments in a street network. Only by power flow through these edges can the demand be satisfied. See section *transmission* below for more details.

### Time

Similarly to space, time is only represented discretely. Two different forms are used in this thesis, as depicted in figure 2.4. The first form in figure 2.4 introduces a set  $T$  with equally spaced, discrete instants  $t_0, t_1 \dots t_N$ . For easier definition of storage technology constraints, the subset  $T_m$  ( $m$  for *modelled* time steps) with the elements  $t_1 \dots t_N$  is defined as well. The initialisation time step  $t_0$  and the final time step  $t_N$  are also used in the definition of initial and terminal storage levels. If not noted separately, the time step duration (parameter  $\Delta t$ ) is one hour, and the number of time steps  $N$  is equal to 8760 (one year). The reason for that choice is quite pragmatic: first, it is precise enough to represent the major issues of scheduling, while still allowing to solve a whole year in a single optimisation problem, in order to capture the variance of seasonal effects. When modelling power markets, a resolution of 15 minutes is better suited, as it is the time resolution for most energy markets and the current time basis for power plant schedules.

The second form in figure 2.4(b) also introduces a set  $T$  of time steps. However, the elements  $t_0, t_1 \dots t_N$  no longer represent a sequence of consecutive instants. Instead, they represent an unordered list of durations (in models quantified as a time step weight  $w_t$ ). This formalism is used by models **dhmin** and **rivus** to represent an appropriate amount of *typical* operation conditions. This way, both short moments of peak load and long durations of typical load can be represented with minimum model size.

### Commodity

Commodities are goods, materials, and forms of energy. They must be representable by extensive quantities, e.g. mass, energy content or volume. In this thesis, energy carriers are represented by their energy content in kilowatt-hours (1 kWh = 3.6 MJ). Greenhouse gas emissions, on the other hand, are represented by their CO<sub>2</sub> mass equivalent in kilogram. They are all elements  $c$  of a set of commodities  $C$ .

To distinguish between different kinds of commodities, several subsets of the commodity set are created. All energy carriers that can be stored in sufficient quantity to allow for practically unrestricted use form the set of *stock commodities*  $C_{st}$  (examples: coal, gas). Energy carriers whose availability fluctuates over time are bundled in the set *supply intermittent*  $C_{sup}$  (solar

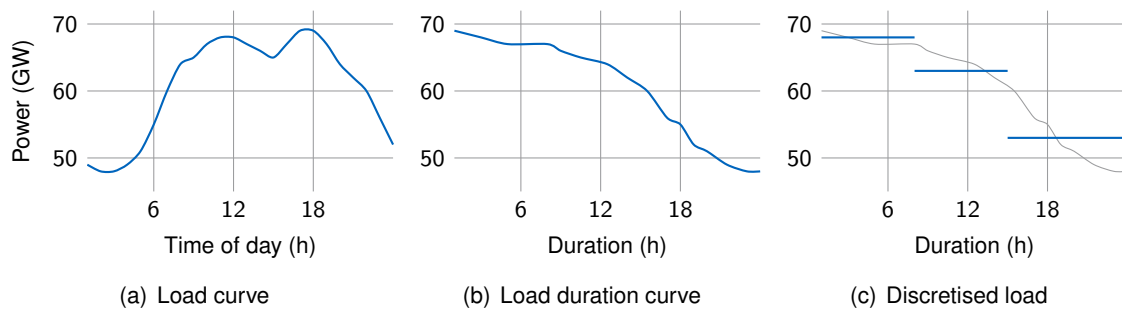


Figure 2.5: Derivation of a discretised electricity load curve for a day. (a) shows the original load curve for a single day, (b) the sorted (in descending order) identical curve. (c) shows a possible discretisation to three steps.

radiation, wind power). Then there are *demand commodities*  $C_{\text{dem}}$  (electricity, heat). Finally, *environmental commodities*  $C_{\text{env}}$  are by-products of energy conversion, whose generation can be limited (examples: greenhouse gases, exhaust gases, solid waste).

### Energy demand

**Load curves** Load curves are the direct form of representing the evolution of the demand for a commodity over time. An exemplary load curve for a country during a single day is depicted in figure 2.5(a). Load curves can be formalised as a time continuous function  $d : \mathbb{R} \rightarrow \mathbb{R}, t \rightarrow d_c(t)$ , with  $d_c(t)$  representing the required power flow for commodity  $c$  at instant  $t$ . For the following mathematical models, a time-discrete representation is more adequate: instead of instantaneous values, average power demands for a set of commodities  $C$  are sampled at certain (usually, but not necessarily equidistantly spaced) *time steps*  $T$  to form a parameter  $d_{ct}, c \in C, t \in T$ . The time resolution used in this thesis is one hour, but the presented models include a time step duration parameter  $\Delta t$  (in hours). This formalism is employed by models **URBS** and **urbs**.

**Load duration curves** By sorting the values of a given load curve in descending order, the load duration curve is derived. An example of a country's load duration curve is shown in figure 2.5(b). While the information on *when* a certain demand occurs has vanished, the information on *how long* a certain demand level occurs can be read directly from the curve. This form is not used by any model in this thesis, but it is an intermediate step to derive the next representation.

**Discretised load duration curves** By sampling the load duration curve at certain levels, the time resolution of the original curve can be greatly reduced with only little loss of accuracy – depending on the original form, of course. The exemplary result is shown in figure 2.5(c) Each sample is now represented by its value (scale factor, parameter  $s_t$ ) and duration (weight, parameter  $w_t$ ). The number  $|T|$  of time steps can be chosen freely. This representation is used by model **dhmin**.

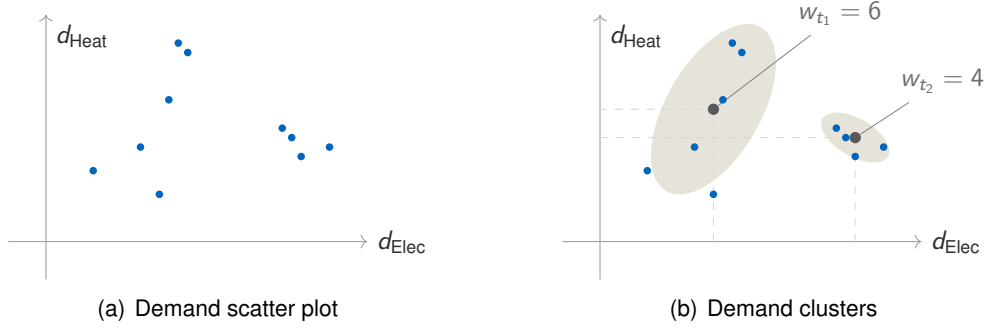


Figure 2.6: Discretisation of multiple load curves

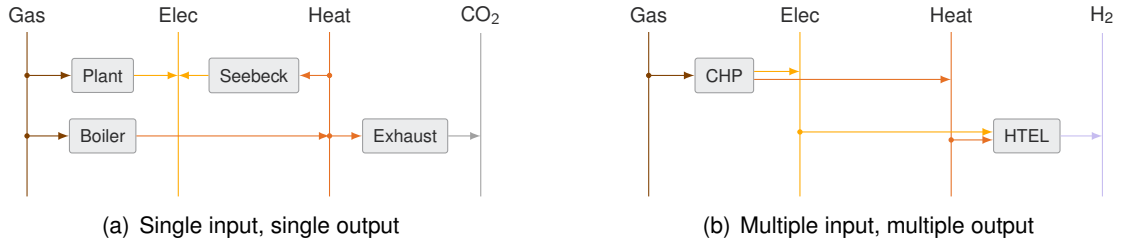


Figure 2.7: Model element: conversion processes

While this representation is quite easy to derive for a *single* demand commodity, it is more challenging to derive the curve for multiple demand commodities. Intuitively, this problem generalises to an  $n$ -dimensional clustering problem, where  $n$  is the number of demand commodities. For  $n = 2$ , a scatter plot for simultaneously sampled demands for electricity and heat is shown in figure 2.6(a). To derive the discretised load curve for such a scatter plot, this thesis draws from a bachelor thesis by Harmat [41]. It employs a customised K-means clustering algorithm that derives *typical load situations* and how often they occur (parameter weight  $w_t$ ) during the course of a year. It is customised, because the time steps with peak demand for each commodity are preserved in the resulting discretised clusters. The levels for a discretised load (or the position of the cluster centre in figure 2.6(b)) is stored in a scaling factor parameter  $s_{ct}$  for each commodity  $c$  and time step  $t$ . This representation finally is used in model **rivus** for scaling the peak demand. A longer description of this algorithm is presented in section 2.4.2.

### Conversion processes

**Single input single output (SISO)** An energy conversion process  $p$  can be conceptualised as a black box that converts an ingoing power flow  $\epsilon_{pt}^{\text{in}}$  to an outgoing power flow  $\epsilon_{pt}^{\text{out}}$  at a given time  $t$ . Multiple conversion processes form a graph of commodities (nodes) and processes (edges) that is shown in figure 2.7(a).

The conversion usually has losses, hence a conversion efficiency  $e$  must be introduced. It links the two flows:  $\epsilon_{pt}^{\text{out}} = \epsilon_{pt}^{\text{in}} e_p$ . The ingoing (or outgoing) power flow can be limited using a

process capacity variable  $\kappa_p$  that requires that  $\epsilon_{pt}^{\text{out}} \leq \kappa_p$ . In short:

$$\epsilon_{pt}^{\text{out}} = \epsilon_{pt}^{\text{in}} e_p \quad (2.15)$$

$$\epsilon_{pt}^{\text{out}} \leq \kappa_p. \quad (2.16)$$

More general formulations do not rely on a linear efficiency factor, but employ an efficiency function  $e_p : \epsilon_{pt}^{\text{out}}/\kappa_p \mapsto \mathbb{R}$  that depends on the process operating point, i.e. the fractional capacity utilisation  $\epsilon_{pt}^{\text{out}}/\kappa_p$ . In general, this formulation is non-linear. For unit-commitment problems ( $\kappa_p$  is fixed), linear formulations that employ piece-wise linear functions can be used. However, these techniques are beyond the scope of this thesis.

As an example, in the documentation of model TIMES (cf. section 2.3.1) [72, p. 227], the parameter `FLO_SUM` is the conversion efficiency in constraint `EQ_PTRANS`, which models the relationship between input and output commodity flows `VAR_FLO`.

**Multiple input multiple output (MIMO)** The previous formulation is limited to a single input and output commodity. But chemical processes in general have multiple reactants and products, as do energy conversion processes. For example, combined heat and power takes a fuel and produces heat and electricity. Also, emissions or other quantities could be seen as inputs and outputs of a process.

For their representation, an additional *throughput* variable  $\tau_{pt}$  must be introduced. Its role is to represent the state or operating point of the conversion process. It also becomes the quantity that is limited by the process capacity  $\kappa_p$ . In the SISO case, one could simply choose the input or output variable for that role. Instead of just one efficiency factor, a set of *input* and *output ratios*  $r_{pc}^{\text{in}}$  and  $r_{pc}^{\text{out}}$  is needed. With that, the formulation becomes

$$\epsilon_{pct}^{\text{in}} = \tau_{pt} r_{pc}^{\text{in}} \quad (2.17)$$

$$\epsilon_{pct}^{\text{out}} = \tau_{pt} r_{pc}^{\text{out}} \quad (2.18)$$

$$\tau_{pt} \leq \kappa_p. \quad (2.19)$$

Note also that the input and output power flows have an additional subscript for the commodity to allow for addressing the possibly multiple in- and outputs. Figure 2.7(b) shows two exemplary processes, one with two outputs – CHP – the other with two inputs – high-temperature electrolysis. Again, in model TIMES, variable `VAR_ACT` captures the activity of a process similar to the throughput variable. The capacity limit is implemented in equation `EQ_CAPACT` [72, p. 179], whose right-hand side is much more involved than the simple  $\kappa_p$  here due to the possibility of inter-temporal investment decisions (variable `VAR_NCAP`).

## Transmission

The simplest possible model for energy transmission is that of a single input, single output process of type  $f \in F$  with a constant transmission efficiency  $e_{af}$  between ingoing power flow  $\pi_{aft}^{\text{in}}$  and outgoing power flow  $\pi_{aft}^{\text{out}}$  through a directed arc  $a \in A$ . That arc is the directed connection from a vertex  $v_1$  to another vertex  $v_2$ . Its corresponding counterpart in the reverse direction is called  $a'$  here. This approach leads to the following set of equations to model the

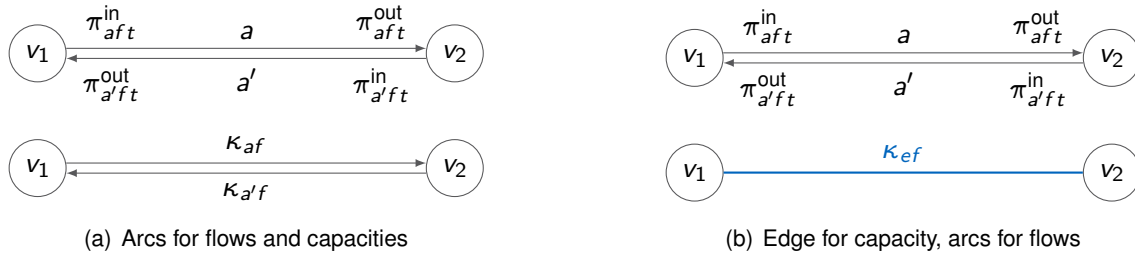


Figure 2.8: Model element: energy transmission

power transmission of a bidirectional transmission technology  $f$ :

$$\forall a \in A, f \in F, t \in T: \quad \pi_{aft}^{\text{out}} = \pi_{aft}^{\text{in}} e_{af} \quad (2.20)$$

$$\forall a \in A, f \in F, t \in T: \quad \pi_{aft}^{\text{in}} \leq \kappa_{af} \quad (2.21)$$

$$\forall a \in A, f \in F, t \in T: \quad \pi_{aft} \geq 0 \quad (2.22)$$

$$\forall a \in A, f \in F: \quad \kappa_{af} = \kappa_{a'f} \quad (2.23)$$

These equations are visualised in figure 2.8(a). Variable  $\kappa_{af}$  is the capacity of the transmission technology  $f \in F$ , which is responsible for transporting a certain, fixed commodity. Note that multiple different technologies for the same commodity with different technological and economic properties can be defined this way. One consequence of defining the transmission capacity  $\kappa_{af}$  over the set of arcs, is that two different capacities for each direction could be defined. Model **urbs** uses this approach of representing one undirected edge  $e$  as a pair of directed arcs  $a$  and  $a'$ . Enforcing symmetry of the transmission capacities in both directions with equation (2.23) allows only for bidirectional technologies, i.e. those which allow for transmission in either direction. In special applications (like transmission techniques that use height differences between vertices and thus have different transmission costs per direction), this symmetry constraint could be left out to allow for technologies that allow for separate independent investment in both directions.

Model **rivus** uses a different approach, which is depicted in figure 2.8(b). In this formulation, the transmission capacity is no longer defined over the set of directed arcs  $A$ , but over the set of undirected edges  $E$ . That way, only one capacity variable for each pair of connected vertices is needed:

$$\forall a \in A, t \in T: \quad \pi_{aft}^{\text{out}} = \pi_{aft}^{\text{in}} e_{af} \quad (2.24)$$

$$\forall a \in A, t \in T, a \in e: \quad \pi_{aft}^{\text{in}} \leq \kappa_{ef} \quad (2.25)$$

In this case, edges and arcs are two independent sets. Consequently, a correspondence mapping between the two must be given. Here, the (strictly incorrect, but intuitive) notation  $a \in e$  is used to indicate that the arc  $a$  must connect the same two vertices as the edge  $e$ , or more formally:

$$a \in e \Leftrightarrow [a = (v_1, v_2) \vee a = (v_2, v_1)] \wedge e = \{v_1, v_2\} \quad (2.26)$$

In model **TIMES**, the arc definition similar to equations (2.20) to (2.23) is employed. Equation **EQ\_IRE** [72, p. 214] relates exported and imported commodity flows (variable **VAR\_IRE**).



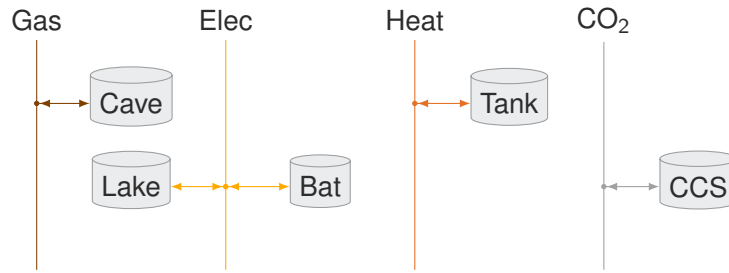


Figure 2.9: Model element: storage

### Storage

Storages can be most easily represented using a simple bathtub metaphor. Ingoing flow is stored up to a maximum capacity. The storage content  $\epsilon_{st}^{\text{con}}$  at one time step  $t$  is calculated by the difference of incoming and outgoing flows, added to the storage content in the previous time step  $\epsilon_{s(t-1)}^{\text{con}}$ . All quantities are scaled by optional efficiency factors  $\eta$ . The models in this thesis currently only implement incoming  $\eta^{\text{in}}$  and outgoing  $\eta^{\text{out}}$  efficiency. Technologies with significant self-discharge require the term  $\eta^{\text{con}}$ .

$$\forall s \in S, t \in T: \quad \epsilon_{st}^{\text{con}} = \epsilon_{s(t-1)}^{\text{con}} \cdot \eta_s^{\text{con}} + \epsilon_{st}^{\text{in}} \cdot \eta_s^{\text{in}} - \epsilon_{st}^{\text{out}} / \eta_s^{\text{out}} \quad (2.27)$$

$$\forall s \in S, t \in T: \quad \epsilon_{st}^{\text{con}} \leq \kappa_s^{\text{c}} \quad (2.28)$$

$$\forall s \in S, t \in T: \quad \epsilon_{st}^{\text{in}}, \epsilon_{st}^{\text{out}} \leq \kappa_s^{\text{p}} \quad (2.29)$$

In model TIMES, variables `VAR_SIN` and `VAR_SOUT` represent the ingoing and outgoing commodity flows into and from a storage technology. Equation `EQ_SRGTS` [72, p. 230] plays a similar role as the storage state equation (2.27).

### Costs

All of the previous components have the notion of a capacity  $\kappa$  and an activity that corresponds to a directed flow of energy  $\epsilon$  or  $\pi$ . By defining cost parameters that are associated with these variables, many different technologies can be parametrised. Refer to the model descriptions in chapter 3 for a discussion of the applicable economic parameters (denoted by letters  $c$  and  $k$ ). Before that, section 2.6.2 deals with methods to pre-process economic parameters like equivalent annual cost (EAC) that is used to convert multi-annual investment decisions to annual cost flows.

## 2.3 Literature review on energy system models

This chapter lists all prior art in modelling, which the models presented in chapter 3 are conceptually based on. A very detailed history of the evolution of energy system models is given in a recent review paper by Bhattacharyya [10]. A perspective on the current challenges for energy system modeling has been given by Pfenninger, Hawkes and Keirstead in 2014 [95]. They describe that the main challenges are four-fold:

- Resolving time and space
- Uncertainty and transparency
- Complexity and optimisation across scales
- Capturing the human dimension

Pfenninger et al., 2014 [95]

This section has three parts: first, “classical” reference energy system models, then network flow models and urban energy system models, and finally selected examples of closed and open source models.

### 2.3.1 Reference energy system

The reference energy system (RES) by Beller [7] is a network representation of “all the technical activities required to supply various forms of energy to end-use activities.” It consists of a graph theoretic abstraction over the energy flow that underlies the energy supply chain. This supply chain is then reconstructed by energy carriers (vertices) and conversion processes (edges) that form a graph flowing from resources (sources) to consumption (sinks). Almost all currently employed energy system models can conceptually be mapped onto this ontology of network elements.

Recent simplification happened from the domain of informatics, reducing the formal definition of RES, while allowing for even more flexible modelling [18]. The following models are based on the RES formalism and use it to represent the systems they are optimising. They are presented in chronological order of publication.

#### MARKAL

MARKet ALocation (MARKAL) [70] is a linear optimisation model for the energy system of one or multiple regions. Its focus lies on intertemporal investment decisions over long discounted time horizons. Its objective function is the net present value (NPV) of annualized total costs. It features elastic demands, i.e. instead of having to supply a given demand, (aggregated) consumers have a price elasticity that leads to a cost tradeoff between increasing supply and reducing demand.

The model includes energy flow from primary resources over intermediate steps to final energy and finally energy services that satisfy demands. Figure 2.10 shows the RES diagram of these energy flows. The solution of a MARKAL simulation is an “intertemporal partial equilibrium on energy markets” [70, p.12]. The standard variant is limited to partial equilibrium, which means that no cross-sectoral interactions among energy consuming sectors may exist. This limitation is lifted in model variant MARKAL-MICRO, for the price of being a non-linear

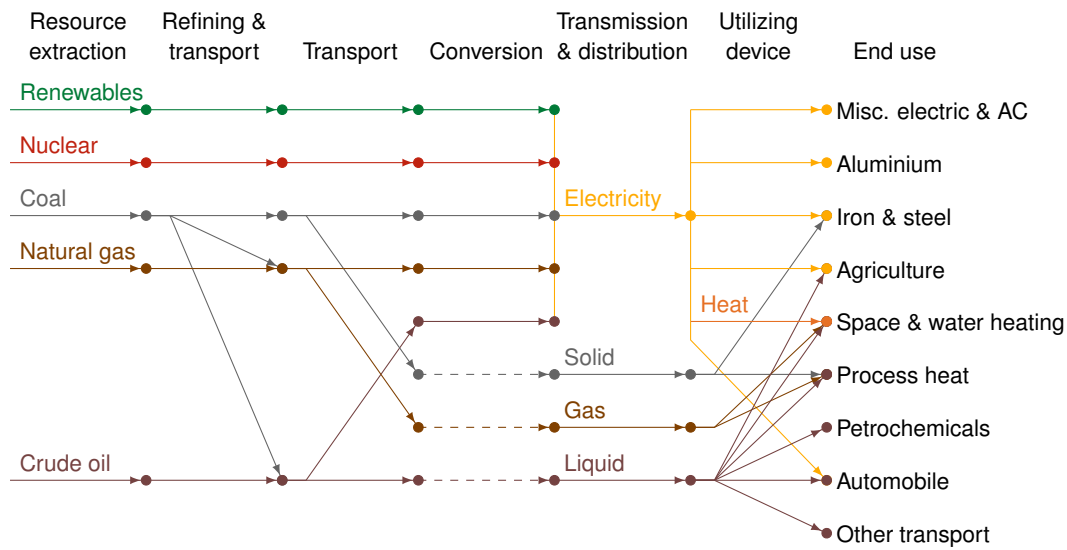


Figure 2.10: Exemplary RES diagram of a MARKAL model. Reproduced from [108, p.6]

programming (NLP) problem. Variant MARKAL-MACRO [69, p.389] merges the standard version with a macro-economic model for optimum inter-temporal economic growth.

While MARKAL allows insights into the aggregate behaviour of the energy sector as a whole, it lacks detail into technical properties of the energy supply chain. These limitations lead to the development of TIMES.

## TIMES

The Integrated MARKAL-EFOM System (TIMES) [71] is an economic model generator for local, national or multi-regional energy systems. It allows to estimate and optimise the development of energy systems over long time periods.

It is formulated as a linear-programming maximisation problem of the NPVs, which provides a technology rich basis for estimating the development of the energy system over a long-term time horizon. It has been developed in the framework of the Energy Technology Systems Analysis Programme (ETSAP) implementing agreement of the International Energy Agency (IEA).

TIMES is defined as a bottom-up technology rich optimisation model generator. The equilibrium is driven by the maximization (via linear programming) of the discounted present value of total surplus, representing the sum of surplus of producers and consumers, which acts as a proxy for welfare in each region of the model. The maximisation is subject to many constraints, such as: supply bounds for the primary resources, technical constraints governing the creation, operation, and abandonment of each technology, balance constraints for all energy forms and emissions, timing of investment payments and other cash flows, and the satisfaction of a set of demands for energy services in all sectors of the economy. Energy and environmental policies can be represented and analysed with accuracy, due to the explicit representation of technologies, fuels, energy related emissions and materials use, in all sectors of economic activity.

Using TIMES model generator, a TIMES-based European model was developed and improved within various projects under the 6th and the 7th Framework Programme. In this model, the energy systems of EU-27 countries, Iceland, Norway and Switzerland are modelled separately in detail and then synthesized by allowing trade of energy commodities among the countries.

## MESSAGE

The Model for Energy Supply Strategy Alternatives and their General Environmental Impact (MESSAGE) [82] is another widely used example of a RES model. Similar to MARKAL, it is based on minimising discounted total costs to satisfy given demands for a set of commodities. Special features include elasticities for demand and supply, discrete sizing of investment decisions (if a mixed integer solver is used), and even non-linear objective functions. Its roots go back to the 1970s, when the first version was released [2]. Version 2 followed in 1984 [81], version 3 in 1995 [82]. Dozens of studies all over the world have used and extended the core model with custom features like stochastic or micro- and macro-economic modules similar to MARKAL. Model extension MAGICC has been developed to model greenhouse gas (GHG) emissions and thus allowing comparison of the effectiveness of different mitigation schemes.

While the model implementation is not freely available, the model core has been extensively documented [82]. A unique feature of the studies conducted at International Institute for Applied Systems Analysis (IIASA) is their database of inputs and results available for review and download as a web application [53].

## Energy hubs

Energy hubs are the application of linear algebra to represent a general multiple input multiple output energy conversion process by matrix vector multiplication. The concept was presented by Geidl in 2007 [31] and then further expanded to formalise a whole energy system optimisation problem [30]. The core idea is based on the following equation:

$$\mathbf{p}_{\text{out}} = \mathbf{C} \cdot \mathbf{p}_{\text{in}} \quad (2.30)$$

Here,  $\mathbf{p}_{\text{in}}$ ,  $\mathbf{p}_{\text{out}}$  are power flow vectors (kW) and  $\mathbf{C}$  is the coupling matrix, a generalised form of the scalar efficiency factor  $\eta$ , which is used for single input single output processes. This method is mentioned here briefly, because it is very similar to the formulation that will be used in model **urbs** later to represent MIMO processes. There, an additional state variable is employed that conceptually resides in between inputs and outputs.

## The method URBS

Richter [101] presented the original implementation of the method Urban Research Toolbox: Energy Systems (URBS) in his doctoral thesis. The model employs a RES representation of the energy system for electricity and heat in Augsburg, aggregated by technology type. As the main focus was not on developing a reusable optimisation model, a full mathematical description of the model is not given, and the code of the original implementation has not been published. He lists models deeco [14, 15] and MESAP [124] as inspiration for the energy system model.

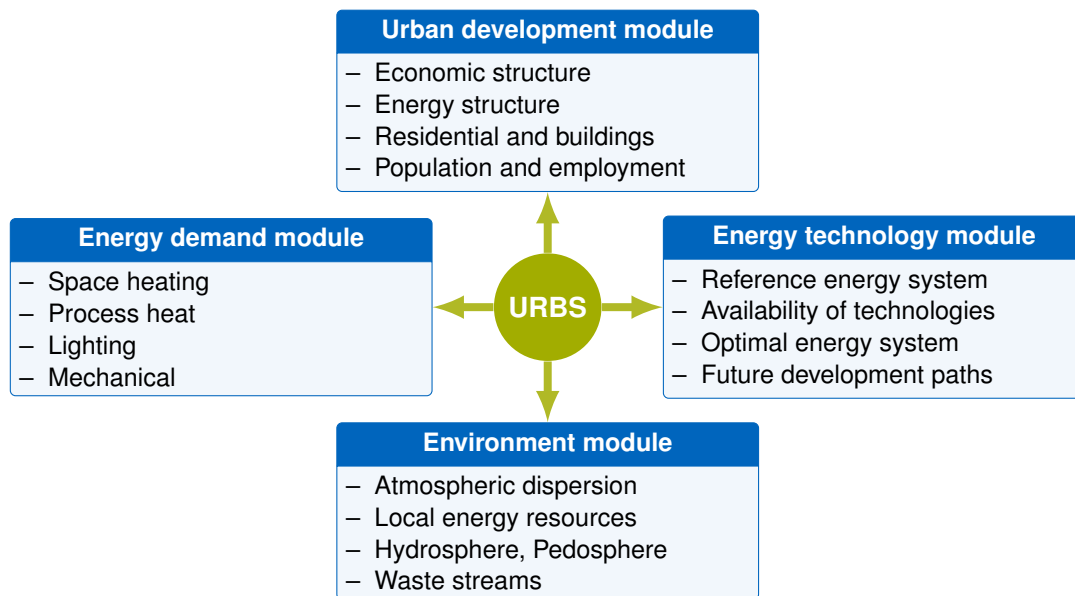


Figure 2.11: Graphical overview on the original method URBS. Based on [101, Figure 2.1]

Figure 2.11 shows a reproduction of the original model overview. The original method URBS not only includes an optimisation model for energy infrastructure, but also three simulation models for predicting urban development, energy demand and environmental effects.

Refer to Sections 3.1 and 3.2 for two similar, but not completely identical models with similar features. The author of this thesis became maintainer of **URBS** for investigating the costs for realising the DESERTEC vision of coupling renewable electricity production among Europe and North Africa, leading to a low-carbon electricity system for both regions. The result of this study was published together with Matthias Huber as a working paper [50]. As this investigation is definitely not on urban scale, it is not further discussed here, but might be interesting for readers planning to conduct energy system optimisation at international scale. For another application at building scale, please refer to a paper by Huber [52], in which the model was used for optimisation of independent/cooperative electricity and heat generation for a group of ten houses. **URBS** was also applied on the international level in various publications [103, 104], most recently by Schaber [102] and Huber [51].

### 2.3.2 Network flow models

The theoretical “groundwork” has been well described by Groscurth, Bruckner and Kümmel in 1995 [40]. Models **dhmin** and especially **rivus** are formally very similar to the more general formalism presented there. A more recent version of a similar framework has been presented in 2012 by Holden et al. [45].

Bentley sisHYD [8] is an example of a commercial network analysis tool that is able perform hydraulic calculation of a pressure pipe system with (in)compressible media, i.e. district heating, district cooling, and gas networks. Starting from GIS or CAD data, a network model is constructed, optionally supported by technical parameters supplied in form of spreadsheets. Hydraulic calculation is static, thermal calculations can be performed dynamically. The result

includes media flow rates, pressures at network nodes, temperatures and (thermal) losses throughout the network.

Such models allow a very detailed simulation of the all state variables of a network. These include flow rates, pressure levels, temperatures and losses. Their application is mainly in operation of existing networks. They are used to play through the outcome of operational decisions to improve flows for minimum losses. Also, different generation schedules can be compared to different predicted (or measured) loads.

Concerning planning of district heating networks, the doctoral thesis by Blesl from 2002 [11] presents a very detailed optimisation approach. Using a multi-period formulation of the investment cost structure for installing district heating network in urban areas, he confirmed that district heating remains an efficient option for areas of high demand density.

Not directly related to district heating or gas network is the excellent book *Mathematical Optimization of Water Networks* [76]. It first outlines the basics of linear, mixed integer, and mathematical optimisation in general; then practical modelling examples for water supply and sewer networks are discussed in detail.

### 2.3.3 Urban energy system models

The latest review on these models is from 2012 by Keirstead et al. [58]. The review begins by discussing how to define the term *urban energy system*. Relying on an article by Ramaswami [98], one could define the system boundaries of a city by three criteria:

**Geographic** Local production within the city boundaries. This limits the extent of the analysis to local plants only, excluding any upstream effects like the composition of imported electricity.

**Geographic-plus** Local production plus traceable supply chains, like electricity generation and big investment good flows. While it is an improvement in analysis quality, this approach still does not track the life cycle of every component in any product, good or process.

**Consumption** Energy consumption/emissions are allocated to the consuming household, not the location of consumption. For example, international tourism is allocated to the residence of the traveller, not the travel destination.

An optimisation model for district cooling systems has been presented in 2007 by Södermann [115]. It is formally very similar to model **dhmin** presented in section 3.4.

Doctoral thesis by Voll from 2013 [122] uses an iterative evolutionary synthesis process, combined with MILP to create a hybrid searching the design space for distributed energy supply systems. Noteworthy features of this approach are that both optimal (time consuming) and near optimal (less time consuming) structures can be synthesised.

The doctoral thesis by Heilek from 2015 [42] last investigated the coupling of power and heat sector at urban scale. Its findings: especially power-to-heat technologies, coupled with local heat storage or distribution are promising candidates for integrating large amounts of intermittent renewable electricity.

Jennings, Fisk and Shah [55] have recently presented a model that is strongly focused on the trade of between demand side and supply side investments for retrofitting residential energy systems.

### 2.3.4 Closed source models

Very prominent is HOMER Energy [46], which used to publish older releases of its software as freeware, but discontinued that practice recently. As of writing, a discount for academic licenses is given. The core of the model is described in a book chapter by Lambert from 2006 [66]. The model has a very gentle learning curve due to its discoverable graphical user interface (GUI). However, the model lacks capabilities for capacity expansion planning. While scheduling is done using a heuristic with foresight, capacities must be fixed for a simulation run. Optimisation can only be performed by specifying a list of discrete considered capacities for each process, followed by exhaustive search of each combination. This approach suffers from combinatorial explosion in all but the easiest cases. If only one node is required, the complexity is manageable this way. On very small scales, the discrete capacities are actually a benefit in accuracy, as small unit sizes are often only available in a small selection of discrete steps.

### 2.3.5 Open source models

In recent years, more and more optimisation models for energy systems in general and infrastructure planning in particular made the move to not only publish detailed technical manuals, but their actual code.

Under the common initiative called **openmod** [90], several energy modellers have come together to share their modelling efforts. Several models with accessible source code and – partly – input data used for publications can be found. These are currently:

Balmorel is an optimisation model for electricity and heat sectors developed by Hans Ravn. It allows both for intertemporal capacity expansion planning and short time unit commitment planning [38]. Its model core is based on GAMS. A graphical user interface and data processing accessories based on MS Office are available upon request.

Calliope [94] by Pfenninger [96] is maintained at Imperial College, London. It is a generic and versatile optimisation framework for reference energy system-style energy systems with variable spatial and temporal resolution. Input is to be provided in both machine and human readable YAML file format. The framework allows for rapid development of custom tool chains around a flexible model core. Technically, the model is based on Pyomo. Pandas DataFrames are used as the result data format.

EMLab-Generation by Richtstein et al. [100] is developed at Delft University of Technology. It uses an agent-based modelling approach. This allows the user to explore impacts of suboptimal investment decision and inequalities between actors. Technically, it is based on Java and uses R for data processing.

OSeMOSYS [49, 125] has been contributed to by several authors and is currently maintained at KTH Royal Institute of Technology. It is designed for optimal planning of energy systems over decades to a century long horizon. Technically, it is based on GNU MathProg, a mathematical description language similar to AMPL. The generated optimisation problems can be solved by open source solver GNU Linear Programming Kit (GLPK), making it an attractive choice for use in classrooms.

Oemof [62] – short for Open Energy System Modelling Framework – is a joint development of Rainer Lemoine Institute in Berlin and Zentrum für nachhaltige Energiesysteme in Flensburg.

It is a framework developed in Python, providing packages for modelling and optimising generic energy systems.

Renpass (Renewable Energy Pathways Simulation System) is developed by Wiese [128] and comprises an electricity unit commitment model for Europe. It consists of 21 regions and finds minimum cost scheduling of power plants to satisfy loads. It is based on R and various packages for data processing.

SciGRID [80] is an open source model of the European electricity transmission grid. It is based on open data, which is pre-processed for modelling purposes to fix mapping errors or missing data.

Temoa is an energy system optimisation toolbox developed by DeCarolis [21]. It uses a formalism similar to MARKAL. It has the concept of a reference energy system with resources as energy sources, transformation technologies as possible intermediate steps. Like MARKAL and TIMES, it has the concept of two time scales (season, time-of-day) to simultaneously allow inter-temporal investment planning, while not completely having to sacrifice the capability to model energy storage technologies. Technically, it relies on the COmmon Optimization Python Repository (Coopr)/Python Optimization Modeling Objects (Pyomo), just like the models presented in this thesis.



## 2.4 Algorithms

This section summarises all algorithms used in this thesis. For standard methods, only a brief mention and a literature reference are given. For specially developed processes, full descriptions with pseudo-code are given. The presented algorithms range from optimisation (linear and mixed integer), to handling of time series data for deriving temporal and generalising spatial data to simplify road networks.

### 2.4.1 Optimisation

While section 2.2 already gave a short overview on the theory behind mathematical optimisation, this section points out the standard methods typically implemented by optimisation solver programs. For more in-depth discussion and especially newer developments, the reader is referred to the referenced literature.

#### Linear programming

The problem of solving linear programs has been tackled in the 1960s with the rising availability of computing hardware. The most commonly used algorithm today is the simplex method by Dantzig [20], which exploits linearity and convexity of the problem structure. Let's recall equation (2.10), the canonical LP:

$$\begin{aligned} \max_{\mathbf{x} \in \mathbb{R}^N} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{Ax} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

We can observe that maximising the expression  $\mathbf{c}^T \mathbf{x}$  is geometrically equivalent to “going” in direction of vector  $\mathbf{c}$ , until constraints become active or the problem is deemed unbounded. The constraint set defined by the two sets inequalities  $\mathbf{Ax} \leq \mathbf{b}$  and  $\mathbf{x} \geq \mathbf{0}$  has the form of a polytope in the vector space  $\mathbb{R}^N$  of possible solutions. A key observation now is that except for degenerate cases, the optimal solution lies on one of the vertices of that polytope, a so-called *extreme point*. The *simplex algorithm* exploits that property by traversing the vertices on the boundary of the feasible region. While that procedure theoretically has exponential worst-case runtime, most practical optimisation problem allow a nearly polynomial solution time. More recent solvers also employ *interior point methods* [86] to cut short the way to the optimum by finding *shortcuts* through the feasible region.

#### Mixed integer linear programming

In mixed integer programming, an integrality constraint is added to some entries of the variable vector  $\mathbf{x}$  to arrive at equation (2.13). Despite the dramatic reduction in size of the feasible region, integer programming in general is much harder than an identically sized linear problem. A common solution algorithm called *cutting plane* solves a sequence of *relaxed* linear programs. Relaxed means that the integrality constraints are dropped. If the optimal solution of that relaxed problem only contains integers, it is also the optimal solution to the original problem. If not, additional constraints are added to the relaxed problem and that modified problem is

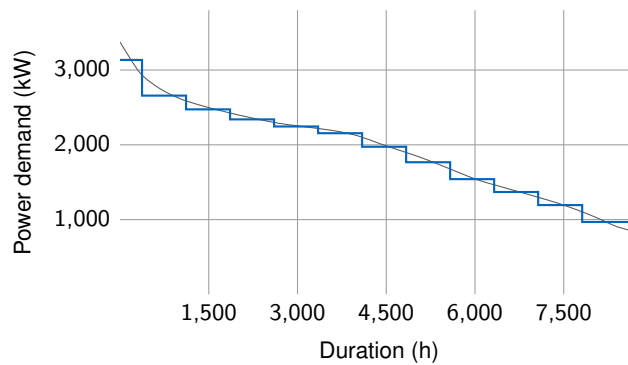


Figure 2.12: Annual load duration curve discretised by 15 samples

then solved again, until no more violations of the integrality conditions occur. Other heuristics create a conceptual tree by fixing the value of one integral variable at a time and comparing the relaxed optima for various values. Depending on the constraint generation strategy, various heuristics like branch-and-bound or branch-and-cut have emerged. The classic textbook by Schrijver [106, part IV] gives a comprehensive overview on the classic methods, while a review paper by Sherali and Driscoll from 2000 [109] and a thesis by Achterberg from 2009 [1, part II] discuss more recent developments.

### 2.4.2 Load duration curve clustering

In section 2.2.2, reducing the resolution of energy demand timeseries was discussed. The following paragraphs describe the use of clustering algorithms for reducing the number of datapoints to represent the annual load duration curve(s) for one and multiple simultaneous energy demands.

#### Single load duration curve

Discretising a single annual load duration curve can be described intuitively as shown in figure 2.12. It shows that for a single load duration curve, its discretisation can be reduced to finding a set of weights  $w_t$  (or durations) and levels  $s_t$  (or scaling factors) for a given curve with a higher original temporal resolution.

The simplest method for reducing the resolution of such a curve is a fixed width sampling. This is analogous to the conversion of a time-continuous signal to a discrete-time sequence of values. However, this naive sampling leads to suboptimal representation of the original load curve.

More elaborate algorithms take into account the slope of the load duration curve and increase the weight of samples at flat sections of the curve, while using smaller weights for sample points at steep sections. The doctoral thesis of Kuhn [64, ch. 4.3.4] describes one such algorithm for variable weight discretisation of a single load curve. It determines the weights and thus the position of the sampling points in order to minimise the squared error between original and discretised curve.

- 
- 1 select  $K$  observations randomly as initial cluster centroids
  - 2 **repeat**
  - 3     assign each observation to the closest centroid
  - 4     recalculate the cluster positions based on the centroids of the assigned observations
  - 5 **until** cluster centroids do not move anymore
- 

Algorithm 2.1: Original K-Means algorithm

### Multiple load duration curves

The generalisation of multiple simultaneous load curves on the other hand can be visualised better as a clustering problem, as shown earlier in figure 2.6. The bachelor thesis of Harmat [41] discusses five different cluster analysis algorithms and applies them to load curves: DiAna, fusion algorithm, K-Means, DBSCAN and representative point selection.

To quantify how suitable each clustering algorithm is for load curve discretisation, four indicators were calculated and compared: *run time*, *peak load preservation*, and *energy demand conservation*. *Run time* simply measures the wall clock execution time of each algorithm. *Peak load preservation* and *energy demand conservation* indicate how much these two properties of the original load curves are preserved in the returned demand clusters.

For the purposes of this thesis, the K-Means approach showed the most desirable properties: simplicity, speed and perfect peak load preservation through a modification of the original algorithm. Both the conventional K-Means algorithm and the modified version developed by Harmat [41, sections 4, 6.2.4] are thus presented here.

Simultaneous demand values for  $D$  different commodities can be interpreted as a sequence of  $D$ -dimensional vectors, one dimension per demand commodity. Here, two dimensions or demand commodities (electricity and heat) are used. Therefore, a single observation  $\mathbf{x}_i \in \mathbb{R}^2$  is formed from the simultaneous heat demands for electricity  $P_{\text{elec}}(i)$  and heat  $P_{\text{heat}}(i)$  at instance  $i$  by the expression

$$\mathbf{x}_i = \begin{bmatrix} P_{\text{elec}}(i)/P_{\text{elec}}^{\max} \\ P_{\text{heat}}(i)/P_{\text{heat}}^{\max} \end{bmatrix} \quad \forall i = 1 \dots N, \quad (2.31)$$

where the divisor is the maximum value for each individual time series, e.g.  $P_{\text{elec}}^{\max} = \max_i P_{\text{elec}}(i)$ . The normalisation by the maximum value is used to give the *relative* differences between different load situations more weight than the *absolute* differences. If no normalisation was used, clustering would basically only differentiate the thermal loads, while electrical loads are more similar due to their lower absolute value (at least in the investigated case study regions).

With these definitions in place, first the unmodified K-Means algorithm is shown in algorithm 2.1. Given a possibly long list of observations  $X_i$ , the procedure returns a shorter list of  $K$  cluster centroids  $\mathbf{C}_k$ ,  $k \in 1 \dots K$ . The number of observations assigned to each centroid is called the cluster weight  $W_k$ . Steps 3 and 4 iteratively recalculate the centroid position of a cluster and then use its updated position to re-classify the observations, which is also called LLoyd's algorithm.

In the original form, the algorithm is bound to never represent the observations that correspond peak loads as cluster centres, thereby neglecting the most critical instants for

- 
- 1 select  $K - D$  observations randomly as initial cluster centroids
  - 2 assign the remaining  $D$  centroids to the maximum observation for each dimension
  - 3 **repeat**
  - 4     assign each observation to the closest centroid
  - 5     recalculate the cluster positions based on the centroids of the assigned observations
  - 6     the position of the centroids assigned in step 2 are not moved
  - 7 **until** cluster centroids do not move anymore
- 

Algorithm 2.2: Modified K-Means algorithm for load curve discretisation. Based on [41, listing 6.4]

capacity planning. In order to preserve the observations that correspond to the peak demands for each commodity, Harmat developed a customised K-Means algorithm shown in algorithm 2.2 (modifications highlighted in blue). For each demand commodity, the peak load observation is selected as one of the cluster centroids. During Lloyd's algorithm, the position of these centroids is kept constant, while the remaining ones are allowed to move. This results in a similar output, but with the knowledge that some clusters will preserve the peak value of the original load curve.

In the case studies shown in this thesis,  $D = 2$  for electricity and heat demand, but the procedure readily generalises to an arbitrary number of simultaneous load curves, as long as the number of clusters  $K$  is chosen sufficiently larger than the dimensionality  $D$ .

The cluster centroids and weights then have the following meaning in the context of the optimisation model **rivus**, where  $s_{ct}$  is the demand scaling factor for demand commodity  $c$  and  $w_t$  is the time step weight:

$$\forall t \in T: \quad \left[ s_{c_1 t} \dots s_{c_D t} \right]^T = \mathbf{C}_t \quad (2.32)$$

$$\forall t \in T: \quad w_t = W_t \quad (2.33)$$

### 2.4.3 Geographic operations

Physically, an urban area is characterised by its buildings and transport networks (streets, rails). In this thesis, these elements also serve as a natural basis for locating energy demand and the energy supply infrastructure. Geographic operations deal with geometric primitive shapes in the form of collections of points, lines and polygons. The common term for these primitive shapes is **geometry**. This section briefly describes the procedures used to perform the spatial analyses presented in chapter 4.

In general, geographic operations happen on either raster or vector data. Raster data, like a pixel image, contains equidistant data points on a rigid grid with fixed extents. Vector data, like a vector image, consists of a collection of geometric primitives described by exact coordinates. In this thesis, only vector data is used:

A **point** on the surface of the earth is generally represented by a tuple (longitude, latitude) of geographic coordinates, given in degrees. Longitude is the angle east or west relative to the Greenwich meridian line. Latitude is the angle north or south relative to the Equator. A **line** then is an ordered list of at least two distinct points. If all the line segments comprising a line do not intersect with each other, a line is called simple. A line is called closed, if its start and

end point are identical. A **polygon** is then defined by the tuple (exterior, interiors): Exterior is a closed line defining the polygon outline. Interiors is a list of closed lines defining holes inside the area delimited by exterior. Generalisations of the line and polygon concept to multi-part geometries are also used to simplify the definition of algorithms: a so-called **multi-line** is collection of lines. A **multi-polygon** is a collection of non-overlapping polygons.

Without a formal definition, the following list of basic operations on geometry objects is given. For a more comprehensive discussion, a textbook on geographic information system (GIS) [68] is recommended.

**Buffering** a geometry enlarges it by a given length in all directions. A point becomes a circle; a line becomes a (lengthy) polygon; a polygon simply grows outward, with holes in the interior possibly vanishing. This property is used by the skeletonisation algorithm below to generalise densely connected street segments.

**Dissolving** a geometry shrinks it by a given length in all directions. It is the inverse of the buffer operation, though it cannot correctly reverse the operation if the buffering step has blurred line segments or swallowed interior polygon holes.

**Centroid** of a polygon calculates the geometric centre of its area. In a physical analogy, it returns the centre of mass of a homogeneous body that has the same shape as that polygon. This operation is used to determine the centre point of building outlines before determining its closest street segment.

**Length** of a line returns the sum of its line segment lengths. For geographic coordinates, these lengths have to be calculated using the great-circle distance. For correctly projected coordinates, the euclidean distance can be used with sufficient accuracy. For more details on geographic coordinate systems, refer to section 2.6.4.

**Area** of a polygon calculates the size of the region enclosed by a polygon. For large polygons (as in earth curvature cannot be neglected), this calculation must be executed using geographic coordinates. Here, this operation is used to estimate the gross floor area of building outlines. Depending on the type of coordinates, the same caveats as in the *length* operation apply.

With these definitions in place, the following more complex spatial operations are introduced: *aggregation* of demand from buildings (polygons) to their nearest street (lines); *graph skeletonisation* to reduce the number of vertices and edges to describe a given street network.

### Demand aggregation

For aggregating per-building data to a lower spatial resolution, multiple methods exist: a simple raster with given x-y resolution could simply sum up or average statistics for all buildings per raster cell. Or one could manually specify district polygons or use administrative borders to derive them automatically. For the network models presented in this thesis, a method that respects the topology of the urban environment more adequately is needed. Therefore, per-building data is aggregated per street segment. For each building, the street segment closest to its centroid is identified. The relevant figure 2.13 shows the result of this operation. The implementation is published on GitHub [22] for reference, together with helper functions to read and write shapefiles.

---

```

1 function AGGREGATE(buildings ( $b \in B$ , area  $A$ , type  $k$ ), street graph  $G = (V, E)$ )
2   let  $K$  be the set of all building types
3   initialise edge area sum matrix  $A_{ek} = 0 \forall e \in E, k \in K$ 
4   for all  $b \in B$  do
5     set  $A$  and  $k$  to the area and type of  $b$ 
6     calculate the centroid  $c \in \mathbb{R}^2$  of  $b$ 
7     set  $e$  to the edge closest to  $c$ 
8     increase  $A_{ek}$  by  $A$ 
9   end for
10  return  $A_{ek}$ 
11 end function

```

---

Algorithm 2.3: Area aggregation

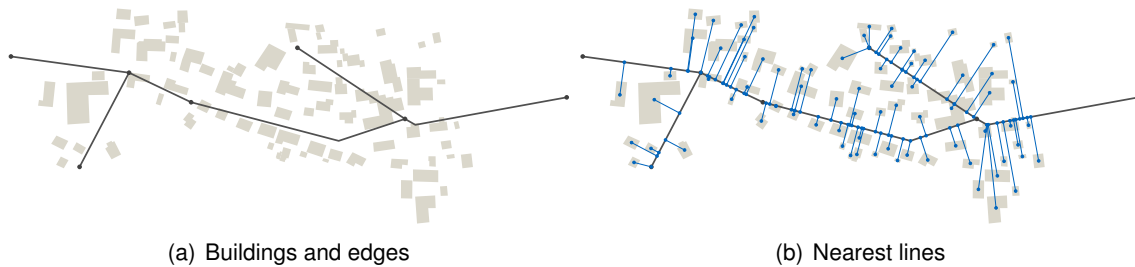


Figure 2.13: Steps of the near operation for finding closest edges

Models **DHMIN** and **dhmin** use this aggregation without the type attribute  $k$  (i.e. the set  $K$  has a single element) to simply sum *all* demands of buildings to their closest edge  $e$  in a given graph.

Model **rivus** depends on this aggregation, summing all building floor areas by building type. These floor areas are then used to estimate peak energy demands for electricity and heat. Refer to section 3.5.3 for how the aggregated demand is used as an input parameter.

### Graph skeletonisation

Polygon skeletonisation usually refers to deriving the *straight skeleton* of a two- or three-dimensional polygon. The more general term is *medial axis*, which is defined for any geometric object as the set of all points that have more than one closest point on the objects boundary. It is a common algorithm in computer vision that is used to construct offset curves and miter joints. Here, the straight skeleton of a polygon derived from raw street lines is used to accomplish a generalisation of the input network.

Algorithm 2.4 is a short description of the steps for finding the idealised street graph from raw street lines. Figure 2.14 summarises the steps graphically. It is adapted from description by Ladak [65]. First, the original street segments are buffered and merged to form one huge, connected polygon with many holes. This polygon then is fed to `qhull` [6] using the wrapper API `Skeleton` [83] to derive its straight skeleton. The wrapper automatically prunes any line segments that touch the polygon boundary. Internally, a Voronoi diagram of the

---

```

1 function SKELETONIZE(road lines,  $l_{\text{buf}}$ ,  $l_{\text{dis}}$ ,  $l_{\text{sim}}$ ,  $l_{\text{psg}}$ )
2   buffer the list of roads lines by length  $l_{\text{buf}}$ 
3   simplify the outline of the resulting road polygon by  $l_{\text{sim}}$ 
4   if number of polygon elements is greater than 1 then
5     only keep largest polygon element by length of outline
6   end if
7   dissolve the road polygon by length  $l_{\text{dis}}$ 
8   procedure SKELETRON.POLYGON_SKELETON_GRAPH(road polygon,  $l_{\text{psg}}$ )
9     sample the road polygon outline at distance  $l_{\text{psg}}$ 
10    determine the voronoi graph of the resulting point cloud
11    intersect the resulting line network with the road polygon
12    truncate all dangling lines
13  end procedure
14  simplify the remaining road graph by  $l_{\text{sim}}$ 
15  return the road graph
16 end function

```

---

Algorithm 2.4: Skeletonisation of road networks, with sketch of the subprocedure by `SkeLetron` [83]

polygon vertices (at fixed distances) is calculated. This results in a noticeable *zigzaging* of the raw output, shown in figure 2.14(c). A final line simplification step then straightens the line, resulting in a clean street graph. As side benefit from the initial buffer step, local resolution is automatically reduced in very dense areas. Several parameters can be tuned to adapt the result to one's requirements. The main parameters and their default values are the buffer distance  $l_{\text{buf}} = 60$  m, dissolve length  $l_{\text{dis}} = 30$  m, Voronoi vertex distance  $l_{\text{psg}} = 150$  m, line simplification threshold  $l_{\text{sim}} = 30$  m. The implementation is published online [23] for reference.

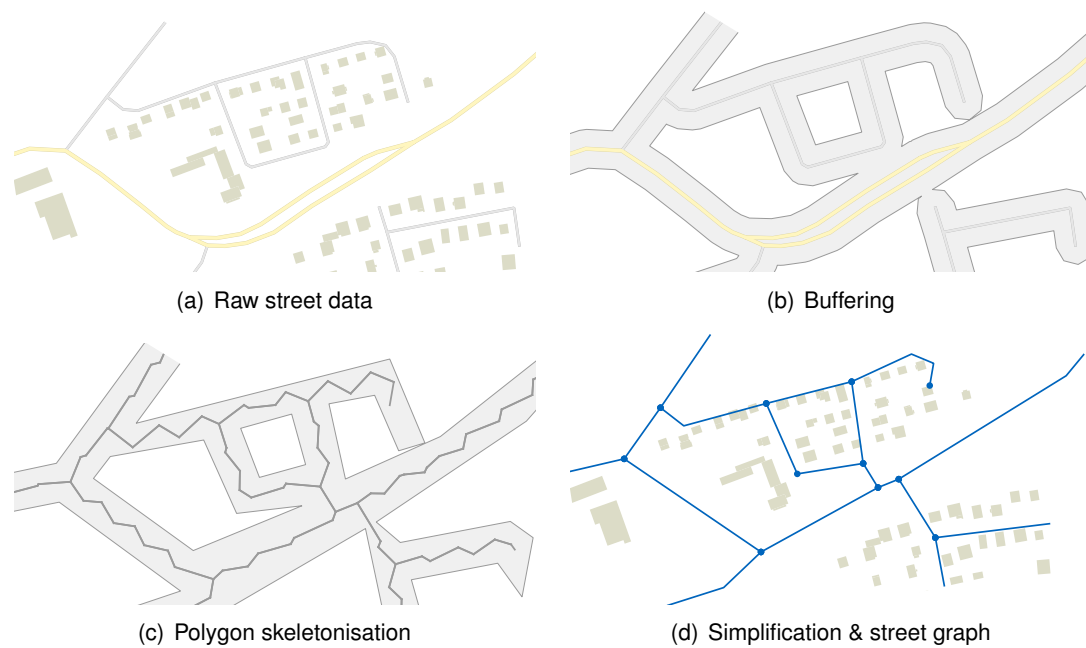


Figure 2.14: Steps of the skeletonization process for deriving a street graph. Raw street data (a) shows a fraction of exemplary input. Step buffering (b) merges adjacent lines and connects nearby edges. Skeletonisation (c) reduces the polygon back to a (now connected) simplified line network. Simplification (d) finally reduces *zigzaging* and introduces only vertices at line crossings.



## 2.5 Software

This section shortly discusses the software packages that were investigated during the course of this thesis. The last subsection then presents the tool chain that is used for the models presented in the following chapter. As the selection largely depended on suitable data formats, section 2.5.2 is of importance, too. Depending on the actual optimisation problem, a good data format can facilitate the collaboration of different software tools.

To start, figure 2.15 shows a stylised modelling tool chain that is used to answer a generic research question that is assumed to be accessible by mathematical modelling. At each step, various software packages are available; some popular choices are listed in the figure. The initiative Computational Infrastructure for Operations Research (COIN-OR)<sup>2</sup> is a good place to start looking for a maintained list of other potentially useful tools.

### 2.5.1 Modelling languages

Optimisation problems can be stated in different ways. The most fuzzy, but usually first way is to state the objective and constraints in natural language, for example:

Find the energy supply infrastructure network that satisfies a given set of energy demands for minimal cost.

The next step lies in formalising this abstract statement into concrete mathematical objects. Algebraic modelling languages allow using set theory to specify collections of objects, and numeric values to describe known quantities (parameters) and unknown values (variables). In

<sup>2</sup><http://www.coin-or.org/projects/>

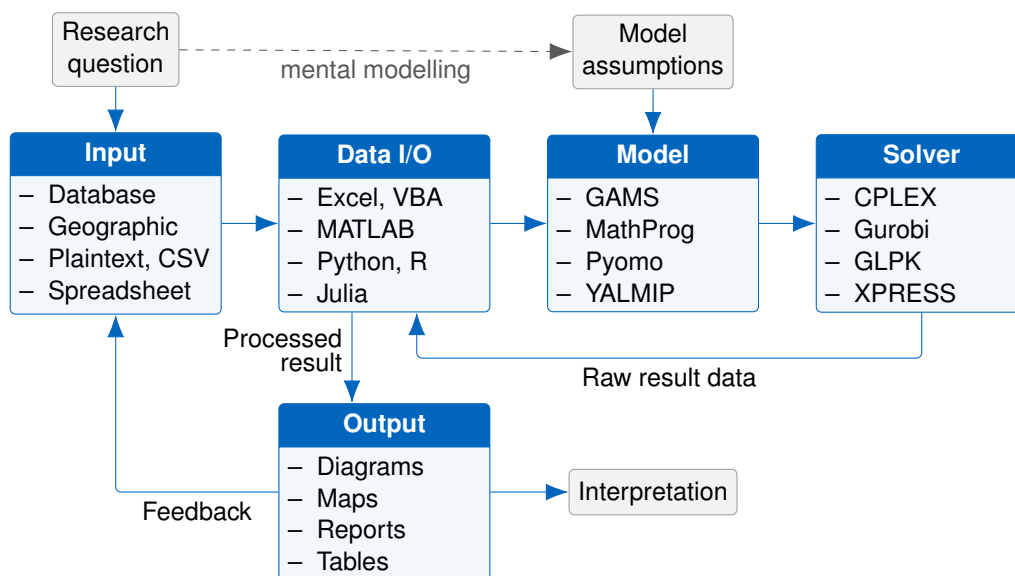


Figure 2.15: Modelling tool chain with stages and exemplary software packages

figure 2.15, this step is represented by the dashed line labelled *mental modelling*. An abridged result of this step could look something like this:

$$\begin{aligned} \min_{\epsilon_{pt}, \kappa_p} & \sum_{p \in P} \left( c_p^{\text{inv}} \kappa_p + c_p^{\text{var}} \sum_{t \in T} \epsilon_{pt} \right) \\ \text{s.t.} & \sum_{p \in P} \epsilon_{pt} \geq d_t \quad \forall t \in T \\ & \epsilon_{pt} \leq \kappa_p \quad \forall t \in T, p \in P \end{aligned}$$

At this stage, the model can be translated into a machine-readable format using an algebraic modelling language. This section presents the most common languages with a short estimate of their strengths and weaknesses.

### AMPL

A Mathematical Programming Language (AMPL) is probably the most common mathematical modelling language. It offers domain-specific language for concise description of mathematical optimisation problems. The language elements `set`, `param`, and `var` define sets (effectively the problem size), parameters (fixed values), and variables (values to be found by the solver). The statements `minimize` or `maximize` define a cost function, while `s.t.` (“so that”) are used for constraints. Expressions support a looping syntax inspired by mathematical sum notation, here demonstrated by a short example. Left is mathematical notation, right is AMPL syntax:

$$\forall t \in T: \sum_{p \in P} \epsilon_{pt}^{\text{out}} \geq d_t \quad \text{s.t. } \text{satisfy\_demand}\{t \text{ in } \text{time}\};$$

```
sum{p in plants} output[p, t] >= demand[t];
```

For solving a concrete optimisation problem, this description plus the input data (set element, parameter values) are combined and compiled to an explicit problem statement that is handed to the solver. Most of the following software packages implement a subset of AMPL language features.

### GAMS

In operations research, the language provided by software General Algebraic Modeling System (GAMS) is commonly used. Its syntax resembles AMPL closely, extended by a macro system that allows for (limited) automation of model runs. Listing 2.1 shows the example of a full optimisation model. The system bundles solver and IDE in a single application that allows fast iteration during model development. The language offers macros to access data in text files, spreadsheets and even databases. Its strengths are the easy installation, good documentation, integration of modelling environment and solver access. The main weakness lies in the high price for the IDE, and the fact that solvers licensed for GAMS can only be used through GAMS, not independently.

### MathProg

GNU MathProg is a special-purpose modelling language for mathematical modelling. It implements a subset of the full AMPL specification and has similar syntax. Listing 2.2 shows an

Listing 2.1: GAMS example model fuelstation.gms

---

```

1 Sets
2   t          time                / 1*8760 /
3   i          type of production / pv, windon, windoff /
4   tfirst(t) first timestep
5   tlast(t)  last timestep ;
6   tfirst(t) = yes$(ord(t) eq 1) ;
7   tlast(t)  = yes$(ord(t) eq card(t)) ;
8
9 Parameters
10  cs        cost of storage tank (kEUR per MWh) / 100 /
11  c(i)      cost of plant (kEUR per MW) / pv 3000, windon 1500, windoff 2500 /
12  d(t)      demand (MWh)
13  p(t,i)    relative (normalized to 1) production of plants;
14  d(t) = uniform(0,1) ;
15  p(t,'pv') = min(max(0, power(sin(ord(t)/24*3.14/2),4)+normal(.1,.1)),1);
16  p(t,'windon') = min(max(0, uniform(0,1)),1);
17  p(t,'windoff') = min(max(0, sqrt(sqrt(uniform(0,1)))),1);
18
19 Variables
20  x(i)      size of production facilities (MW)
21  s         size of accumulator (MWh)
22  st(t)     evolution of accumulator SOC (MWh)
23  tp(t)     total production of plants per timestep (MWh)
24  w(t)      wasted energy (MWh)
25  z         total cost (kEUR);
26
27 Positive Variables x, s, st, waste;
28
29 Equations
30  cost      definition of objective function
31  pp(t)     calculates tp (total production) from p and x
32  dd(t)     assures that demand is always satisfied
33  storage(t) new_storage = storage + input - demand
34  ss(t)     restrict storage content by its size
35  ss0(t)    define initial storage content (half filled)
36  ssN(t)    require final storage content (half filled);
37
38  cost..    z =e= sum(i, x(i)*c(i)) + cs*s;
39  pp(t)..   tp(t) =e= sum(i, p(t,i)*x(i));
40  dd(t)..   d(t) =l= tp(t) + st(t);
41  storage(t).. st(t+1) =e= st(t) + tp(t) - d(t) - w(t);
42  ss(t)..   st(t) =l= s;
43  ss0(tlast).. st(tlast) =g= 0.5 * s;
44  ssN(tfirst).. st(tfirst) =e= 0.5 * s;
45
46 Model    fuelstation / all / ;
47
48 Solve    fuelstation using lp minimizing z ;

```

---

Listing 2.2: MathProg example model (full model in listing B.1)

---

```

1 set time;
2 param demand{time} >= 0; # (kWh/h)
3 param supply{time} >= 0; # (kWh/h)
4 param electricity_price{time}; # (EUR/kWh)
5 param storage_cost; # (EUR/kWh)
6
7 var energy_balance{time}; # (kWh)
8 var storage_capacity >= 0; # (kWh)
9 var storage_level{time} >= 0; # (kWh)
10 var energy_purchase{time} >= 0; # (kWh)
11 var energy_sold{time} >= 0; # (kWh)
12 var costs;
13
14 minimize obj: costs;
15
16 s.t. def_costs:
17     costs =
18     storage_cost * storage_capacity +
19     sum{t in time} electricity_price[t] * energy_purchase[t] -
20     sum{t in time} electricity_price[t] * energy_sold[t];
21
22 s.t. def_balance{t in time}:
23     energy_balance[t] = supply[t] - demand[t] +
24     energy_purchase[t] - energy_sold[t];
25 s.t. def_storage_state{t in time: t>1}:
26     storage_level[t] = storage_level[t-1] + energy_balance[t];
27
28 s.t. def_storage_initial{t in time: t=1}:
29     storage_level[t] = 0.5 * storage_capacity;
30
31 s.t. res_storage_final{t in time: t=card(time)}:
32     storage_level[t] >= 0.5 * storage_capacity;
33
34 s.t. res_storage_capacity{t in time}:
35     storage_level[t] <= storage_capacity;

```

---

example of a linear energy system model for optimal energy storage sizing and operation. Data is excluded for brevity. The full model including an example data set is listed in appendix B. The MathProg language is bundled with the open source solver GLPK, which can be used either to solve them or to convert them to other formats. A weakness of MathProg is lack of a software ecosystem that would allow for easier input and result data handling than processing bare text files.

## YALMIP

YALMIP is a free modelling language written for MATLAB and actively developed. It allows to formulate linear, general convex and even non-convex optimisation problems using MATLAB syntax. Listing 2.3 shows a minimum example of a non-linear optimisation problem with

Listing 2.3: YALMIP example model

---

```

1 x = sdpvar(2,1);
2 constr = [sum(x) <= 1, ...
3           x(1) == 0, ...
4           x(2) >= 0.5];
5 obj = x'*x + norm(x);
6 opt = sdpsettings('verbose',1,'solver','cplex');
7 sol = optimize(constr,obj,options);
8 value(x)

```

---

two decision variables  $x_1, x_2 \in \mathbb{R}$  and three linear constraints. If one is already working in the MATLAB environment for data processing, YALMIP is an attractive option to extend the capabilities towards mathematical modelling. For the goals of this thesis, the commercial license for its host system (MATLAB) is its main disadvantage.

### Pyomo

The Python package Pyomo provides all the tools to write mathematical models in a similar fashion as in GAMS for the Python programming language. Listing A.1 in appendix A shows the general structure of the optimisation models developed in this thesis. Objects like `Set` or `Constraint` take the role of custom commands. The Python syntax leads to more verbose code, but on the other hand allows tight control over the model creation process. Also, the input/output code can be integrated more tightly. In the models presented here, Pyomo models use DataFrames directly to initialise constraints without declaring separate parameter objects.

A clear weakness of Pyomo is its performance. Due to the large number of objects (one for each scalar model element) that need to be initialised, both model initialisation, model serialisation (using pickle) and result access take longer than for example accessing a GDX file created by GAMS. This weakness perceivably lengthens the runtime for a Pyomo implementation of model **urbs**. However, for a MILP problem like **rivus** the solver runtime clearly dominates the pre- and post-processing time.

### JuMP

Only published in 2015, package JuMP [25] for the Julia programming language [9] is a new algebraic modelling language. It promises performance comparable to AMPL and a feature set comparable to Pyomo. It already offers all of the basic features and supports several solvers, both commercial and open source. One speciality of this library is a convenient application programming interface (API) for specifying MIP callbacks (lazy constraints). If one is just starting to investigate different options for a fast yet high-level language to formulate one's optimisation problem in, this package is definitely worth considering.

## 2.5.2 Data management

One often overlooked challenge in designing an optimisation tool chain is the friction in development and execution time for data conversion. Whenever time is spent on writing custom

scripts that parse CSV files and on sorting, splicing and pruning such files, that time could alternatively be used once to identify an environment that automates such conversions. Or one could spend time to swap one element in the chain and eliminate the need for data conversion altogether.

While CSV files are the greatest common denominator in data management, the lack of standardisation and dialects makes them a double-edged sword. Reading and writing performance is mediocre due to their low data density, especially for numeric data; however, compression can help with that disadvantage.

The Hierarchical Data Format (HDF) is a very attractive option to exchange large quantities of homogeneous numeric data. This might also be the reason that MATLAB has changed the backend for its `MAT` format to HDF.

For geographic data, shapefiles<sup>3</sup> are the de-facto standard exchange format. For larger datasets, geographic databases are the go-to solution despite the higher effort to setup. A popular database is the PostgreSQL DBMS with the PostGIS extension.

For small, manually curated datasets, nothing beats the ease-of-use of standard spreadsheet applications. Inline visualisation and conditional formatting make it easy to come close to the functionality of a purpose-built UI without much effort. However, the two-dimensional nature does not let them scale well to datasets with thousands of values. Those call for numeric processing environments like MATLAB, R, or packages like NumPy or Pandas. A reading recommendation for a good way to think about data processing is the split-apply-combine strategy [127] that allows to concisely describe a large class of operations.

### 2.5.3 Solvers

The availability of general purpose solvers is one of the main motivations to use mathematical modelling techniques to express one's optimisation problems. These programs can digest "any" problem that can be transformed into the standard form and return either a proven optimal solution (in the case of LP), a solution with guaranteed maximum "distance" from a dynamic lower bound for the theoretical optimum (in case of MILP), or a proof that the problem is either infeasible or unbounded. In any case, this leaves all of the modeller's time to improve one's understanding of the actual problem at hand.

#### Open source solvers

GLPK is the most well known open source solver for LP and MILP. However, due to lack of development time, its performance is bound to be several orders of magnitude [sic] slower than of commercial solvers. While this difference is practically negligible in small problems or educational use, any large scale application suffers severely. For MILP, the heuristics used in the branch & bound algorithm do not manage to find a single feasible solution after several hours.

Other widely cited solvers (that were not used in this thesis though) are COIN-OR branch and Cut (CBC) developed by the COIN-OR project, and Solving Constraint Integer Programs

---

<sup>3</sup>The name *shapefile* is misleading: it actually consists of multiple identically named files with the endings `dbf` for the attribute table, `shp` for geometries, `shx` for a geometry index and possibly more (e.g. `prj` for CRS), depending on the creating software

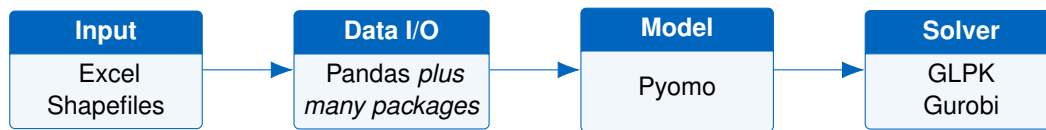


Figure 2.16: Modelling tool chain with software used in this thesis

(SCIP) by Zuse Institute Berlin (ZIB). Especially SCIP is interesting as it is a contender for the title of the fastest open source MILP solver.

### Closed source solvers

IBM ILOG CPLEX Optimization Studio (CPLEX) is the industry standard solver package for large scale optimisation problems. It features extensive preprocessing, automatically eliminating or reducing redundant constraints, multiple heuristics that can be triggered by the structure of the optimisation problem itself. CPLEX is not limited to LP/MILPs, but can be also used for quadratic or non-linear problems.

XPRESS is a contender for the title of the fastest LP solver. In one case study involving a large capacity expansion problem, it managed to find solutions within a GAMS stack when CPLEX could no longer solve it. However, such a finding is only a single data point that cannot be used to generalise the solver performance for any problem class.

Gurobi is another high-performance solver for LP, MILP and other problem classes. It claims to be faster than CPLEX especially for mixed integer problems. It is used in this thesis for solving the last case study.

### 2.5.4 Final tool chain

Having presented a multitude of options for setting up one's optimisation tool chain, this section presents the choice that this thesis relies on. The choice is summarised in figure 2.16 and justified in the following. The chain is deceptively short: A spreadsheet is used for data preparation with Pandas for processing; Pyomo is used for model creation; GLPK (or Gurobi) optimises the generated problem.

Python is used as the base language of all operations. The main reason is for this choice is that over a decade of wide adoption in many scientific disciplines have lead to a rich ecosystem of freely available packages. The official Python package index<sup>4</sup> lists over 65 000 packages. This means that to almost any topic, one can find one or even multiple pre-existing implementations of standard algorithms. Package Pandas [77] is the reason that R did not overrule the ecosystem advantage due to lack of a good data handling solution. The DataFrame structure alone can cover almost all of the data processing tasks in very short, concise code fragments.

Another reason for choosing Python is the IPython notebook. It allows for an interactive explorative coding and data analysis, whose code and results can then easily shared<sup>5</sup>. There are whole papers written in the IPython format, allowing for the analysis to be scrutinised on the basis of working code, not a proxy description.

<sup>4</sup><https://pypi.python.org/pypi>

<sup>5</sup>e.g. <http://nbviewer.ipython.org/gist/ojdo/829dc61bcf0b98bc886f/monthly-means.ipynb>

Package Pandas is used for its label-based multi-dimensional data structures with a rich portfolio of efficiently implemented high-level operations: slicing, sorting, grouping, filters, aggregating, reshaping, resampling can all be done in a vectorised way, without having to write loops. Some of *many* other packages to be mentioned are NumPy for fast, vectorised operations like in MATLAB, the SciPy stack as a repository for tools like plotting, and Pyomo already presented above in section 2.5.1. Geographic operations are based on package Shapely [33].

Many iterations have been spent trying out other options: coupling GAMS with Excel lacks power in data transformations, GAMS with MATLAB is very powerful, but not suitable for broad usage. The first modelling package tried in Python was Pymprog<sup>6</sup>. It worked, but only supports GLPK as a solver. Pyomo finally had support for multiple solvers and a cleaner API.

---

<sup>6</sup><http://pymprog.sourceforge.net/>



## 2.6 Data

This section outlines data formats and sources. They are used as parameters across the optimisation models. It briefly discusses sources for technical and economic parameters for urban energy infrastructure technologies, while section 2.4 put focus on processing time series and geographic data.

### 2.6.1 Technical

A useful resource for thermal calculations on the building scale are norms. The ASHRAE Handbook [93] for example provides many technical parameters about thermal behaviour of buildings. This ranges from ventilation, refrigeration to heating and insulation. While a full building hull calculation is beyond the scope of this thesis, typical values help to validate empirical values found by other studies. Concerning basic parameters like conversion efficiency, greenhouse gas emissions, and full load hours, text books like Oeding [89] provide a good first overview.

### 2.6.2 Economic

**Parameters** Economic parameters are basically specific cost and revenue data. Costs that serve the purpose of this thesis can be divided into investment costs, fixed costs, variable costs, and fuel costs. Investment costs occur for the build-up of capacity. In general, they are a non-linear function of the size of equipment. For the purpose of linear programming, either a linear or affine (linear with offset) relationship between equipment size is assumed. While this might seem like a strong simplification, they manage to represent the economics of scale with sufficient accuracy. Fixed costs arise for the continued operation & maintenance (O&M) of built or existing equipment. In contrast to variable costs, however, their amount does not depend on the actual operation of the equipment, like staff, insurance, and taxes. Finally, variable costs occur for operating the equipment. Any wear-and-tear to the equipment can be accounted for here. Fuel costs are a property of the consumed commodity and depend on the location (where on the globe, how easy to reach) and at which scale the consumption (single household to industrial plant) happens.

Every energy conversion, energy storage, and energy transmission process has a different set of costs. For some process types, most of these values can have value zero. Whether a given real cost is represented as included in investment, fixed or variable cost is a matter of the research question at hand, or – if small enough – even does not influence the optimal solution at all.

Any conversion process that employs moving parts, for example an internal combustion engine, has usually non-negligible variable costs for lubrication and fixed costs for maintenance of moving parts. A photovoltaic plant usually has near-zero variable costs and only fixed costs for regular maintenance and cleaning of panels.

By analogy, electric battery storage experiences ageing (that could be modelled as a variable cost term), whereas a pump hydro storage plant has a different size of variable costs for wear-and-tear of pump and turbine parts.

**Equivalent annual cost method** The EAC method [61, p.182] is used to calculate the annual costs of operating an investment good over its economic lifetime. Its value is calculated by multiplying the NPV of an investment with the *annuity factor*  $ANF(n, i)$ . This factor is a function of the economic lifetime  $n$  (in years) and the capital interest rate  $i$ :

$$ANF(n, i) = \frac{(1 + i)^n \cdot i}{(1 + i)^n - 1} \quad (2.34)$$

Not surprisingly, this factor is exactly the inverse of the factor to be used to calculate the net present value of a sequence of annual payments. The annuity factor appears as a parameter in models **URBS** and **urbs** as a convenience function, to allow to enter different investment costs, lifetimes and interest rates for different equipment, while keeping overnight investment costs in the input spreadsheet. Note that the use of the annuity method assumes a similar or at least comparable risk level of different investment options. If the risk levels of compared options are considerably different, the values could be adjusted by factoring in risk as an additional cost. For simplicity, models **DHMIN**, **dhmin**, and **rivus** do not include an annuity factor in their formulation but assume that investment costs already have been suitable scaled to an annual cost figure.

Which values for  $n$  and  $i$  are appropriate for a given case study or context is a matter of ongoing discussion. The reader is referred to an article by Gronemann and Döring [39] or the well-known Stern review [114], especially chapter 2. One remark: if the study is for a corporate setting, the value for the interest rate  $i$  to be taken is the weighted average cost of capital (WACC).

**CAPEX** In some case studies, the term capacity expenditure (CAPEX) is used. It is the sum of annualised investment costs and the annual fixed costs, thereby summarising all the capacity-dependent costs for a piece of equipment.

**OPEX** In some case studies, the term operational expenditure (OPEX) is used. Here, it is synonymous for the sum of variable costs and fuel costs for a technology, thereby summarising all the operation-dependent costs for a piece of equipment.

### 2.6.3 Time series

Models **URBS** and **urbs** take some of their input in form of time series. For the case studies in this thesis, a time resolution of 1 hour is used. The two main types of time series are for demand and supply.<sup>7</sup>

<sup>7</sup>The online version of **urbs** – thanks to a contributor – now also features time series for purchasing and selling commodities from and to outside of the modelled system. This feature is not described in this thesis, but in the online model documentation at [urbs.readthedocs.org](http://urbs.readthedocs.org).



Figure 2.17: Clusters of cholera cases in the London epidemic of 1854 [111]

**Demand** A thorough treatment of electric loads is the doctoral thesis by Hellwig [44] from 2003. It describes the development of parametrisable building load profiles for thermal energy based on sigmoid function fit to the scatter plot of daily demand over average temperature. Newer developments are summarised and extended in the doctoral thesis by Gobmaier from 2013 [35]. He presents a method for synthesising per-building electrical load profiles for a given climate region.

Heat was covered in temporal resolution by Gadd [29] recently, and even more in depth in the older thesis by Pálsson [97]. This thesis assumes that a heat demand curve summarises all non-industrial uses for heat in a single demand curve. This includes foremost hot water preparation and space heating. In other words: possible complications or synergies of different temperature levels are *ignored* here.

**Supply** For his doctoral thesis, Janker [54] has created and documented the process of converting multiple datasets published by NASA – mainly the modern-era retrospective analysis for research and applications (MERRA) dataset – into globally available, high spatial high temporal resolution datasets. The result are so-called *capacity factors* for wind power and solar power, together with multiple supplementary parameters like temperature, wind speeds and air pressure. Case study Haag uses solar and wind capacity factors derived from that thesis as raw data for use with model **urbs**. Capacity factors encode what fraction of a given installed equipment's nominal power is output under the simulated environmental conditions. All non-linearities, delays or overshoots are thus already included in the derivation of that parameter.

#### 2.6.4 Geographic information

A short historical note on the difference between cartography and spatial analysis of data shall motivate why geographic data is presented here on its own: During a cholera outbreak

Listing 2.4: A street segment encoded using OpenStreetMap XML

---

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <osm version="0.6" generator="Overpass API">
3   <node id="11216747" lat="48.1673468" lon="11.4249068"/>
4   <node id="11218477" lat="48.1752601" lon="11.4104194"/>
5   <node id="11218478" lat="48.1775925" lon="11.4061901"/>
6   <way id="99341613">
7     <nd ref="11216747"/>
8     <nd ref="11218477"/>
9     <nd ref="11218478"/>
10    <tag k="highway" v="primary_link"/>
11  </way>
12 </osm>

```

---

in London in 1854, physician John Snow employed a quantitative representation of already collected data to find the source of the outbreak. Using a dot map (Figure 2.17), he correctly identified a water pump as the source of infection. This happened at a time when the cause for cholera had still not been identified. This example shows that a simple change of representation can greatly alter the deductions one can draw from a given set of data.

Technically, geographic data can be categorised in two types: raster and vector data. Raster data includes image data and any dataset whose values are specified on a regular spatial grid. Vector data consists of collections of geometric primitives. These primitives are mainly points, lines and polygons. In this thesis, only vector data is used.

Data source for this thesis is mainly OpenStreetMap (OSM) [91], with small exceptions that require slanted aerial imagery, as provided by Google Maps [37]. A small excerpt of the raw extensible markup language (XML) data that OSM uses is shown in listing 2.4. Shown are the two main simple primitives `node` to declare point coordinates and `way` to link several nodes. Meaning is given to ways using an arbitrary way segment with a (principally unlimited) number of `tags`, whose usage is subject to a steady evolution. Not shown is the third main element, `relation`, which allows to express more complex concepts that encompass multiple nodes, ways or other relations to form semantic trees of meaning.

The main data format for storing spatial data in this thesis are shapefiles. These consist of a collection of identically named files with different endings: `file.shp` contains the geometries (points, lines or polygons) in a binary, but openly specified [26] format. `file.shx` is a binary index file that points to the starting positions of individual geometries in the first file. The third `file.dbf` is a tabular that can contain numeric and textual attributes for each geometry. As this original standard does not contain information on the coordinate reference system (CRS) used for specifying the coordinates, various standards emerged for adding this piece of information: `file.prj` or `file.qpj` (or both) may contain a plaintext string that identifies the used CRS by stating its EPSG code.

Spatial heat demand analysis was performed in the doctoral thesis by Böhme [17]. It is a comprehensive example of what can be achieved data-wise on an urban scale to estimate energy demands with enough time, dedication and data. A widely cited example is the study of Howard et al. [48], estimating spatial electricity and heat demand distribution in New York City.

# Chapter 3

## Models

*Essentially, all models are wrong, but some are useful.*

George E. P. Box (1919–2013)

This chapter describes the models that are used in this thesis. They employ the model formulations presented in section 2.2.2, where the building blocks of LP/MILP models are outlined individually. Together they form a family of related energy system optimisation models. They share a common codebase and can work closely together. First, the motivation and design goal for each model is shortly presented, followed by a complete mathematical description. Algorithms for data preprocessing are presented per case study in chapter 4. Models with uppercase names (**URBS**, **DHMIN**) are prior art and were only re-implemented during this thesis. The lowercase names (**urbs**, **dhmin**, **rivus**) are own developments with additional features and a shared code structure, as discussed in appendix A.

Table 3.1: Model comparison of **URBS**, **urbs**, **DHMIN**, **dhmin**, and **rivus** by aspect. Ratings range from neglected (empty) over low suitability (●) to high suitability (▬). ●\* means: **rivus** in principle could support these features fully, but for sake of simplicity they are not implemented at present.

Aspect	Model	<b>URBS</b> (SISO)	<b>urbs</b> (MIMO)	<b>DHMIN</b>	<b>dhmin</b> (season.)	<b>rivus</b>
Time		▬	▬	●	▬	▬
Space		▬	▬	▬	▬	▬
Transmission		▬	▬	▬	▬	▬
Processes		▬	▬			▬
Storage		▬	▬			
Capacity expansion		▬	▬	▬	▬	▬
—with exist. capacity		▬	▬	▬	▬	●*
Unit commitment		▬	▬	●	▬	▬
Redundancy					▬	●*
Fixed invest costs				▬	▬	▬

Table 3.1 gives an overview on the coverage of different aspects for each of the five models. **URBS** (SISO) and **urbs** (MIMO) focus on processes, i.e. units of energy conversion. In contrast, **DHMIN**, **dhmin** (seasonal) and **rivus** put more stress on the spatial aspect of energy systems. This is motivated by the fact that *heat* as the crucial interlinking commodity is intrinsically hard to transport efficiently over large distances. However, this high spatial resolution leads to intrinsic weaknesses in the temporal resolution in these *network models*. Therefore, they are not capable of representing challenges and benefits of temporal variations in energy demand and supply.

The second half of the table deals with certain subtasks in energy modelling. All models are capable of *capacity expansion planning*. The two district heating models **DHMIN** and **dhmin** have the intrinsic restriction of being written for a single energy carrier, and thus can only plan capacities for its transmission. If existing (i.e. installed) capacities must be respected, **rivus** falls flat for now, as no parameters for their inclusion are implemented at the moment. *Unit commitment*, at least without elaborate technical restrictions, is one of the design tasks of the **urbs** model family, including conversion, transmission and storage in arbitrary resolution. **DHMIN** does not have time at all, but **dhmin** and **rivus** allow at least for basic operations planning (base load, peak load, . . .) in representative operating points. Through this feature, **dhmin** allows for planning network capacities in case of heat generation outages. Although **rivus** does not have this feature presently, transferring the equations would allow for redundant planning.

### 3.1 URBS (SISO processes)

This section presents a linear energy system optimisation model. It only features energy conversion processes with a single input and a single output (SISO). In the context of this thesis, it serves as a development baseline to motivate the further development which lead to the following model **urbs** (MIMO processes) that is described in the subsequent section 3.2.

#### 3.1.1 Origin

In section 2.3.1, *the method URBS* from the thesis by Richter back in 2004 [101] has already been mentioned. This model is very similar, but not identical to the energy technology module that he described in his thesis. It supported multiple input and output processes, but was hard-coded in the number of transportable commodities. One important feature of the original, which is lacking from both **URBS** and **urbs**, is the notion of *proportional processes*. These are conversion technologies that must satisfy an arbitrary, but constant fraction of a demand time series. This property is crucial to accurately represent local, but distributed technologies like heat boilers or CHP units.<sup>1</sup> The model was implemented in C++, but no implementation details were given.

#### 3.1.2 Conceptual overview

**URBS** is a mathematical optimisation model for designing and operating a *distributed* energy system. It minimises total cost  $\zeta$  for providing energy in form of required demand commodities. Demand is provided as input in form of several (fixed) timeseries. **URBS** introduces multiple locations, called *sites*, as well as a new model component to allow for transport of commodities between sites, called *transmission*. Again, the solution of the optimisation program shows the cost-optimal use of resources  $\rho$ , investment in capacities  $\kappa$  and scheduling of conversion units  $\epsilon$  and transmission  $\pi$  for the available technologies to satisfy a given energy demand, while not violating a limit for greenhouse gas emissions:

<sup>1</sup>Actually, proportional processes have been implemented in the meantime in model **urbs** in branch [haag15](#) for properly handling case study Haag i. OB. Refer to section 4.4 for a description of these changes.

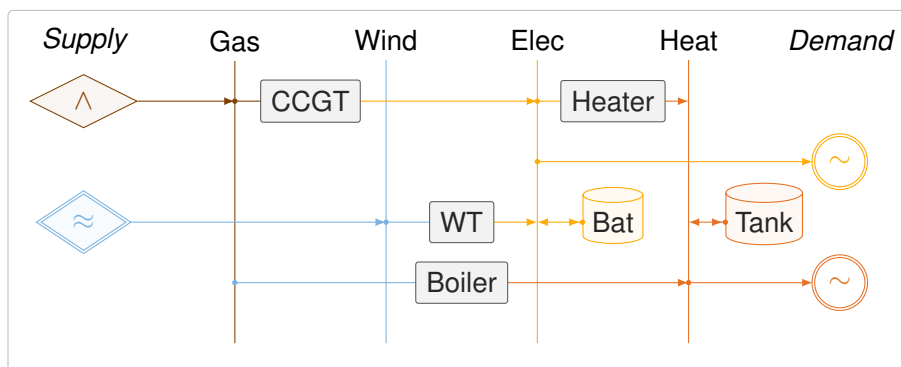


Figure 3.1: Exemplary RES diagram of one site in an **URBS** model

$$\zeta^* = \min_{\rho, \kappa, \epsilon, \pi} \zeta \quad (3.1)$$

The use of resources  $\rho = (\rho_{vct})$  is possible for source commodities at different sites for pre-determined costs, possibly limited by a maximum hourly or annual supply cap. Capacities  $\kappa = (\kappa_{vp}, \kappa_{vs}^c, \kappa_{vs}^d, \kappa_{af})$  of energy conversion ( $p$ ) and storage ( $s$ ) processes and capacities for transmission lines ( $f$ ) are located in sites ( $v$ ) or between them ( $a$ ). For each time step, values for state variables in sites  $\epsilon = (\epsilon_{vpt}^{\text{in}}, \epsilon_{vst}^{\text{con}})$  and between sites  $\pi = (\pi_{aft}^{\text{in}}, \pi_{aft}^{\text{out}})$  are optimised. This does not happen iteratively, but globally with perfect foresight to yield the best conceivable operation strategy (and size) for each considered technology.

Figure 3.1 shows the model structure within one such site. It represents an instance of a RES, a graph representation of energy supply chains. Note that the commodities and the connected conversion processes and storage technologies not necessarily correspond to existing infrastructure. They merely represent a technology in form of their main technical (efficiency) and economic (costs) properties. Whether and how much a given technology is then actually installed at a given site, is highly dependent on the availability of *supply*, the type and shape of *demand* and how cheap or expensive the transmission of commodities can happen from neighbouring sites.

### 3.1.3 Example model result

Figures 3.2 and 3.3 show an exemplary result of a small model run with 3 countries called North, Mid, and South. Optimisation covers a period of five summer days in the Northern hemisphere. Figure 3.2 shows the base scenario, in which Mid meets its demand by burning mainly gas and lignite, while South and North can mainly rely on renewable sources. Figure 3.3 shows a different scenario, in which a strict 95% CO<sub>2</sub> emission reduction constraint is introduced. Additionally, availability of hydro and biomass in North is reduced to 50% and 25%. While there is no energy transmission in the base scenario, Mid relies heavily on power transmission from South in the second scenario.

Several model features are on display in this simple example. Violet areas show use of storage at various sizes. Site Mid for example relies on much bigger storage capacities to cover its load in the cooperative scenario. Site South converts from a self-sufficient solar electricity production to a huge exporter, using large storage capacities to deliver base load export electricity. Site North, already having huge natural storage capacities, has least trouble to cover its load from fluctuating renewable sources, at least during the five days shown.

### 3.1.4 Mathematical description

In the following, all model sets, parameters, variables, the objective and all constraints are introduced and briefly discussed.

#### Sets

All sets are summarised in table 3.2. The ordered set of *time steps*  $T = \{t_0, \dots, t_N\}$  contains a sequence of successive points in time, sampled at identical time intervals. The subset of



Table 3.2: URBS model sets

Set	Subsets	Description
$t \in T$	$T_m, t_0, t_N$	Time steps, modelled time steps, initial/final time step
$v \in V$	—	Sites
$a \in A$	$A_v^p, A_v^s$	Arcs, incoming/outgoing arcs in site $v$
$c \in C$	$C_{st}, C_{sup}, \dots$	Commodities, commodity type subsets
$p \in P$	$P_{vc}^{in}, P_{vc}^{out}$	Processes, consuming/producing commodity $c$
$s \in S$	$S_{vc}$	Storage, storing commodity $c$
$f \in F$	$F_{vc}^{exp}, F_{vc}^{imp}$	Transmission, exporting/importing transmission

modelled time steps  $T_m = \{t_1, \dots, t_N\}$ , which omits the initial time step  $t_0$ , is introduced for shorter definition of the storage state equation.

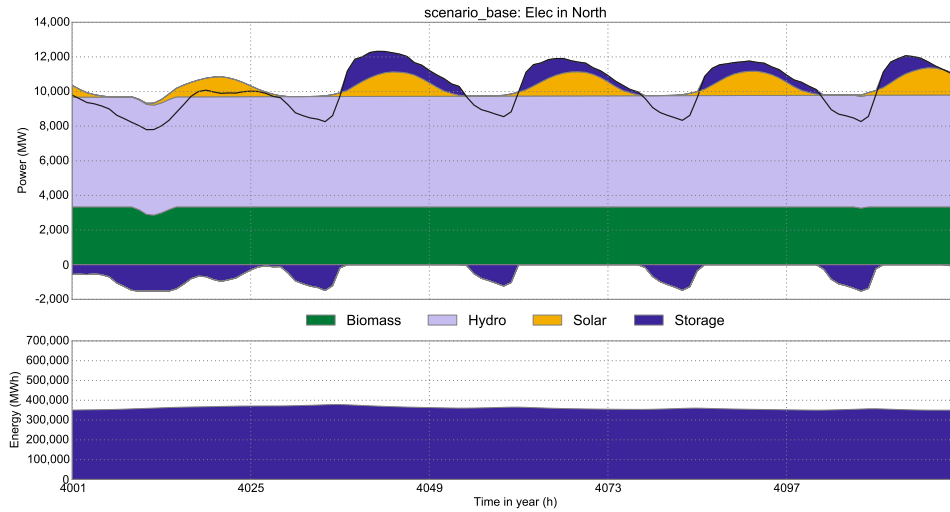
The set of sites  $v \in V$  is a list of distinct locations, possibly regions or countries. Conceptually, they represent nodes in a graph of locations, connected by edges that are defined by possible directed *transmission* lines (or *arcs*)  $a \in A \subseteq V^2$  between them. For each site  $v$ , the sets  $A_v^p$  and  $A_v^s$  contain all incoming (*predecessor*) and outgoing (*successor*) arcs.

The set of commodities  $c \in C$  is a list of resources available as supply, demand, or other goods that can be converted into other commodities. Each commodity is also member of exactly one of the four subsets  $C_{st}, C_{sup}, C_{dem}$  and  $C_{env}$ , given by its commodity type. Each of the commodity types has constraints (and thus parameters) only applicable to them. Each site  $v$  can have different subset of  $C_v$  available in it. For example, the set of stock commodities available in site  $v$  can be found by calculating  $C_v \cap C_{st}$ . *Stock commodities* are buyable at any time for a given price. Supply can be limited per time step or for a whole year. Examples are coal, gas, uranium or biomass. *Supply intermittent (supim)* are fluctuating resources like solar radiation and wind energy, which are available according to a time series of values, which could be derived from weather data. *Demand commodities* have a time series for the requirement associated and must be provided by output from other process or from storage. *Environmental commodities* are represented by the hard-coded commodity  $CO_2$  and represents the amount (in tons) of greenhouse gas emissions from processes. Its total amount can be limited by parameters to investigate the effect of emission reduction goals on the optimal system configuration.

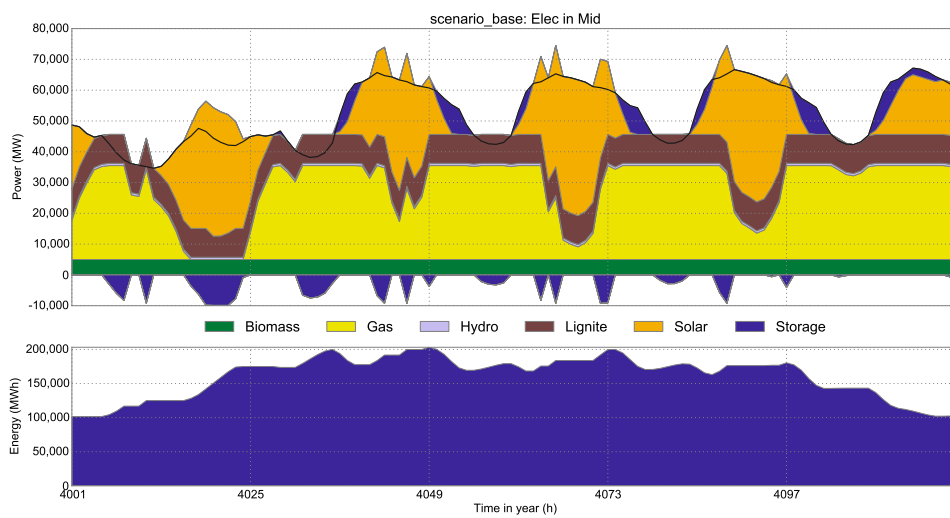
Processes  $p \in P$  represent energy conversion from one input commodity to an output commodity. For each combination  $(v, c)$  of site  $v$  and commodity  $c$ , the sets  $P_{vc}^{in}$  (and  $P_{vc}^{out}$ ) contain all processes that take commodity  $c$  as input (or output) at site  $v$ .

Storage processes  $s \in S$  can store one commodity at one time step and release it (with losses) at a later time step. For each combination  $(v, c)$  of site  $v$  and commodity  $c$ , the set  $S_{vc}$  comprises the available storage technologies.

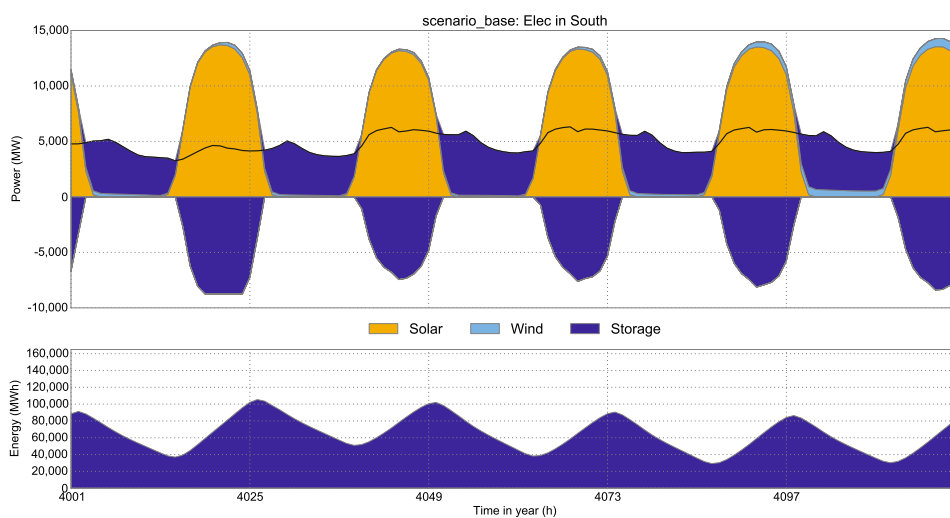
Transmission lines  $f \in F$  (think *powerflow*) are defined between ordered pairs of sites  $(v_{in}, v_{out}) \in V^2$  and are able to transport a certain commodity. For each combination  $(v, c)$  of site  $v$  and commodity  $c$ , the sets  $F_{vc}^{imp}$  and  $F_{vc}^{exp}$  contain the transmission processes that allow to import or export the given commodity.



(a) North: hydropower and biomass

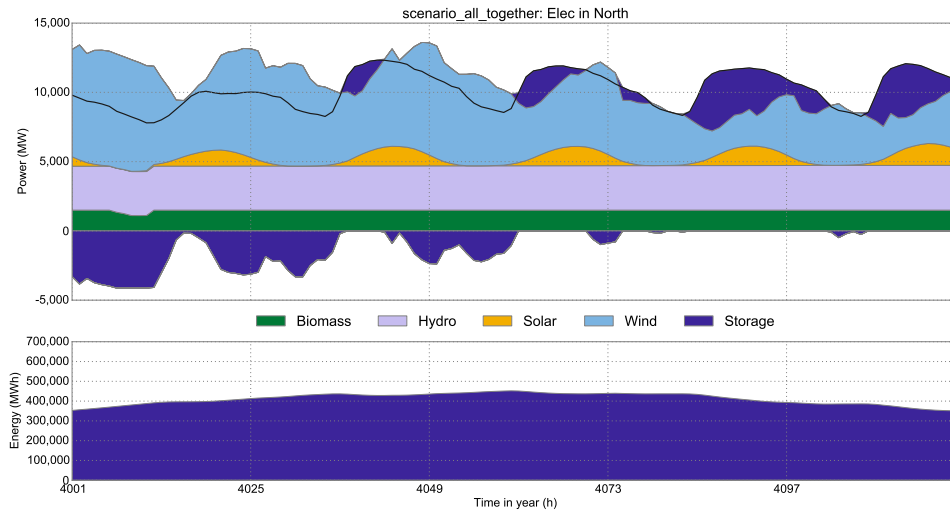


(b) Mid: gas, lignite and solar

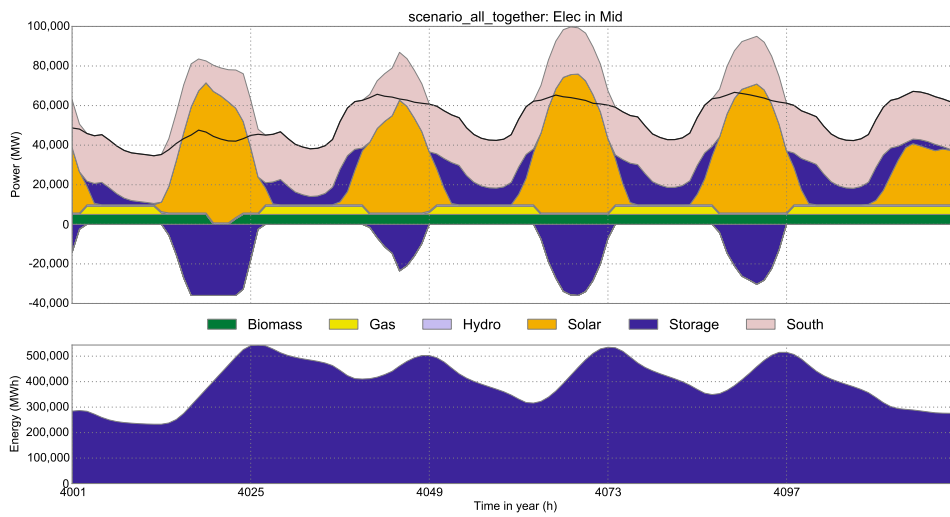


(c) South: solar only, storage for night

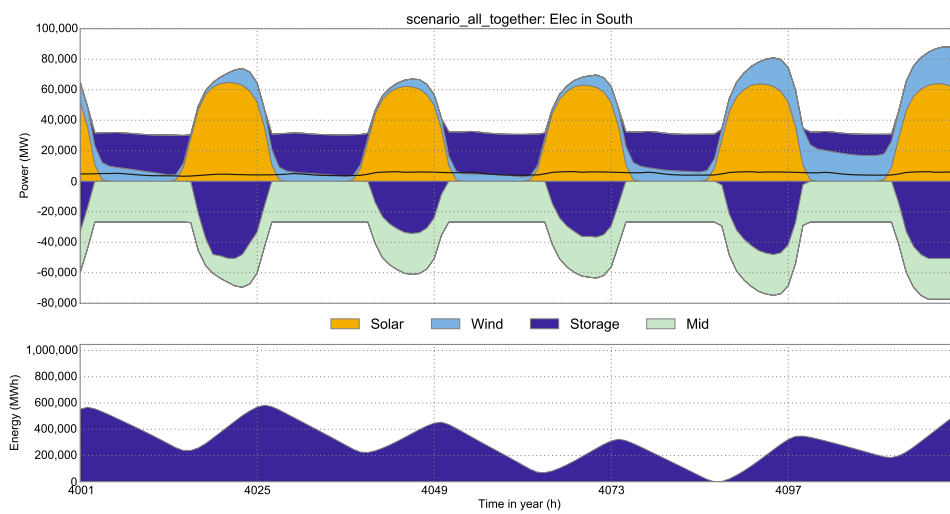
Figure 3.2: Base scenario, local generation satisfies load



(a) North: wind and less hydro and biomass (due to constraint)



(b) Mid: solar and import from South



(c) South: solar and storage for export to Mid

Figure 3.3: Cooperative scenario, reduced CO<sub>2</sub> limit leads to transmission

Table 3.3: URBS technical model parameters

Name	Unit	Description
$\bar{G}$	t	maximum allowed CO <sub>2</sub> emissions
$w$	h	time step weight
$d_{vct}$	kWh/h	demand per time step
$s_{vct}$	1	intermittent supply capacity factor
$\bar{I}_{vc}$	kWh/h	maximum stock supply per time step
$\bar{L}_{vc}$	kWh	maximum total stock supply
$e_{vp}$	1	process conversion efficiency
$g_{vp}$	t/kWh	process CO <sub>2</sub> emissions per input
$\underline{K}_{vp}, K_{vp}, \bar{K}_{vp}$	kW	process capacity (lower bound, installed, upper bound)
$I_{vs}$	1	initial and final storage content (relative)
$e_{vs}^{in}, e_{vs}^{out}$	1	storage efficiency (during charge, discharge)
$\underline{K}_{vs}^c, K_{vs}^c, \bar{K}_{vs}^c$	kWh	storage content (lower bound, installed, upper bound)
$\underline{K}_{vs}^p, K_{vs}^p, \bar{K}_{vs}^p$	kW	storage power (lower bound, installed, upper bound)
$e_{af}$	1	transmission efficiency
$\underline{K}_{af}, K_{af}, \bar{K}_{af}$	kW	transmission capacity (lower bound, installed, upper bound)

### Parameters

Parameters are all pre-defined numerical values in the optimisation problem, i.e. their values determine – by *parametrising* the following constraint equations – the location of the desired optimum. There are two types of parameters: technical and economic.

**Technical parameters** Table 3.3 lists all technical parameters. They start with the greenhouse gas emission limit  $\bar{G}$ , which is available only as a global constant. That means, only a single emission goal can be set for the whole distributed system<sup>2</sup>. The time step weight  $w$  is used to scale the variable costs, that occur during the simulation time frame (typically some days or weeks) to a full year.

**Economic parameters** Table 3.4 summarises all economic model parameters. Not surprisingly, they are all costs of different system components. They are grouped by system components, i.e. commodities, processes, storage and transmission. For commodities, only fuel costs are considered. For processes, investment, fixed and variable costs are discerned. Similarly, storage and transmission are handled. For storage, a separate set of costs parameters for storage capacity (in MWh) and storage power (in MW) are given. If one of these two cost types is dominant for a certain technology, the other can be set to (near) zero, while limiting the corresponding capacity upper bound to a safe (positive) value. Investment costs are to be discounted to annual payments; the suggested method for this is multiplying the overnight costs with an annuity factor as described in section 2.6.2.

<sup>2</sup>If required, per-site goals could be implemented easily by limiting greenhouse gas emissions  $\sum_{t,p} \gamma_{vpt}$  per site. Note that this features is implemented in **urbs**

Table 3.4: URBS economic model parameters

Name	Unit	Description
$k_{vc}^{\text{fuel}}$	€/MWh	stock commodity fuel costs
$k_{vp}^{\text{inv}}$	€/(MW a)	annualised process capacity investment
$k_{vp}^{\text{fix}}$	€/(MW a)	process capacity fixed costs
$k_{vp}^{\text{var}}$	€/MWh	process variable costs
$k_{vs}^{\text{p,inv}}$	€/(MW a)	annualised storage power investment
$k_{vs}^{\text{p,fix}}$	€/(MW a)	annual storage power fixed costs
$k_{vs}^{\text{p,var}}$	€/MWh	storage power variable costs
$k_{vs}^{\text{c,inv}}$	€/(MWh a)	annualised storage size investment
$k_{vs}^{\text{c,fix}}$	€/(MWh a)	annual storage size fixed costs
$k_{vs}^{\text{c,var}}$	€/MWh	storage usage variable costs
$k_{af}^{\text{inv}}$	€/(MW a)	annualised transmission capacity investment
$k_{af}^{\text{fix}}$	€/(MW a)	annual transmission capacity fixed costs
$k_{af}^{\text{var}}$	€/MWh	transmission usage variable costs

## Variables

All model variables are listed in table 3.5. The system costs  $\zeta$  is the quantity to be minimised. It is the sum of investment costs  $\zeta_{\text{inv}}$  for erecting new capacities for energy conversion, transmission or storage. Next are fixed costs  $\zeta_{\text{fix}}$  for maintaining both existing and new capacities, independent from operation. Then there are operation dependent costs  $\zeta_{\text{var}}$  for all non-fuel costs that occur for operation. Finally there is the fuel cost term  $\zeta_{\text{fuel}}$  for procurement of energy in form of stock commodities.

The variable for representing stock commodity procurement is  $\rho_{vct}$ . Together with the intermittent supply, they are the only source of energy to any model.

Next are process variables. Each process has a pair of input  $\epsilon_{vpt}^{\text{in}}$  and output  $\epsilon_{vpt}^{\text{out}}$  power flows. Its greenhouse gas emissions are captured in variable  $\gamma_{vpt}$ . The output power flow is limited by the total process capacity  $\kappa_{vp}$ . The newly constructed process capacity is stored in  $\hat{\kappa}_{vp}$  for calculating the associated investment costs.

Storage processes have a similar set of variables. First is a pair of input  $\epsilon_{vst}^{\text{in}}$  and output  $\epsilon_{vst}^{\text{out}}$  power flows into and out of the storage. The current storage content is represented by variable  $\epsilon_{vst}^{\text{con}}$ . Capacity-wise, the model makes a distinction between storage content  $\kappa_{vs}^{\text{c}}$  and storage power  $\kappa_{vs}^{\text{p}}$ . Both variables have their associated newly built counterpart  $\hat{\kappa}_{vs}^{\text{c}}$  and  $\hat{\kappa}_{vs}^{\text{p}}$ .

Transmission processes again have a pair of ingoing  $\pi_{aft}^{\text{in}}$  and outgoing  $\pi_{aft}^{\text{out}}$  power flows for each directed arc  $a$ , transmission technology  $f$  and time step  $t$ . Transmission capacity is stored in variable  $\kappa_{af}$ , newly built capacity in  $\hat{\kappa}_{af}$ .

## Equations

**Cost function** The objective is to minimise the value of cost variable  $\zeta$ . This section gives the definition of its four parts *investment costs*, *fixed costs*, *variables costs* and *fuel costs*. Note

Table 3.5: URBS model variables

Name	Unit	Description
$\zeta$	€	Total system cost (Inv, Fix, Var, Fuel)
$\rho_{vct}$	MWh/h	stock commodity source term
$\epsilon_{vpt}^{in}, \epsilon_{vpt}^{out}$	MWh/h	process power flow (input, output)
$\gamma_{vpt}$	t/h	process greenhouse gas emissions
$\kappa_{vp}, \hat{\kappa}_{vp}$	MW	process capacity (total, new)
$\epsilon_{vst}^{in}, \epsilon_{vst}^{out}$	MWh/h	storage power flow (in=charge, out=discharge)
$\epsilon_{vst}^{con}$	MWh	storage energy content
$\kappa_{vs}^c, \hat{\kappa}_{vs}^c$	MWh	storage content (total, new)
$\kappa_{vs}^p, \hat{\kappa}_{vs}^p$	MW	storage power (total, new)
$\pi_{aft}^{in}, \pi_{aft}^{out}$	MWh/h	transmission power flow (input, output)
$\kappa_{af}, \hat{\kappa}_{af}$	MW	transmission capacity (total, new)

the absence of environmental costs, as environmental concerns are modelled here as an optional limit on total greenhouse gas emissions (parameter  $\bar{G}$ ).

The investment costs  $\zeta_{inv}$  sum the already annualised investment costs for processes, storage (both storage power and storage size) and transmission for all newly constructed ( $\hat{\kappa}$ ) capacities. Similarly, fixed costs  $\zeta_{fix}$  are calculated, but they are incurred for the total ( $\kappa$ ) capacities per process, storage and connection.

Variable costs  $\zeta_{var}$  can occur for all process activities. They are multiplied with the time step weight parameter  $w$  to scale them up to a duration of one year, independent of the simulation duration  $|T_m| \cdot 1$  h. Process costs occur per unit of output  $\epsilon_{vpt}^{out}$ . Transmission costs are due per unit of inflowing power  $\pi_{aft}^{in}$ . Storage costs can occur both for maintaining a certain storage content  $k_{vs}^{c,var}$  as well as for charging and discharging  $k_{vs}^{p,var}$  the storage, depending on the modelled technology. Fuel costs finally sum the source power flow  $\rho_{vct}$  for purchased stock commodities. Like variable costs, they are scaled up to a duration of one year with the weight parameter  $w$ .

$$\zeta = \zeta_{inv} + \zeta_{fix} + \zeta_{var} + \zeta_{fuel} \quad (3.2)$$

$$\zeta_{inv} = \sum_{\substack{v \in V \\ p \in P}} \hat{\kappa}_{vp} k_{vp}^{inv} + \sum_{\substack{v \in V \\ s \in S}} \left( \hat{\kappa}_{vs}^c k_{vs}^{c,inv} + \hat{\kappa}_{vs}^p k_{vs}^{p,inv} \right) + \sum_{\substack{a \in A \\ f \in F}} \hat{\kappa}_{af} k_{af}^{inv} \quad (3.3)$$

$$\zeta_{fix} = \sum_{\substack{v \in V \\ p \in P}} \kappa_{vp} k_{vp}^{fix} + \sum_{\substack{v \in V \\ s \in S}} \left( \kappa_{vs}^c k_{vs}^{c,fix} + \kappa_{vs}^p k_{vs}^{p,fix} \right) + \sum_{\substack{a \in A \\ f \in F}} \kappa_{af} k_{af}^{fix} \quad (3.4)$$

$$\zeta_{\text{var}} = w \sum_{t \in T_m} \left( \sum_{\substack{v \in V \\ p \in P}} \epsilon_{vpt}^{\text{out}} k_{vp}^{\text{var}} + \sum_{\substack{a \in A \\ f \in F}} \pi_{af}^{\text{in}} k_{af}^{\text{var}} + \sum_{\substack{v \in V \\ s \in S}} \left[ \epsilon_{vst}^{\text{con}} k_{vs}^{\text{c,var}} + (\epsilon_{vst}^{\text{in}} + \epsilon_{vst}^{\text{out}}) k_{vs}^{\text{p,var}} \right] \right) \quad (3.5)$$

$$\zeta_{\text{fuel}} = w \sum_{t \in T_m} \sum_{v \in V} \sum_{c \in C_{\text{st}}} \rho_{vct} k_{vc}^{\text{fuel}} \quad (3.6)$$

**Constraints** The border of the feasible solution space is formed by the following sequence of equations and inequalities. Each numbered equation describes a list of constraints, defined for each element of the set(s) it is defined over.

To shorten the definition of several constraints, the following helper function CB (commodity balance) is introduced. It is defined for each time step  $t$ , location  $v$  and commodity  $c$ . Its value is equal to the net balance of all generated (positive) and consumed (negative) energy content for that time step, location and commodity. If its value is positive, energy in form of this commodity is generated, while a negative value indicates the net consumption of a commodity. A positive balance is required if  $c$  is a demand commodity and the demand parameter  $d_{vct}$  is non-zero, as stated in equation (3.8). A negative balance may occur if  $c$  is a stock commodity and supplied by the source term  $\rho_{vct}$  in equation (3.9). The commodity balance  $\text{CB}(v, c, t)$  is defined as:

$$\text{CB}(v, c, t) = \sum_{p \in P_{vc}^{\text{out}}} \epsilon_{vpt}^{\text{out}} - \sum_{p \in P_{vc}^{\text{in}}} \epsilon_{vpt}^{\text{in}} + \sum_{s \in S_{vc}} (\epsilon_{vst}^{\text{out}} - \epsilon_{vst}^{\text{in}}) + \sum_{\substack{a \in A_v^{\text{p}} \\ f \in F_{vc}^{\text{imp}}}} \pi_{aft}^{\text{out}} - \sum_{\substack{a \in A_v^{\text{s}} \\ f \in F_{vc}^{\text{exp}}}} \pi_{aft}^{\text{in}} \quad (3.7)$$

*Demand satisfaction* is the main commodity constraint. It requires that the commodity balance is positive, i.e. that there are processes that generate it in sufficient quantities.

$$\forall v \in V, c \in C_{\text{dem}}, t \in T: \quad d_{vct} \leq \text{CB}(v, c, t) \quad (3.8)$$

*Stock restrictions* limit the amount of a stock commodity that can be introduced (“bought”) into the model per time step or per whole simulation run. The first equation identifies the amount of introduced energy as the source term  $\rho_{vct}$ . The second line then enforces the hourly limit  $\bar{l}_{vc}$ , the third line the annual limit on stock commodity use  $\bar{L}_{vc}$ :

$$\forall v \in V, c \in C_{\text{st}}, t \in T: \quad \rho_{vct} = -\text{CB}(v, c, t) \quad (3.9)$$

$$\forall v \in V, c \in C_{\text{st}}, t \in T: \quad \rho_{vct} \leq \bar{l}_{vc} \quad (3.10)$$

$$\forall v \in V, c \in C_{\text{st}}: \quad w \sum_{t \in T} \rho_{vct} \leq \bar{L}_{vc} \quad (3.11)$$

*Process capacity* calculates the total capacity of a process by the sum of already installed capacity (parameter) and newly built capacity (variable). Figure 3.4 shows this relationship,

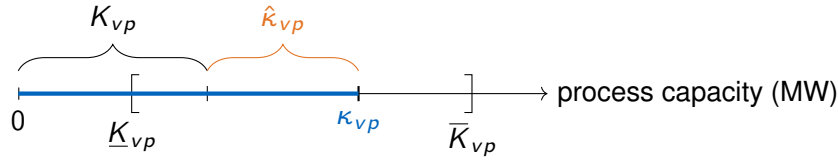


Figure 3.4: Process capacity constraints: parameters are the installed capacity  $K_{vp}$  and the lower/upper bounds  $\underline{K}_{vp}/\bar{K}_{vp}$ ; variables new capacity  $\hat{\kappa}_{vp}$  and total capacity  $\kappa_{vp}$  must assume values, so that the total capacity lies within the range defined by the bounds.

together with the capacity limits enforced by equation (3.15). Note that the installed capacity may be lower than the lower bound, which could be used to model fixed (planned) capacity expansion targets, whose costs should occur within the model:

$$\forall v \in V, p \in P: \quad \kappa_{vp} = K_{vp} + \hat{\kappa}_{vp} \quad (3.12)$$

*Process output* connects input and output of each process with the process efficiency parameter  $e_{vp}$ , which is to be set to the ratio of process output power flow to input power flow.

$$\forall v \in V, p \in P, t \in T: \quad \epsilon_{vpt}^{\text{out}} = \epsilon_{vpt}^{\text{in}} e_{vp} \quad (3.13)$$

*Intermittent supply* defines the input of a volatile process by the time series value of its input commodity  $s_{vct}$  multiplied by the total installed process capacity  $\kappa_{vp}$ . This implies that the production scales linearly in the installed capacity. In consequence, any non-linearities in the process' behaviour (e.g. a wind turbine characteristic) must already be factored in when preparing the time series  $s_{vct}$ :

$$\forall v \in V, c \in C_{\text{sup}}, p \in P_c^{\text{in}}, t \in T: \quad \epsilon_{vpt}^{\text{in}} = \kappa_{vp} s_{vct} \quad (3.14)$$

*Process capacity limit* enforces that the resulting total process capacity lies between the specified bounds, while *process output limit* limits the process output per time step to the installed capacity:

$$\forall v \in V, p \in P: \quad \underline{K}_{vp} \leq \kappa_{vp} \leq \bar{K}_{vp} \quad (3.15)$$

$$\forall v \in V, p \in P, t \in T: \quad \epsilon_{vpt}^{\text{out}} \leq \kappa_{vp} \quad (3.16)$$

*Greenhouse gas emissions* determines the amount of emissions  $\gamma_{vpt}$  using the process emission parameter  $g_{vp}$ , depending on the process activity  $\epsilon_{vpt}^{\text{in}}$ :

$$\forall v \in V, p \in P, t \in T: \quad \gamma_{vpt} = w g_{vp} \epsilon_{vpt}^{\text{in}} \quad (3.17)$$

The greenhouse gas emission equation concludes the process equations.

*Storage state* is the main storage constraint. It calculates the current storage energy content (for batteries: the state of charge (SOC)) as the previous state plus ingoing minus outgoing energy. Both during storing and retrieving energy, energy losses can be modelled



by two separate efficiencies for ingoing  $e_{vs}^{\text{in}}$  and outgoing  $e_{vs}^{\text{out}}$  power flow. Note that ingoing efficiency is a multiplier, while outgoing is a divisor in the formula:

$$\forall t \in T_m, \forall v \in V, \forall s \in S: \quad \epsilon_{vst}^{\text{con}} = \epsilon_{vst(t-1)}^{\text{con}} + \epsilon_{vst}^{\text{in}} \cdot e_{vs}^{\text{in}} - \epsilon_{vst}^{\text{out}} / e_{vs}^{\text{out}} \quad (3.18)$$

The two *storage capacity* equations calculate the total capacities for storage content ( $\kappa_{vs}^{\text{c}}$ ) and storage power ( $\kappa_{vs}^{\text{p}}$ ) as the sum of already installed capacity (parameter) and newly built capacity (variable):

$$\forall v \in V, s \in S: \quad \kappa_{vs}^{\text{c}} = K_{vs}^{\text{c}} + \hat{\kappa}_{vs}^{\text{c}} \quad (3.19)$$

$$\forall v \in V, s \in S: \quad \kappa_{vs}^{\text{p}} = K_{vs}^{\text{p}} + \hat{\kappa}_{vs}^{\text{p}} \quad (3.20)$$

*Storage capacity limits* are two constraints that restrict the two storage capacity variables to lie within the specified lower and upper bounds:

$$\forall v \in V, s \in S: \quad \underline{K}_{vs}^{\text{c}} \leq \kappa_{vs}^{\text{c}} \leq \overline{K}_{vs}^{\text{c}} \quad (3.21)$$

$$\forall v \in V, s \in S: \quad \underline{K}_{vs}^{\text{p}} \leq \kappa_{vs}^{\text{p}} \leq \overline{K}_{vs}^{\text{p}} \quad (3.22)$$

*Storage content limit* limits the storage content by the corresponding capacity variable. The same applies to storage input and output, and its capacity variable:

$$\forall v \in V, s \in S, t \in T: \quad \epsilon_{vst}^{\text{con}} \leq \kappa_{vs}^{\text{c}} \quad (3.23)$$

$$\forall v \in V, s \in S, t \in T: \quad \epsilon_{vst}^{\text{in}}, \epsilon_{vst}^{\text{out}} \leq \kappa_{vs}^{\text{p}} \quad (3.24)$$

*Initial and final storage state* define and restrict the storage state at the beginning and end of the simulation time span to a quantity relative to the total capacity:

$$\forall v \in V, s \in S: \quad \epsilon_{vst_0}^{\text{con}} = \kappa_{vs}^{\text{c}} I_{vs} \quad (3.25)$$

$$\forall v \in V, s \in S: \quad \epsilon_{vst_N}^{\text{con}} \geq \kappa_{vs}^{\text{c}} I_{vs} \quad (3.26)$$

This formulation for the initial and final storage state constraint is less general than requesting the storage to be at least as full in the end  $t_N$  as in the beginning  $t_0$  by stating  $\epsilon_{vst_0} \leq \epsilon_{vst_N}$ . However, this loss of generality greatly decreases the required computation time, especially for large problem sizes. If there is no prior knowledge about seasonality (and thus a sensible initial storage state),  $I_s = 0.5$  has been used with good success. If more than about  $10^3$  time steps are simulated, the influence of this parameter on the overall solution is negligible, while the computational performance benefit grows significantly.

*Transmission output* is the main defining transmission equation, which states that the output of a transmission process is equal to its input, reduced by the efficiency:

$$\forall a \in A, f \in F, t \in T: \quad \pi_{aft}^{\text{out}} = e_{af} \pi_{aft}^{\text{in}} \quad (3.27)$$

*Transmission capacity* defines the total transmission capacity  $\kappa$  as the sum of already existing capacity  $K$  plus newly installed capacity  $\hat{\kappa}$ , for which investment costs have to be paid:

$$\forall a \in A, f \in F: \quad \kappa_{af} = K_{af} + \hat{\kappa}_{af} \quad (3.28)$$

*Transmission input by capacity* limits the incoming power flow by the transmission line capacity:

$$\forall a \in A, f \in F, t \in T: \quad \pi_{aft}^{\text{in}} \leq \kappa_{af} \quad (3.29)$$

*Transmission capacity* limits the transmission line capacity by the lower and upper bounds:

$$\forall a \in A, f \in F: \quad \underline{K}_{af} \leq \kappa_{af} \leq \bar{K}_{af} \quad (3.30)$$

*Transmission symmetry* enforces that capacities in the two (directed) transmission lines  $a, a'$  of one (undirected) edge in the graph have identical capacities. If needed, this constraint could be conditioned on a binary attribute `is-symmetric` to disable it for one-way transmission lines, e.g. gravitational pipelines.

$$\forall a \in A, f \in F: \quad \kappa_{af} = \kappa_{a'f} \quad (3.31)$$

The constraints are concluded by the final equation *emissions*, limiting the sum of all greenhouse gas emissions to the specified maximum:

$$\bar{G} \geq \sum_{v \in V} \sum_{p \in P} \sum_{t \in T} \gamma_{vpt} \quad (3.32)$$

Equations (3.2) to (3.32) together define the linear program, whose optimal solution has the desired minimum costs  $\zeta^*$ .

## 3.2 *urbs* (MIMO processes)

### 3.2.1 Motivation for development

The model **URBS** presented in section 3.1 was limited to a single input and a single output commodity per process. The secondary output of the CO<sub>2</sub> was implemented using the special parameter  $g_{vp}$  and equation (3.17). While these restrictions were not a limiting factor in studies that focus on a single energy carrier for end use, they do not allow for representing systems with relevant processes that consume or produce more than one modelled commodity.

As outlined earlier, coupled generation of multiple output commodities, especially electricity and heat in cogeneration processes becoming more and more relevant. Extending the previously defined SISO process formulation would either require the definition of additional coupling constraints for a set of multiple processes, or recreating the formulation for conversion processes from the ground up. This consideration lead to the development of this model.

### 3.2.2 Design goal

**urbs** is a generalisation of the preceding **URBS** model. The main new feature here are processes with an arbitrary number of input/output commodities per process. This feature allows to drop the special handling of emissions as a hard-coded process attribute. Instead, CO<sub>2</sub> emissions are just an additional process output. This change is graphically shown in figure 3.5. Mathematically, this change requires the introduction of an additional state variable for the process state, called throughput  $\tau_{vpt}$ . This additional variable per process then allows to define multiple inputs  $\epsilon_{vcpt}^{in}$  and multiple outputs  $\epsilon_{vcpt}^{out}$ , which are connected to the throughput variable. The relative amounts of how much of each commodity a process generates or consumes is defined by a set of constant input and output ratios. They are the generalisation of the efficiency factor  $e_{vp}$  in **URBS**.

Figure 3.6 shows an exemplary RES diagram with two multiple input multiple output processes: *CHP* and *Boiler*. One can also see that greenhouse gas emissions can be included directly into the list of process outputs, instead of having to resort to a hard-coded variable like before. This opens up a whole family of possible model features with no or only very little implementation changes.

### 3.2.3 Example model result

Figures 3.8 and 3.9 shows an exemplary result of a model run on a case study with four imaginary islands with varying availability of renewable electricity generation potentials. Figure 3.7 shows their relative location. Figure 3.8 shows the resulting electricity production on *one of the*

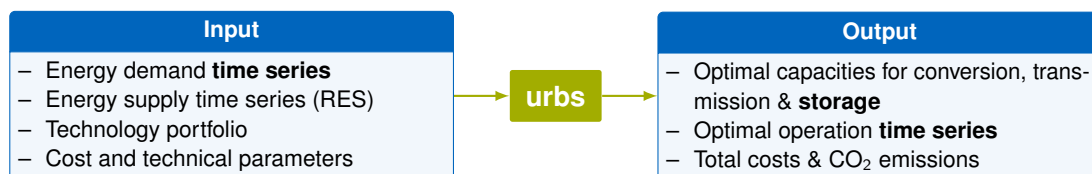


Figure 3.5: Input/output flow chart of model **urbs**

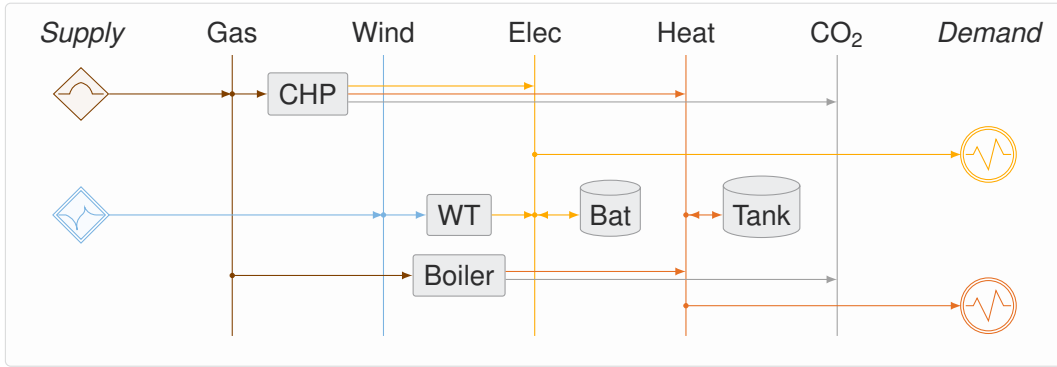


Figure 3.6: Exemplary RES diagram of one site in an **urbs** model

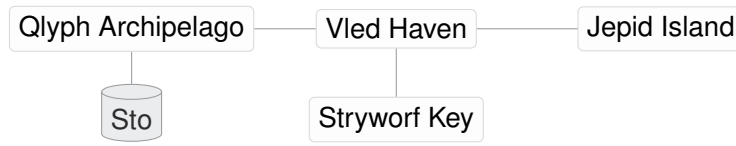


Figure 3.7: Example model site locations, with storage position

four islands in case of a base scenario with investment and variable costs similar to today's situation in European countries. The plots show four representative weeks from each of the seasons. Figure 3.9 shows the changed result with a strict limit on CO<sub>2</sub> emissions. Qlyph Archipelago, with its storage capacities (not shown), becomes the main backbone for the main island's power supply.

### 3.2.4 Mathematical description

This section contains the mathematical formulation of optimisation model **urbs**. Its objective is to minimise total costs  $\zeta$  to satisfy a given energy demand  $d_{vct}$  for a set of commodities, used in this thesis for – but not limited to – electricity and heat. Like **URBS**, the optimisation includes use of resources  $\rho$ , investment in capacities  $\kappa$  and scheduling of conversion units  $\epsilon$  and transmission  $\pi$  for the available technologies:

$$\zeta^* = \min_{\rho, \kappa, \epsilon, \pi} \zeta \quad (3.33)$$

#### Sets

Table 3.6 summarises all model sets. Time is represented by an ordered set of *time steps*  $T = \{t_0, \dots, t_N\}$  that contain a sequence of equally spaced instances. The initial time step  $t_0$  is dropped for the definition of the subset *modelled time steps*  $T_m = \{t_1, \dots, t_N\}$ . This set facilitates the definition of the storage state equation (3.57).

Space is represented by a set of *vertices*  $v \in V$ . They represent distinct locations from individual buildings, regions, countries or continents. They are nodes on an imagined graph of locations, connected by edges. These (undirected) edges are represented by a pair of (directed) *arcs*  $a \in A \subseteq V^2$ . For each vertex  $v$ , the sets  $A_v^p$  and  $A_v^s$  contain the incoming (*predecessor* in graph theory) and outgoing (*successor*) arcs.

Table 3.6: **urbs** model sets

Set	Subsets	Description
$t \in T$	$T_m, t_0, t_N$	Time steps, modelled time steps, initial/final time step
$v \in V$	—	Sites
$a \in A$	$A_v^p, A_v^s$	Arcs, incoming/outgoing arcs in site $v$
$c \in C$	$C_{st}, C_{sup}, C_{env}, C_{dem}$	Commodities, commodity type subsets
$p \in P$	$C_{vp}^{in}, C_{vp}^{out}$	Processes and their input/output commodities
$s \in S$	$S_{vc}$	Storage, storing commodity $c$
$f \in F$	$F_{vc}^{exp}, F_{vc}^{imp}$	Transmission, exporting/importing transmission

*Commodities*  $c \in C$  are all resources relevant to the modelled energy system. These include all energy carriers as fuels, intermediate substances as well as end products. Emissions in form of green house gases can be included as a commodity as well. Each commodity is member of exactly one of the four mutually exclusive subsets  $C_{st}$  for stock,  $C_{sup}$  for intermittent supply,  $C_{env}$  for environmental, and  $C_{dem}$  for demand commodities. This membership triggers the creation of different constraints. Those conditions are given both in the mathematical definition and the textual description of the affected constraints.

*Processes*  $p \in P$  represent technical conversion of a set of *input commodities* with one or more elements to another set of *output commodities*. The sets  $C_{vp}^{in}$  and  $C_{vp}^{out}$  list all commodities that are consumed (*in*) or generated (*out*) by process  $p$  in vertex  $v$  to facilitate the definition of several constraints.

*Storage processes*  $s \in S$  represent technical facilities to store and later retrieve a commodity. The subset  $S_{vc}$  lists all storages that store a given commodity  $c$  at vertex  $v$ .

*Transmission processes*  $f \in F$  represent technical means to transfer a commodity from one vertex to another along an arc  $a$ . The sets  $F_{vc}^{exp}$  and  $F_{vc}^{imp}$  list all transmission processes that allow to export (or import) commodity  $c$  from (or to) site  $v$ .

## Parameters

Like in model **URBS**, parameters are all pre-defined numerical values in the optimisation problem. They are grouped in two types: technical and economic parameters.

**Technical parameters** Table 3.7 summarises the technical model parameters. The first two parameters concern *time*. The scaling factor  $w$  is used to scale variable costs up to annual costs. That way, the cost function value  $\zeta$  always corresponds to annual costs, independent of the number of time steps. The time step duration parameter  $\Delta t$ , together with the number of time steps  $|T_m|$  determines the value for  $w$  by the expression

$$w = \frac{8760 \text{ h}}{|T_m| \cdot \Delta t}. \quad (3.34)$$

The main technical parameter is the *demand*  $d_{vct}$  time series. It is defined for each demand commodity  $c \in C_{dem}$ . Its value is the mean power flow required for that commodity at instance  $t \in T$  for the time interval of length  $\Delta t$ . The canonical unit for all power quantities here is

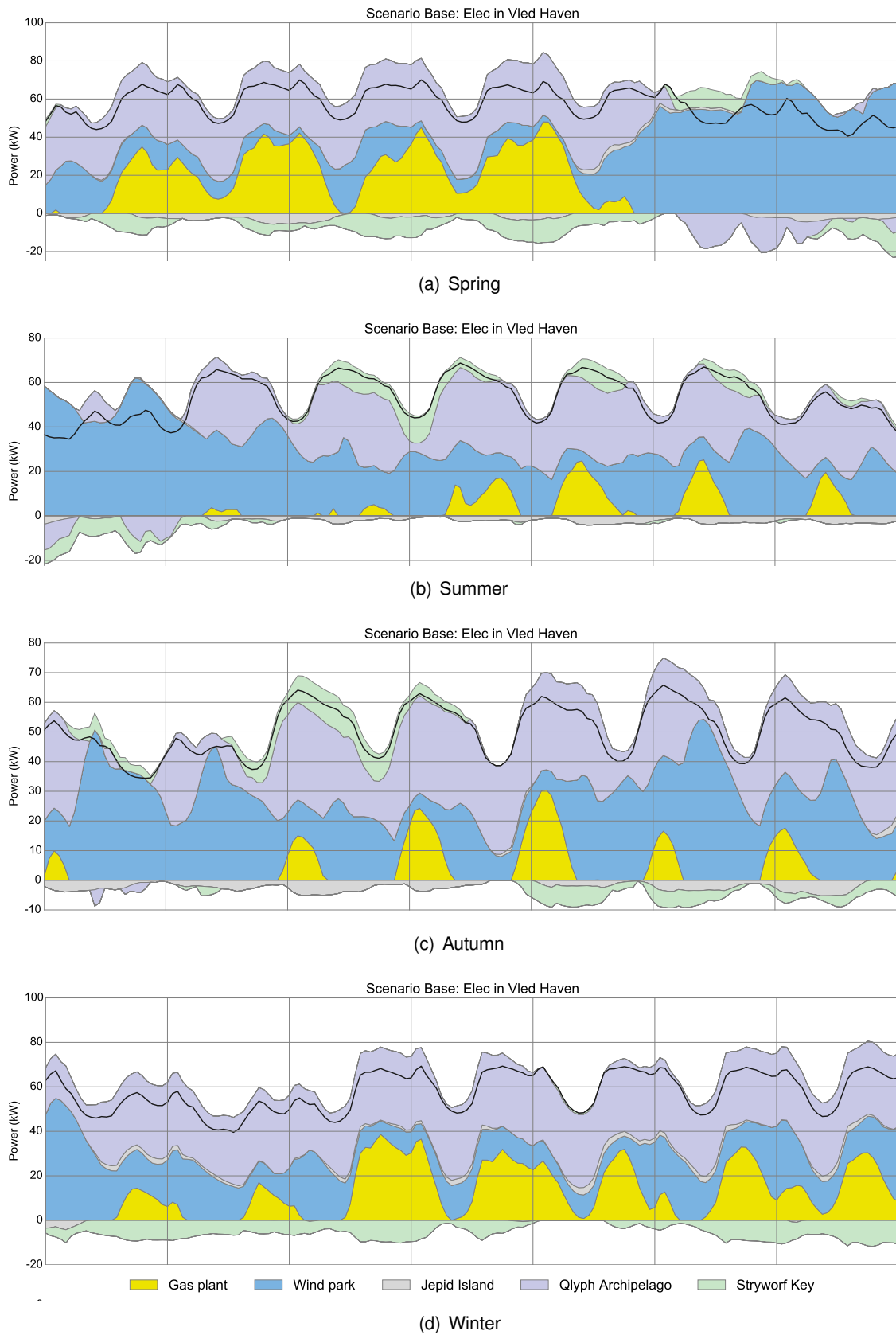


Figure 3.8: Example plot, base scenario; occasional gas usage as backup for wind power

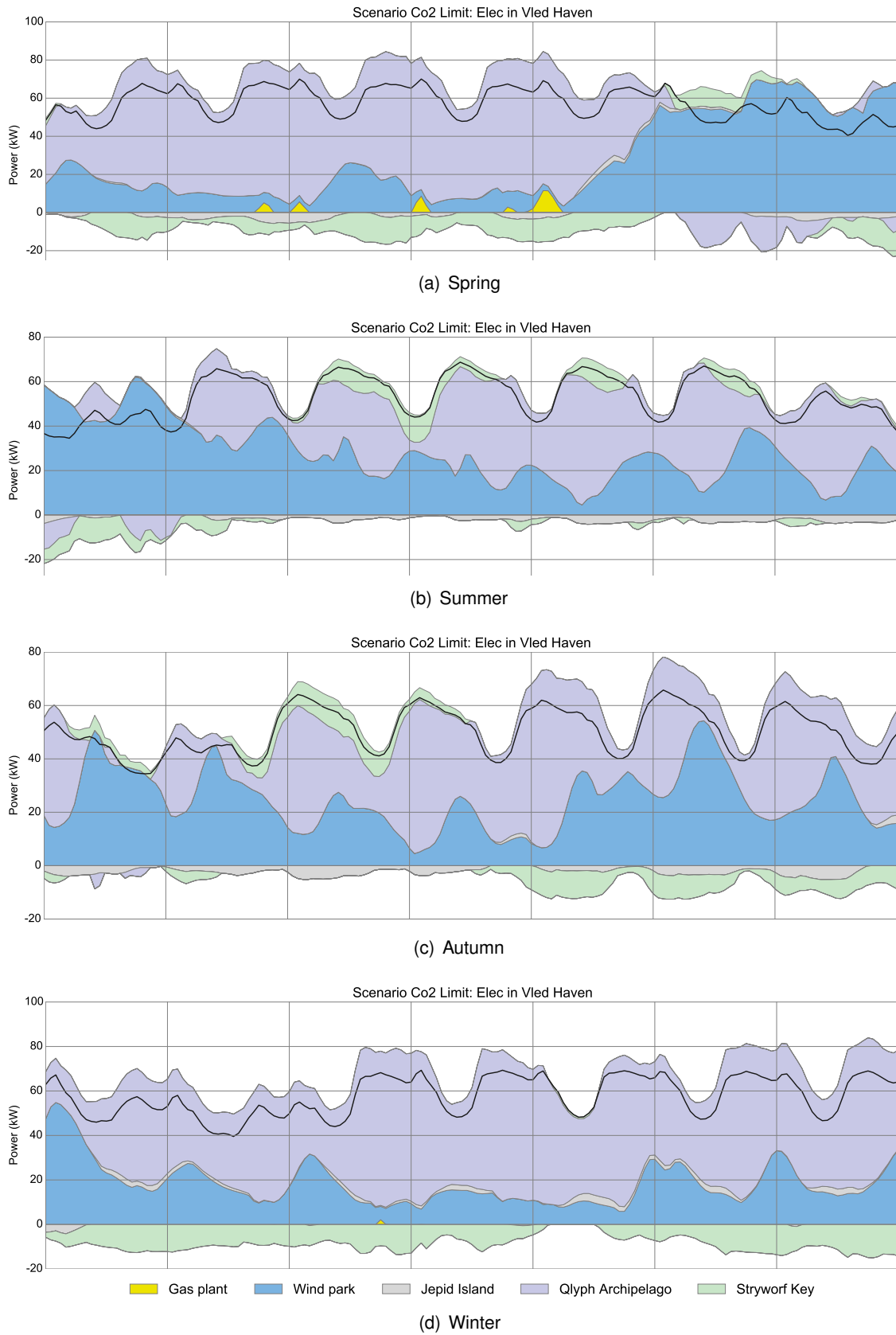


Figure 3.9: CO<sub>2</sub> limit scenario; almost no gas usage; storage in Qlyph Archipelago used as backup

kW, MW, or GW (depending on the scale). For energy quantities (e.g. storage content), the corresponding unit kWh, MWh, or GWh is to be used.

All units can be changed by the model user without any code changes. However, **urbs** does not support unit conversions within the model. That means that all parameters (i.e. costs and capacities) must be scaled with respect to those units. So, if one wants to use every quantity in J, ktoe or btu, convert *all* parameters of *all* processes, storage and transmission processes to the same unit.

The supply time series  $s_{vct}$  provides the normalised production for each intermittent commodity  $c \in C_{\text{sup}}$ . Each process with an intermittent input derives its activity directly from the evolution of that curve. That means, the process activity is directly derived from that input by equations (3.47) and (3.49), leading to the following relationship between the intermittent supply time series  $s_{vct}$  and the process activity  $\tau_{vpt}$ :

$$\forall v \in V, p \in P, c \in C_{\text{sup}}, t \in T: \quad \tau_{vpt} = (\kappa_{vp} s_{vct}) / r_{pc}^{\text{in}}. \quad (3.35)$$

In other words: intermittent supply commodities directly determine the throughput  $\tau_{vpt}$  or activity of processes that consume those commodities. Therefore, a single process cannot simultaneously consume *multiple* intermittent supply commodities without immediately leading to a infeasible problem statement: both commodities' time series would demand different process activities for the same time step. If such a feature is required, equation (3.49) must be changed to an inequality instead.

The next parameters are limits on the use or creation of certain commodities. The parameters  $\bar{l}_{vc}$  and  $\bar{L}_{vc}$  limit how much of a stock commodity  $c \in C_{\text{st}}$  may be used per time step and per year. If the simulation duration is less than a year, the scaling factor  $w$  is used for upscaling. The parameters  $\bar{m}_{vc}$  and  $\bar{M}_{vc}$  on the other hand limit how much of an environmental commodity  $c \in C_{\text{env}}$  may be produced per time step and per year. Again, the scaling factor  $w$  is used for scaling the annual limit to the simulation duration. The special limit parameter  $\bar{L}_{\text{CO}_2}$  for greenhouse gas commodity  $\text{CO}_2$  can be used to limit the annual generation of that commodity over all sites  $V$ . Its main use is to find the cost-optimal carbon abatement energy system for a given region by not having to specify emission targets for single locations  $v \in V$  through the  $\bar{M}_{vc}$  limit parameter.

The next five parameters concern processes. Existing capacities of installed processes are encoded in parameter  $K_{vp}$ , while minimum required and maximum allowed capacities can be set using the lower and upper bound parameters  $\underline{K}_{vp}$  and  $\bar{K}_{vp}$ . These bounds can be set to 0 or  $\infty$ , respectively, to deactivate them. The input and output ratios  $r_{pc}^{\text{in}}$  and  $r_{pc}^{\text{out}}$  are the next two process parameters and the key difference to the previous models **URBS**, which only features a single efficiency parameter per process. Here, they establish a fixed ratio between input (or output) of a process and its *throughput*, which can be conceptualised as the process' activity, relative to its total installed capacity.

Input and output ratios create an additional degree of freedom to the semantics of modelling a conversion process. By multiplying both input and output ratios of a process by a common



factor, the relationships between inputs and outputs remain unchanged, while the value of the throughput variable increases or decreases. This means, the *meaning* of the throughput variable can be *chosen* as either to correspond to an output quantity or alternatively the main consumed input. As a consequence, the capacity limits and investment costs must be adjusted to refer to the same quantity.

A common example is the modelling of a power plant by means of an energy conversion process with a conversion efficiency of fuel to output of 35 %. Either, one could choose  $r_{pc}^{in} = 1$  and  $r_{pc}^{out} = 0.35$ . Then, the throughput variable and the capacity variable have the same scaling (and thus meaning) as the *thermal power* of the power plant. Cost parameters for investment costs must therefore be suitably adjusted. If one chooses the values  $r_{pc}^{in} = 2.86 \approx \frac{1}{.35}$  and  $r_{pc}^{out} = 1$ , the meaning of the throughput variable is changed to the electricity output, and variable and investment costs therefore refer to the electrical output capacity, as is common in energy economy.

Next are storage parameters. The initial energy content of a storage technology  $s \in S$  at location  $v \in V$  is given, relative to its total storage capacity  $\kappa_{vs}$ , by the parameter  $I_{vs}$ . This is specified in equation (3.64). It is also the required minimum energy content at the final time as required by equation (3.65). Just like processes, both bounds and installed capacities can be specified per technology and location for both storage capacity (superscript  $o^c$ ) and storage power (superscript  $o^p$ ), leading to a total of six parameters. Storage losses are modelled during storing and retrieving energy by the efficiency parameters  $e_{vs}^{in}$  and  $e_{vs}^{out}$ . Both capture the proportion of energy that is retained (storing) or returned (retrieval) in relation to the storage content.

The final technical parameters concern energy transmission. For each connection between two locations  $a = (v_1, v_2)$  and transmission technology  $f \in F$ , a transmission efficiency  $e_{af}$  must be defined. By convention, it is identical for both directed arcs  $a$  and  $a' = (v_2, v_1)$ . This convention also applies to the three capacity parameters for lower/upper  $\underline{K}_{af}/\overline{K}_{af}$  bound and installed capacity  $K_{af}$ .

**Economic parameters** Table 3.8 summarises the economic model parameters. Every capacity increase causes investment costs, every capacity causes fixed costs, and any activity (i.e. energy flow) causes variable costs. How strongly something induces costs (if at all), is a matter of parametrisation according to the modelled system, technologies and commodities.

Stock commodity use can be taxed by charging for its introduction into the model using the *fuel cost parameter*  $k_{vc}^{fuel}$ . They can be location dependent, allowing for equal (world market, no significant transport costs involved) or different (no world market, significant transport costs involved) prices for certain goods at different places.

Energy conversion processes, just like storage and energy transmission, may exhibit three types of costs, which apply differently to different technologies. Investment costs  $k_{vp}^{inv}$  are the annualised investment costs associated with an increase of capacity of that process at that location  $\hat{\kappa}_{vp}$ . That means: existing capacity  $K_{vp}$  does not cause investment costs. Fixed costs  $k_{vp}^{fix}$  on the other hand apply to the total capacity  $\kappa_{vp}$  of a process. Variable costs are similar to fuel costs  $k_{vp}^{var}$  in that they are dependent on the process activity  $\tau_{vpt}$ , but different in that they

Table 3.7: **urbs** technical model parameters

Name	Unit	Description
$w$	1	Scaling factor for variable costs to annual duration
$\Delta t$	h	Time step duration; default: 1 h
$d_{vct}$	kW	Demand for commodity $c$ at time $t$ in site $v$
$s_{vct}$	kW	Intermittent supply capacity factor
$\bar{I}_{vc}$	kW	Maximum stock supply limit per time step
$\bar{L}_{vc}$	kWh	Maximum annual stock supply limit
$\bar{m}_{vc}$	kg/h	Maximum environmental output per time step
$\bar{M}_{vc}$	kg	Maximum annual environmental output
$\bar{L}_{CO_2}$	kg	Maximum global annual CO <sub>2</sub> emission limit
$\underline{K}_{vp}, K_{vp}, \bar{K}_{vp}$	kW	process capacity (lower bound, installed, upper bound)
$r_{pc}^{in}, r_{pc}^{out}$	1	process input/output ratio
$I_{vs}$	1	initial and final storage content (relative)
$e_{vs}^{in}, e_{vs}^{out}$	1	storage efficiency (during charge, discharge)
$\underline{K}_{vs}^c, K_{vs}^c, \bar{K}_{vs}^c$	kWh	storage content (lower bound, installed, upper bound)
$\underline{K}_{vs}^p, K_{vs}^p, \bar{K}_{vs}^p$	kW	storage power (lower bound, installed, upper bound)
$e_{af}$	1	transmission efficiency
$\underline{K}_{af}, K_{af}, \bar{K}_{af}$	kW	transmission capacity (lower bound, installed, upper bound)

represent any activity-dependent costs like wear-and-tear of equipment, thermal loads or any other effect that degrades the technology at hand, the more it is used.

Storage technologies share all these parameters, but have distinct sets for both storage capacity (superscript  $^c$ ) and storage power (superscript  $^p$ ). This allows for a wide range of technologies to be modelled comprehensively, at the cost of having to introduce many parameters with value 0 for any given technology. The storage power investment costs  $k_{vs}^{p,inv}$  are to be paid to increase both the maximum possible storing and retrieval powers at location  $v \in V$  for storage technology  $s \in S$ .<sup>3</sup> Storage size investment costs  $k_{vs}^{c,inv}$  are to be paid for expansion of the storage size  $\hat{k}_{vs}^c$ . Fixed costs are to be paid for the total capacity, both existing and newly built. Variable costs for storage power  $k_{vs}^{p,var}$  are activity-dependent costs for storing or retrieving energy.

*Investment costs:* Pumped hydro storage are a prime example for a technology with independent investment costs for capacity (the water reservoirs) and power (pumps and generators). Batteries, on the other hand, usually have coupled power and capacity. To model them in **urbs**, either the storage size or storage power cost parameter should be set (close) to zero, depending on which capacity will usually be the limiting (i.e. cost-driving) quantity. For stationary lithium batteries, for example, cells usually have a much higher power potential than is used. In that case, storage power investment costs can be set to a near-zero value, while the storage size investment costs should reflect the full storage system investment costs.

<sup>3</sup>If needed, even separate cost parameters and variables for ingoing and outgoing power flow capacities could be defined.

Table 3.8: **urbs** economic model parameters

Name	Unit	Description
$k_{vc}^{fuel}$	€/kWh	stock commodity fuel costs
$k_{vp}^{inv}$	€/(kW a)	annualised process capacity investment
$k_{vp}^{fix}$	€/(kW a)	process capacity fixed costs
$k_{vp}^{var}$	€/kWh	process variable costs
$k_{vs}^{p,inv}$	€/(kW a)	annualised storage power investment
$k_{vs}^{p,fix}$	€/(kW a)	annual storage power fixed costs
$k_{vs}^{p,var}$	€/kWh	storage power variable costs
$k_{vs}^{c,inv}$	€/(kWh a)	annualised storage size investment
$k_{vs}^{c,fix}$	€/(kWh a)	annual storage size fixed costs
$k_{vs}^{c,var}$	€/kWh	storage usage variable costs
$k_{af}^{inv}$	€/(kW a)	annualised transmission capacity investment
$k_{af}^{fix}$	€/(kW a)	annual transmission capacity fixed costs
$k_{af}^{var}$	€/kWh	transmission usage variable costs

*Variable costs:* Most storage processes will not have costs associated with their storage content  $k_{vs}^{c,var}$ , so this parameter can be set to 0 for most storage technologies. A spinning wheel, however, might have significant wear and tear associated with maintaining its rotation, causing significant costs for maintaining a certain energy content  $\epsilon_{vst}^{con}$ . This technology therefore could well receive more realistic operations schedules by setting this parameter to a non-zero value.

## Variables

Table 3.9 summarises all variables of model **urbs**. It should not surprise that there is only a small change from table 3.5 (model **URBS**). The table is grouped roughly by type: costs, commodity, process, storage and transmission.

The first row contains the *cost* variable  $\zeta$ , which is the value of the objective function. Actually, the cost variable is the sum of four separate cost variables: Investment costs are summed in  $\zeta_{inv}$ , fixed costs in  $\zeta_{fix}$ , variable costs in  $\zeta_{var}$  and fuel costs in  $\zeta_{fuel}$ .

The next row deals with *commodities*, which are represented by the single variable  $\rho_{vct}$ . It represents the purchase of energy in form of stock commodities. Together with the intermittent supply parameter  $s_{vct}$ , they both constitute all sources of energy in the model. The variable is also used to calculate the fuel costs in equation (3.40).

*Process* variables represent power flows and capacities. The power flow through a process, or its activity level, is represented by throughput variable  $\tau_{vpt}$ . It measures the weighted sum of all input commodity flows  $\epsilon_{vpct}^{in}$ , as defined in equation (3.47). Conversely, the outputs of a process  $\epsilon_{vcp}^{out}$  are derived from the process throughput using equation (3.48). The throughput is limited by the total capacity  $\kappa_{vp}$  of that process. It can be increased from the existing capacity (parameter  $K_{vp}$ ) by installing new capacity  $\hat{\kappa}_{vp}$ . The total capacity is the sum of existing and newly installed capacity, as defined in equation (3.46).

Table 3.9: **urbs** model variables

Name	Unit	Description
$\zeta$	€	Total system cost, by type (Inv, Fix, Var, Fuel)
$\rho_{vct}$	kW	stock commodity source term
$\epsilon_{vcpt}^{in}, \epsilon_{vcpt}^{out}$	kW	process input/output commodity flow
$\tau_{vpt}$	kW	process throughput
$\kappa_{vp}, \hat{\kappa}_{vp}$	kW	process capacity (total, new)
$\epsilon_{vst}^{in}, \epsilon_{vst}^{out}$	kW	storage power flow (in=charge, out=discharge)
$\epsilon_{vst}^{con}$	kWh	storage energy content
$\kappa_{vs}^c, \hat{\kappa}_{vs}^c$	kWh	storage content capacity (total, new)
$\kappa_{vs}^p, \hat{\kappa}_{vs}^p$	kW	storage power capacity (total, new)
$\pi_{aft}^{in}, \pi_{aft}^{out}$	kWh/h	transmission power flow (input, output)
$\kappa_{af}, \hat{\kappa}_{af}$	kW	transmission capacity (total, new)

*Storage* variables also fall into the two categories of flows and capacities. The main variable is the storage energy content  $\epsilon_{vst}^{con}$ . It is increased by ingoing power flow  $\epsilon_{vst}^{in}$  and reduced by outgoing power flow  $\epsilon_{vst}^{out}$ . The storage content is limited by the storage content capacity  $\kappa_{vs}^c$ . Both ingoing and outgoing power flow are limited by the storage power capacity  $\kappa_{vs}^p$ . Storage does not include a self-discharge feature, but both ingoing and outgoing power flow are subject to an efficiency parameter as described in the previous section, or specified in equation (3.57).

*Transmission* variables also consist of flow and capacity variables. This means that symmetric bidirectional infrastructure (like most pressure pipe networks or power networks) is represented by two (identical) transmission capacities for both directional arcs  $a$  and  $a'$  between a given pair of neighbored vertices.

### Equations

Equations are grouped into the objective function, called *cost function* here, and the longer list of *constraints*.

**Cost function** Like in model **URBS**, the objective is to minimise the value of cost variable  $\zeta$ . This section gives the definition of its four parts *investment costs*, *fixed costs*, *variables costs* and *fuel costs*. Like in the previous model, environmental constraints are modelled as an optional limit on environmental commodities (parameters  $\bar{m}_{vc}, \bar{M}_{vc}$ ) and total greenhouse gas emissions (parameter  $\bar{L}_{CO_2}$ ).

Investment costs  $\zeta_{inv}$  are incurred for all capacities that are newly built ( $\hat{\kappa}$ ) for conversion processes, storage size/power, and energy transmission. Fixed costs  $\zeta_{fix}$  have to be paid for the total capacities ( $\kappa$ ) of all technologies.

Variable costs  $\zeta_{var}$  are calculated per modelled time step  $t \in T_m$ . For energy conversion, costs are calculated based on the process throughput  $\tau_{vpt}$ . For energy energy transmission, the ingoing power flow  $\pi_{aft}^{in}$  determines the costs. Storage technologies can have costs both for maintaining a certain storage level  $k_{vs}^{c,var}$  as well as charging/discharging costs  $k_{vs}^{p,var}$ . All variable costs are multiplied by the time step length  $\Delta t$  and the time step weight  $w$  to scale

them up to one year, independent of the simulation duration. Fuel costs  $\zeta_{\text{fuel}}$  finally price the purchased stock commodity power flow  $\rho_{vct}$ .

$$\zeta = \zeta_{\text{inv}} + \zeta_{\text{fix}} + \zeta_{\text{var}} + \zeta_{\text{fuel}} \quad (3.36)$$

$$\zeta_{\text{inv}} = \sum_{\substack{v \in V \\ p \in P}} \hat{\kappa}_{vp} k_p^{\text{inv}} + \sum_{\substack{v \in V \\ s \in S}} \left( \hat{\kappa}_{vs}^c k_{vs}^{\text{c,inv}} + \hat{\kappa}_{vs}^p k_{vs}^{\text{p,inv}} \right) + \sum_{\substack{a \in A \\ f \in F}} \hat{\kappa}_{af} k_{af}^{\text{inv}} \quad (3.37)$$

$$\zeta_{\text{fix}} = \sum_{\substack{v \in V \\ p \in P}} \kappa_{vp} k_{vp}^{\text{fix}} + \sum_{\substack{v \in V \\ s \in S}} \left( \kappa_{vs}^c k_{vs}^{\text{c,fix}} + \kappa_{vs}^p k_{vs}^{\text{p,fix}} \right) + \sum_{\substack{a \in A \\ f \in F}} \kappa_{af} k_{af}^{\text{fix}} \quad (3.38)$$

$$\zeta_{\text{var}} = w \sum_{t \in T_m} \left( \sum_{\substack{v \in V \\ p \in P}} \tau_{vpt} k_{vp}^{\text{var}} \Delta t + \sum_{\substack{a \in A \\ f \in F}} \pi_{aft}^{\text{in}} k_{af}^{\text{var}} \Delta t + \sum_{\substack{v \in V \\ s \in S}} \left[ \epsilon_{vst}^{\text{con}} k_{vs}^{\text{c,var}} + \left( \epsilon_{vst}^{\text{in}} + \epsilon_{vst}^{\text{out}} \right) k_{vs}^{\text{p,var}} \Delta t \right] \right) \quad (3.39)$$

$$\zeta_{\text{fuel}} = w \sum_{t \in T_m} \sum_{v \in V} \sum_{c \in C_{\text{stock}}} \rho_{vct} k_{vc}^{\text{fuel}} \Delta t \quad (3.40)$$

**Constraints** To facilitate the formulation of commodity constraints, the following term is introduced.  $\text{CB}(v, c, t)$  represents the *commodity balance* of commodity  $c$  in vertex  $v$  at time step  $t$ . Consumption, export and storage input are counted positive, while creation, import and storage output are counted negative:

$$\text{CB}(v, c, t) = \sum_{p|c \in C_{vp}^{\text{in}}} \epsilon_{vcpt}^{\text{in}} - \sum_{p|c \in C_{vp}^{\text{out}}} \epsilon_{vcpt}^{\text{out}} + \sum_{s \in S_{vc}} \left( \epsilon_{vst}^{\text{in}} - \epsilon_{vst}^{\text{out}} \right) + \sum_{\substack{a \in A_v^{\text{s}} \\ f \in F_{vc}^{\text{exp}}}} \pi_{aft}^{\text{in}} - \sum_{\substack{a \in A_v^{\text{p}} \\ f \in F_{vc}^{\text{imp}}}} \pi_{aft}^{\text{out}}$$

*Vertex equation* is the main constraint that has to be satisfied for every commodity in the reduced set  $C^v = C \setminus (C_{\text{env}} \cup C_{\text{sup}})$ . For non-demand commodities, the summand  $d_{vct}$  is omitted, while for non-stock commodities, the term  $\rho_{vct}$  is skipped or set to zero. The inequality requires, that any consumption ( $\text{CB} > 0$ ) must be supplied by a corresponding source term ( $\rho_{vct} > 0$ ), or that any demand ( $d_{vct} > 0$ ) must be met either by conversion from other commodities, storage or import ( $\text{CB} < 0$ ). The formulation as an inequality allows surplus of a demand commodity without penalty, i.e. it assumes that waste energy can be dumped for negligible costs.

$$\forall v \in V, c \in C^v, t \in T: \quad \rho_{vct} - \text{CB}(v, c, t) - d_{vct} \geq 0 \quad (3.41)$$

*Commodity constraints* concern *stock* and *environmental* commodities separately. The first constraint limits the amount of stock commodity use per time step. The second constraint

limits the total stock commodity use over the whole time range. The fraction of both right-hand side parameters  $\bar{I}_{vc}/I_{vc}$  is indicative of how much of the annual possible supply could be used during a single time step.

$$\forall v \in V, c \in C_{st}, t \in T: \quad \rho_{vct} \leq \bar{I}_{vc} \quad (3.42)$$

$$\forall v \in V, c \in C_{st}: \quad w \sum_{t \in T} \Delta t \rho_{vct} \leq \bar{I}_{vc} \quad (3.43)$$

The *environmental* constraints work the same way, first per time step, second for a whole year. The commodity balance function is used to capture the creation (CB < 0) of all environmental commodities  $c \in C_{env}$ :

$$\forall v \in V, c \in C_{env}, t \in T: \quad -CB(v, c, t) \leq \bar{m}_{vc} \quad (3.44)$$

$$\forall v \in V, c \in C_{env}: \quad -w \sum_{t \in T} \Delta t CB(v, c, t) \leq \bar{M}_{vc} \quad (3.45)$$

*Process* constraints govern their capacities and the operation restricted by these capacities. The first constraint defines the total installed capacity as the sum of existing capacity plus newly installed capacity:

$$\forall v \in V, p \in P: \quad \kappa_{vp} = K_{vp} + \hat{\kappa}_{vp} \quad (3.46)$$

*Process input/output* definitions link the throughput variable to the input and output streams via the input and output ratios:

$$\forall v \in V, p \in P, t \in T: \quad \epsilon_{vpct}^{in} = \tau_{vpt} r_{pc}^{in} \quad (3.47)$$

$$\forall v \in V, p \in P, t \in T: \quad \epsilon_{vpct}^{out} = \tau_{vpt} r_{pc}^{out} \quad (3.48)$$

If a process has intermittent input commodities, their availability is determined (fixed) by the capacity factor time series  $s_{vct}$ :

$$\forall v \in V, p \in P, c \in C_{sup}, t \in T: \quad \epsilon_{vpct}^{in} = \kappa_{vp} s_{vct} \quad (3.49)$$

The *process throughput by capacity* constraint states that the process throughput is limited by the total installed process capacity:

$$\forall v \in V, p \in P, t \in T: \quad \tau_{vpt} \leq \kappa_{vp} \quad (3.50)$$

Finally, *process capacity* restricts the total installed capacity to the range given by lower and upper bounds:

$$\forall v \in V, p \in P: \quad \underline{K}_{vp} \leq \kappa_{vp} \leq \bar{K}_{vp} \quad (3.51)$$

*Transmission* has five constraints: the first constraint defines the total transmission capacity by the sum of already existing and newly installed capacity:

$$\forall a \in A, f \in F: \quad \kappa_{af} = K_{af} + \hat{\kappa}_{af} \quad (3.52)$$

*Transmission output* calculates the output power of a transmission by multiplying its input by the transmission efficiency  $e_{af}$ :

$$\forall a \in A, f \in F, t \in T: \quad \pi_{aft}^{\text{out}} = \pi_{aft}^{\text{in}} e_{af} \quad (3.53)$$

*Transmission input by capacity* limits the ingoing power flow to a transmission by its total capacity:

$$\forall a \in A, f \in F, t \in T: \quad \pi_{aft}^{\text{in}} \leq \kappa_{af} \quad (3.54)$$

*Transmission capacity* restricts the total transmission capacity to lie within the range spanned by the lower and upper bound parameters for that technology and location:

$$\forall a \in A, f \in F: \quad \underline{K}_{af} \leq \kappa_{af} \leq \bar{K}_{af} \quad (3.55)$$

*Transmission symmetry* requires that the total transmission capacity for a transmission  $a = (v_1, v_2)$  is identical to its reversed counterpart  $a' = (v_2, v_1)$ . This design choice is explained further in paragraph **Transmission** in section 2.2.2.

$$\forall a \in A, f \in F: \quad \kappa_{af} = \kappa_{a'f} \quad (3.56)$$

*Storage state* is the main storage constraint. It calculates the current state of charge/filling as the previous state plus ingoing minus outgoing energy. Storing/retrieval efficiencies ( $e_{vs}^{\text{in}}/e_{vs}^{\text{out}}$ ) are included as well:

$$\forall v \in V, s \in S, t \in T_m: \quad \epsilon_{vst}^{\text{con}} = \epsilon_{vst(t-1)}^{\text{con}} + \epsilon_{vst}^{\text{in}} \cdot e_{vs}^{\text{in}} - \epsilon_{vst}^{\text{out}} / e_{vs}^{\text{out}} \quad (3.57)$$

The two *storage capacity* rules define the total capacities for storage content ( $\kappa_{vs}^{\text{c}}$ ) and storage power ( $\kappa_{vs}^{\text{p}}$ ) as the sum of already installed capacity (parameter) and newly built capacity (variable):

$$\forall v \in V, s \in S: \quad \kappa_{vs}^{\text{c}} = K_{vs}^{\text{c}} + \hat{\kappa}_{vs}^{\text{c}} \quad (3.58)$$

$$\forall v \in V, s \in S: \quad \kappa_{vs}^{\text{p}} = K_{vs}^{\text{p}} + \hat{\kappa}_{vs}^{\text{p}} \quad (3.59)$$

*Storage capacity limits* restrict the two storage capacity variables to lie within the specified lower and upper bounds:

$$\forall v \in V, s \in S: \quad \underline{K}_{vs}^{\text{c}} \leq \kappa_{vs}^{\text{c}} \leq \bar{K}_{vs}^{\text{c}} \quad (3.60)$$

$$\forall v \in V, s \in S: \quad \underline{K}_{vs}^{\text{p}} \leq \kappa_{vs}^{\text{p}} \leq \bar{K}_{vs}^{\text{p}} \quad (3.61)$$

*Storage content limit* the storage content by the corresponding capacity variable. The same applies to storage input and output, and its capacity variable:

$$\forall v \in V, s \in S, t \in T: \quad \epsilon_{vst}^{\text{con}} \leq \kappa_{vs}^{\text{c}} \quad (3.62)$$

$$\forall v \in V, s \in S, t \in T: \quad \epsilon_{vst}^{\text{in}}, \epsilon_{vst}^{\text{out}} \leq \kappa_{vs}^{\text{p}} \quad (3.63)$$

*Initial and final storage state* define and restrict the storage state at the beginning and end of the simulation time span of a quantity relative to the total capacity:

$$\forall v \in V, s \in S: \quad \epsilon_{vst_0}^{\text{con}} = \kappa_{vs}^c I_{vs} \quad (3.64)$$

$$\forall v \in V, s \in S: \quad \epsilon_{vst_N}^{\text{con}} \geq \kappa_{vs}^c I_{vs} \quad (3.65)$$

Like noted in the presentation of the previous model, this formulation is less general than requesting the storage to be at least as full in the end  $t_N$  as in the beginning  $t_0$ . However, this loss of generality greatly decreases the required computation time. If there is no prior knowledge about seasonality (and thus a sensible initial storage state),  $I_s = 0.5$  has been used with good success. If more than about  $10^3$  time steps are simulated, the influence of this parameter on the overall solution is negligible.

**Emissions** A special hard-coded rule is added to allow for setting a maximum global upper bound on the output of the environmental commodity  $\text{CO}_2$ . Different from the settings in parameter  $\bar{M}_{vc}$ , which applies for individual sites  $v$ , the solver can be forced to decide on *where* to save emissions:

$$w \sum_{t \in T_m} \sum_{v \in V} \text{CB}(v, \text{CO}_2, t) \leq \bar{L}_{\text{CO}_2} \quad (3.66)$$

Equations (3.36) to (3.66) together define the linear program **urbs**, whose optimal solution has the desired minimum costs  $\zeta^*$ .



### 3.3 DHMIN

This section presents a mixed-integer optimisation program for finding the cost-optimal structure and size of a single-commodity flow network for a given graph, originally published in IEEE Transactions on Smart Grid in 2014 [24]. The decision whether to build or not to build a pipeline on a given edge is modelled as a binary variable, while the costs for construction scale linearly with the thermal input power into the pipe. While it would be more realistic to consider a finite set of discrete pipe diameters, the resulting problem becomes prohibitively difficult to solve for modest problem sizes.

Unlike **urbs** and **rivus**, the **DHMIN** model family does *not require* the resulting network to satisfy *any energy demand at all*. It is rather the task of the solver to find regions of profitable streets in the graph that have sufficiently high energy demand to pay off the investment costs for their connection to energy sources. In the formulation presented here, the model does not include connection costs for individual customers.

#### 3.3.1 Origin

The model is a slightly modified, but completely re-implemented version of the (unpublished) optimisation model **DHoptimal** that was created for and used in several projects with city municipalities for designing cost-optimal district heating networks. However, just like the original URBS method, it was never fully documented. The key change from **DHoptimal** to **DHMIN** is the removal of a *discrete* variable for the capacity of a pipe. In **DHoptimal**, a fixed set of pipe capacities, derived from the DN diameter specifications for pipe diameters, are allowed. **DHMIN** on the other hand allows for continuous pipe capacities. This difference is visualised in figure 3.10. Left shows a discrete cost function that has a small finite number of cost values. If the capacity of one pipe diameter is exceeded, the next bigger capacity must be installed instead. On the right the linear continuous approximation is shown instead. While it is true that certain desired capacities are suboptimal within the framework of fixed DN pipe diameters, the variability in operation can spoil the benefit of the more precise cost function.

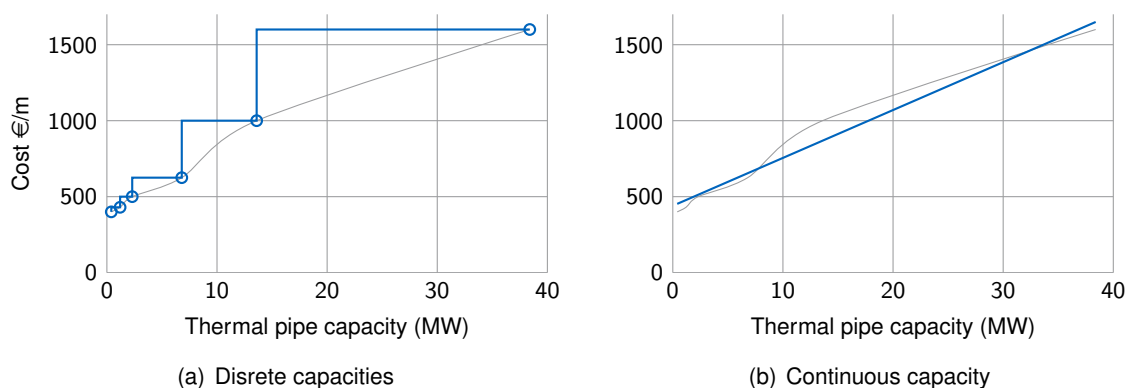


Figure 3.10: Comparison of discrete and continuous pipe capacity costs



Figure 3.11: Input/output flow chart of model **DHMIN**

### 3.3.2 Design goal

The model approaches the planning task for a district heating network from the perspective of a city utility company. It focuses on answering the question: what is the cost-optimal extent of a district heating network? The model therefore has to balance costs for generating and distributing thermal energy, given a set of plant locations that can deliver heat of a given cost. It can then derive the regions in a city that are most economic places for new district heating pipes. The input/output schema of this model is summarised in figure 3.11.

### 3.3.3 Mathematical description

Historically, this model's use of the word *edge* is different from the remaining thesis: here, they refer to an *ordered* pair of vertices. In the remaining document, such a pair is referred to as an *arc*.

The model's optimal solution minimises the total costs  $\zeta$  of generating and distributing heat through a district heating network, reduced by revenues for supplied heat energy. The key result is the most economic size and topology of the heat distribution network, represented by the value of power flow variable  $\pi_{ij}^{\text{in}}, e_{ij} \in E$  in case of minimum costs:

$$\pi_{ij}^* = \underset{\pi_{ij}^{\text{in}}}{\operatorname{argmin}} \zeta$$

#### Sets

In this model, a district or city is represented as a graph of vertices and edges. This graph can be derived from the street network, as it covers most potential sites for burying a pipe network. Let  $V$  be the set of vertices  $v_i$ , corresponding to street intersections or endpoints. Set  $E$  of edges then comprises ordered tuples of vertices  $e_{ij} = (v_i, v_j)$  with  $i \neq j$ . Here,  $E$  is constructed to be symmetric, that means either both or none of the pair  $e_{ij}$  and  $e_{ji}$  are elements of  $E$ . For readability, the subscript  $\circ_i$  is used to denote any parameter or variable that is defined over vertices  $v_i \in V$ , while  $\circ_{ij}$  is used to denote a quantity defined over edges  $e_{ij} \in E$ .

The set  $V_0 \subseteq V$  defines so-called *source vertices*, which represent locations of possible heat sources. Usually, there is only a small number ( $\leq 10$ ) of source vertices. The example graph in Figure 3.12 shows a visual representation of a small graph. Each line between two vertices represents a pair of symmetric edges.

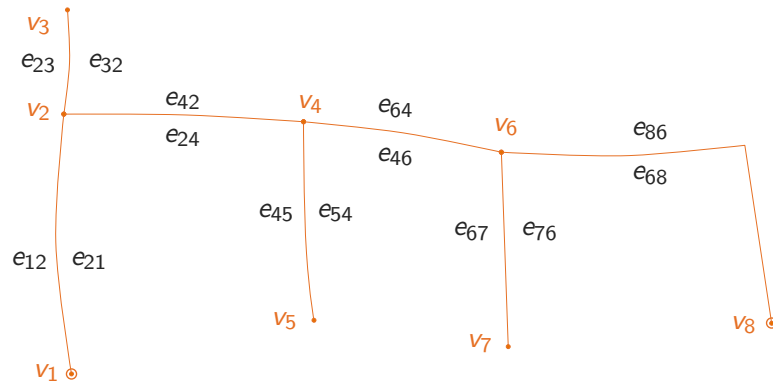


Figure 3.12: Example street graph with 8 vertices, 14 edges, and 2 source vertices  $v_1, v_8$

The set of neighbors  $N_i$  for a vertex  $v_i$  is defined by the indices of all vertices that are connected to it by an edge:  $N_i = \{k \mid e_{ki} \in E\}$ . In the example graph in Figure 3.12, the neighbor set of vertex  $v_4$  is equal to  $N_4 = \{2, 5, 6\}$ .

### Parameters

Parameters are the numerical inputs of the optimisation problem. They are summarised in table 3.10. The upper part contains global parameters; the lower part summarises edge and vertex parameters.

**Vertices** Vertices have a maximum capacity  $Q_i^{\max}$ , the thermal output power given in kW. For all non-source vertices, i.e. almost all vertices, this parameter is set to 0.

**Edges** Edges have the three main attributes  $l_{ij}$ ,  $d_{ij}$  and  $D_{ij}$  for length (m), peak demand (kW) and annual demand (kWh/a). Peak demand determines the required pipe size, while annual demand determines revenue for supplying heat. Parameter  $\epsilon_{ij}$  is a binary indicator that denotes existing ( $\epsilon_{ij} = 1$ ) pipelines. Parameter  $C_{ij}^{\max}$  indicates the maximum thermal power capacity (kW) in an edge, which can be derived from the maximum available pipe diameter. Attribute values for the symmetric edge pairs  $e_{ij}$  and  $e_{ji}$  are identical.

**Global** The letter  $c$  and a subscript denote *economic parameters*. These are investment costs for building the pipe network  $c_{\text{fix}}$  and  $c_{\text{var}}$ , maintenance  $c_{\text{om}}$ , costs of providing  $c_{\text{heat}}$  heat, and *revenue* for delivering  $c_{\text{rev}}$  heat to consumers. While  $c_{\text{fix}}$  contains the size-independent costs (mainly earth works),  $c_{\text{var}}$  contains costs that are dependent on the thermal capacity (diameter) of the pipe.

*Technical parameters* concern thermal losses in the pipe network  $w_{\text{fix}}$  and  $w_{\text{var}}$  and the full load hours  $T_{\text{flh}}$  (operation time) of the source vertices. The value of  $T_{\text{flh}}$  can be derived from the average value of the quotient  $D_{ij}/d_{ij}$  in the investigated area and is mainly determined by the local climate and user behaviour. The dimensionless parameters  $b$  and  $q$  quantify the effects due to the aggregation of demand on street level. Parameter  $b$  is the concurrence effect, caused by the probabilistic demand for heat per consumer; it reduces the required peak demand for the heat supply network in comparison to the sum of peak demands. Parameter  $q$

Table 3.10: **DHMIN** Model Parameters

Name	Unit	Description
$c_{\text{fix}}$	€/m	Fixed investment costs
$c_{\text{var}}$	€/(m kW)	Variable investment costs
$c_{\text{om}}$	€/(m a)	Operation & maintenance costs
$c_{\text{heat}}$	€/kWh	Heat generation costs
$c_{\text{rev}}$	€/kWh	<i>Revenue</i> for delivered heat
$w_{\text{fix}}$	kW/m	Fixed thermal losses
$w_{\text{var}}$	kW/(kW m)	Variable thermal losses
$T_{\text{flh}}$	h/a	Full load hours of source vertices
$b$	1	Concurrence effect
$q$	1	Connect quota
$u$	1/a	Annuity factor for investment costs
$l_{ij}$	m	Edge length
$d_{ij}$	kW	Edge peak demand
$D_{ij}$	kWh/a	Edge annual demand
$g_{ij}$	—	Existence of a pipe (1=yes, 0=no)
$C_{ij}^{\text{max}}$	kW	Maximum pipe capacity
$Q_i^{\text{max}}$	kW	Source vertex capacity

Table 3.11: **DHMIN** Model Variables

Name	Unit	Description
$\zeta$	€	Total system cost (Inv, O&M, Rev, Heat)
$\xi_{ij}$	kW	Binary decision variable: 1 = use pipe
$\pi_{ij}^{\text{in}}$	kW	Thermal power flow from $v_i$ into edge $e_{ij}$
$\pi_{ij}^{\text{out}}$	kW	Thermal power flow out of edge $e_{ij}$ into $v_j$
$\rho_i$	kW	Heat generation power in source vertex $v_i$

is the connect quota, reflecting the fact that not all buildings in a given street will be connected by district heating. Consequently,  $b$  reduces investment costs for heat infrastructure by lowering peak demand, while  $q$  reduces the revenue for heat by lowering annual demand.

### Variables

The main optimisation happens by finding values for the *binary* decision variable  $\xi_{ij}$ . If its value is one, a pipe in edge  $e_{ij}$  is built/used and there must be a power flow  $\pi_{ij}^{\text{in}}$  in the direction  $i \rightarrow j$  into the pipe. At the same time  $\xi_{ij} = 1$  implies that the demand  $d_{ij}$  of this edge has to be satisfied. The power flow variable at the other end of the pipe is called  $\pi_{ij}^{\text{out}}$  and is reduced by thermal losses and heat demand of consumers along the edge.

The non-negative variable  $\rho_i$  represents thermal power output from source vertices.

Table 3.11 summarises all model variables. The following line shows their allowed value ranges:

$$z \in \mathbb{R} \quad \xi_{ij} \in \{0, 1\} \quad \pi_{ij}^{\text{in}}, \pi_{ij}^{\text{out}}, \rho_i \in \mathbb{R}_0^+$$

## Equations

**Cost function** Total cost  $\zeta$  is the sum of costs for investment (annualised), operation & maintenance (O&M), cost of heat generation minus revenue for delivered heat, for the period of one year. To shorten the definition of this equation, the following *derived parameters* are introduced, whose definitions are given below:  $k_{ij}^{\text{fix}}$  (€/a) and  $k_{ij}^{\text{var}}$  (€/kW a) cover the infrastructure costs (investment, O&M).  $k_{ij}^{\text{heat}}$  (€/kW) represents heat generation costs, while  $r_{ij}^{\text{heat}}$  (€/a) represents revenue for sold heat.

$$\zeta = \sum_{e_{ij} \in E} \left[ \left( k_{ij}^{\text{fix}} - r_{ij}^{\text{heat}} \right) \xi_{ij} + k_{ij}^{\text{var}} \pi_{ij}^{\text{in}} \right] + \sum_{v_i \in V_0} k_i^{\text{heat}} \rho_i \quad (3.67)$$

The terms for *investment, O&M cost*  $k_{ij}^{\text{fix}}$  and  $k_{ij}^{\text{var}}$ , together with variables  $\xi_{ij}$  and  $\pi_{ij}^{\text{in}}$ , form a piece-wise linear function. Fixed costs include power-independent investment costs (mainly earthworks) and O&M costs. Variable costs depend on the size (thermal capacity) of the built pipes. Investment costs arise only for pipes that do not exist ( $g_{ij} = 0$ ), while O&M have to be paid also for existing pipes, as long as they are used ( $\xi_{ij} = 1$ ). These considerations lead to the following definitions for  $k_{ij}^{\text{fix}}$  and  $k_{ij}^{\text{var}}$ . Parameter  $u$  is the annuity factor for the investment cost terms:

$$\forall e_{ij} \in E: \quad k_{ij}^{\text{fix}} = c_{\text{fix}} l_{ij} u (1 - g_{ij}) + c_{\text{om}} l_{ij} \quad (3.68)$$

$$\forall e_{ij} \in E: \quad k_{ij}^{\text{var}} = c_{\text{var}} l_{ij} u (1 - g_{ij}) \quad (3.69)$$

In order to visualise the behavior of the cost function for the isolated decision of building a single pipe in edge  $e_{ij}$ , Figure 3.13 shows the length-normalised cost term for investment cost, O&M, plotted as a function of the decision variable  $\xi_{ij}$  and the incoming power flow  $\pi_{ij}^{\text{in}}$ , extracted from the *cost function* (3.67):

$$\begin{aligned} & (k_{ij}^{\text{fix}} \xi_{ij} + k_{ij}^{\text{var}} \pi_{ij}^{\text{in}}) / l_{ij} \\ & = (u c_{\text{fix}} + c_{\text{om}}) \xi_{ij} + \alpha c_{\text{var}} \pi_{ij}^{\text{in}} \end{aligned}$$

Important is the unsteady jump from zero to  $u c_{\text{fix}} + c_{\text{om}}$  for any non-zero value of  $\pi_{ij}^{\text{in}}$ . The link between the binary decision variable  $\xi_{ij}$  and  $\pi_{ij}^{\text{in}}$  is enforced through the restriction inequality *pipe capacity* (3.77).

*Heat revenue*  $r_{ij}^{\text{heat}}$  is the second parameter in the cost equation. For each edge, it equals the annual heat demand  $D_{ij}$  in that edge, multiplied with the consumer price  $c_{\text{rev}}$  and scaled by the connect quota factor  $q$ :

$$\forall e_{ij} \in E: \quad r_{ij}^{\text{heat}} = c_{\text{rev}} D_{ij} q \quad (3.70)$$

*Heat generation costs*  $k_{ij}^{\text{heat}}$  occur in source nodes. The peak output power  $\pi_i$  is multiplied by the annual full-load hours  $T_{\text{flh}}$  to estimate the annual thermal *energy* output. Multiplication with the specific heat generation costs  $c_{\text{heat}}$  in €/kWh yields the heat generation costs. Division by  $b$  is required to remove the down-scaling of the cumulated peak *power* due to concurrence effects:

$$\forall v_i \in V: \quad k_i^{\text{heat}} = \frac{T_{\text{flh}} c_{\text{heat}}}{b} \quad (3.71)$$

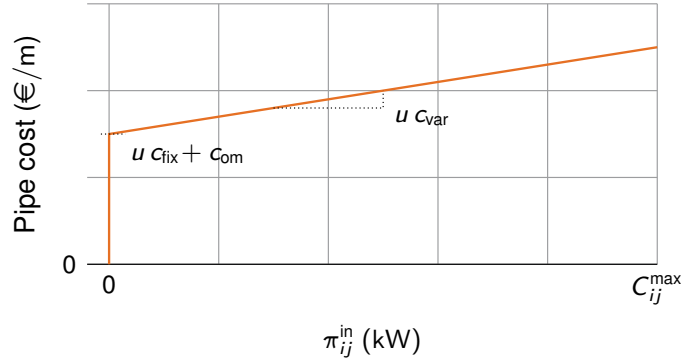


Figure 3.13: Investment and O&M costs for a single edge  $e_{ij}$  over  $\pi_{ij}^{\text{in}}$

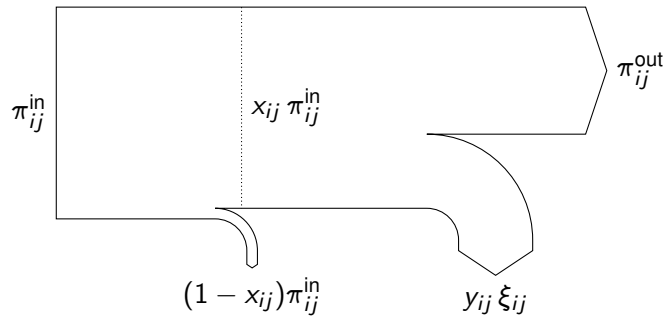


Figure 3.14: Sankey diagram of equation *demand satisfaction* (3.73)

Note that  $k_i^{\text{heat}}$  here is identical for all source nodes  $v_i \in V_0$ . However, by using plant-dependent run times  $T_i^{\text{flh}}$  and heat generation costs  $c_i^{\text{heat}}$ , this can be generalised to heat sources with different generation costs and run times.

**Constraints** *Energy conservation* is the *main vertex equation* formulated as an inequality that allows for energy to vanish. This waste energy is punished economically and thus will not occur in the optimal configuration  $\pi_{ij}^*$ . The sum of all outgoing power  $\pi_{im}^{\text{in}}$  to neighbour vertices  $v_m$ ,  $m \in N_i$  must be met either by incoming power  $\pi_{mi}^{\text{out}}$  or by power  $\pi_i$  from the source, which only can be non-zero if  $v_i$  is a source vertex because of equation (3.78):

$$\forall v_i \in V: \quad \sum_{m \in N_i} (\pi_{im}^{\text{in}} - \pi_{mi}^{\text{out}}) \leq \rho_i \quad (3.72)$$

This inequality is thus a slack-relaxed version of the law of energy conservation. For all non source vertices, the difference between outgoing and incoming power must be smaller than or equal to zero. In source vertices, a positive difference may remain, when it is met by an equal amount of input power  $\rho_i$  into the network. This constraint also ensures that power flow can be tracked back to a source vertex.

*Demand satisfaction* is the *main edge equation* linking the pipe building decision to a reduction of the power flow out of the pipe by the peak demand  $d_{ij}$ . It also includes a power- and length-dependent loss term parametrised by parameters  $w_{\text{fix}}$  and  $w_{\text{var}}$ :

$$\forall e_{ij} \in E: \quad \xi_{ij} \pi_{ij}^{\text{in}} - \pi_{ij}^{\text{out}} = y_{ij} x_{ij} \quad (3.73)$$

Here, the auxiliary parameters  $x_{ij}$  and  $y_{ij}$  are given by the following expressions:

$$\forall e_{ij} \in E: \quad x_{ij} = 1 - l_{ij} w_{\text{var}} \quad (3.74)$$

$$\forall e_{ij} \in E: \quad y_{ij} = d_{ij} b q + l_{ij} w_{\text{fix}} \quad (3.75)$$

Figure 3.14 illustrates equation (3.73) graphically. It works like a two-state switch: if  $\xi_{ij} = 1$ , i.e. a pipe in edge  $e_{ij}$  is used, the heat demand in that same edge must be satisfied, thus requiring a positive incoming power flow. On the other hand,  $\xi_{ij} = 0$  together with equation (3.77) implies that  $\pi_{ij}^{\text{in}} = 0$ , forcing incoming and outgoing power flow to zero.

*Unidirectionality* ensures that only one direction for construction, power flow and revenue is allowed per edge:

$$\forall e_{ij} \in E: \quad \xi_{ij} + \xi_{ji} \leq 1 \quad (3.76)$$

*Pipe capacity* allows power flow only if a pipe  $ij$  is built. This inequality links the binary decision variable  $\xi_{ij}$  to the investment cost term  $k_{\text{inv}}$ . Edge parameter  $C_{ij}^{\text{max}}$  is the maximum thermal power that can be transported through a pipe. It is determined by the pipe diameter that can be built (or, if  $g_{ij} = 1$ , that exists) on that edge:

$$\forall e_{ij} \in E: \quad \pi_{ij}^{\text{in}} \leq \xi_{ij} C_{ij}^{\text{max}} \quad (3.77)$$

*Source vertices* limits heat output  $Q_i$  in a vertex to  $Q_i^{\text{max}}$ , representing the maximum capacity available in that vertex. Non-source vertices have  $Q_i^{\text{max}}$  set to zero:

$$\forall v_i \in V: \quad \rho_i \leq Q_i^{\text{max}} \quad (3.78)$$

Equations (3.67) to (3.78) together define a linear mixed-integer program, whose optimal solution is the desired  $\pi_{ij}^*$ .

## 3.4 dhmin (seasonal)

### 3.4.1 Motivation for model development

While the model **DHMIN** is very usable for large-scale network *design*, it lacks one key aspect that is very important for its operation: *redundancy*. The optimised networks are only capable of satisfying the peak demand, if all used heat source vertices are fully available. This important condition, however, is often violated due to planned and unplanned outages of heat production facilities. These outages then require that heat must be supplied from a more distant source. Pipes along that alternative route often require additional capacity. What is the optimal network design, while considering these alternative power routes?

### 3.4.2 Main design idea

The key idea of model **dhmin** is to introduce the concept of time in form of multiple time steps, just like in **urbs**. But instead of using tens or hundreds of time steps, only a small amount of representative instances are considered. These instances still represent moments of peak load, but instead of having all source nodes operational, some of them are assumed to be unavailable. Alternatively (or additionally), time steps can be used to include a discretised annual load curve as discussed in section 2.2.2.

Mathematically, this change leads to the introduction of the time step set  $T$ . The power flow variable thus becomes time-dependent. A new variable for the pipe capacity is then needed for calculating the pipe investment costs. A new binary parameter *availability* is introduced for each source vertex and time step. A value of 1 means that the corresponding source vertex is operational, while 0 stands for an outage (planned or unplanned). Figure 3.15 summarises the resulting input/output graph of the new model, with changes highlighted in bold (cf. figure 3.11 for comparison).

### 3.4.3 Example result

Before considering the full mathematical definition in detail, this section gives a behavioural description of the additional questions the new constraints allow to investigate. This is done by a small, artificial optimisation problem.




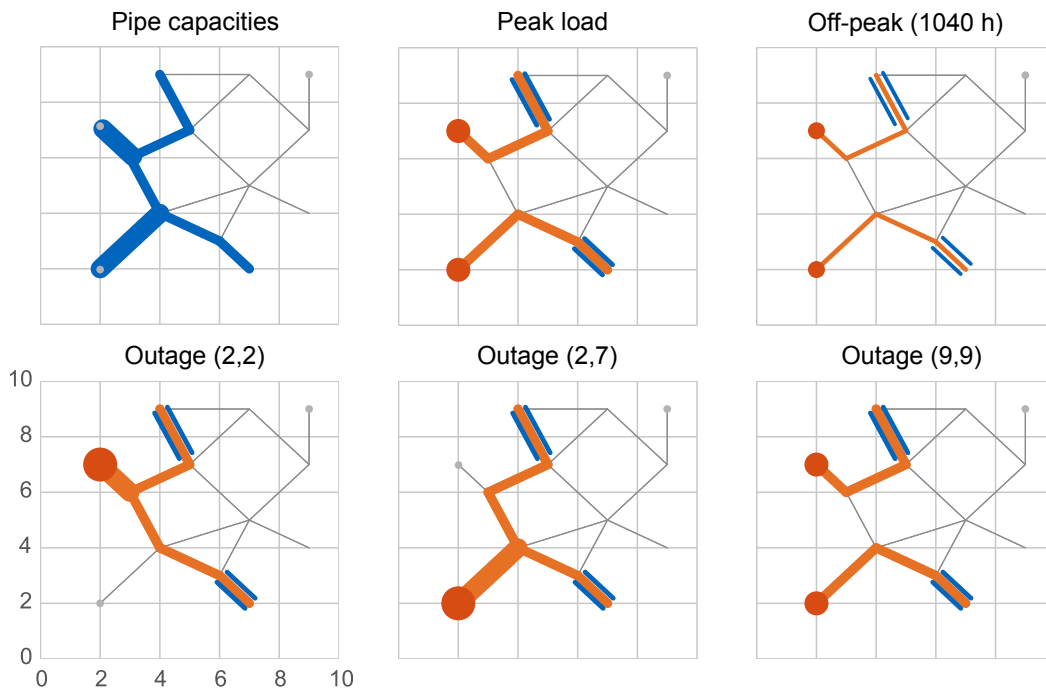
Figure 3.16 shows setup and result of the optimisation problem for a network with 13 vertices and 16 edges, defined on a grid with integer coordinates in the range from (1,1) to (9,9). The energy demand () is concentrated in two edges (5,7) to (4,9) and (6,3) to (7,2) and can be satisfied by three source vertices ( ) located at coordinates (2,2), (2,7) and (9,9).

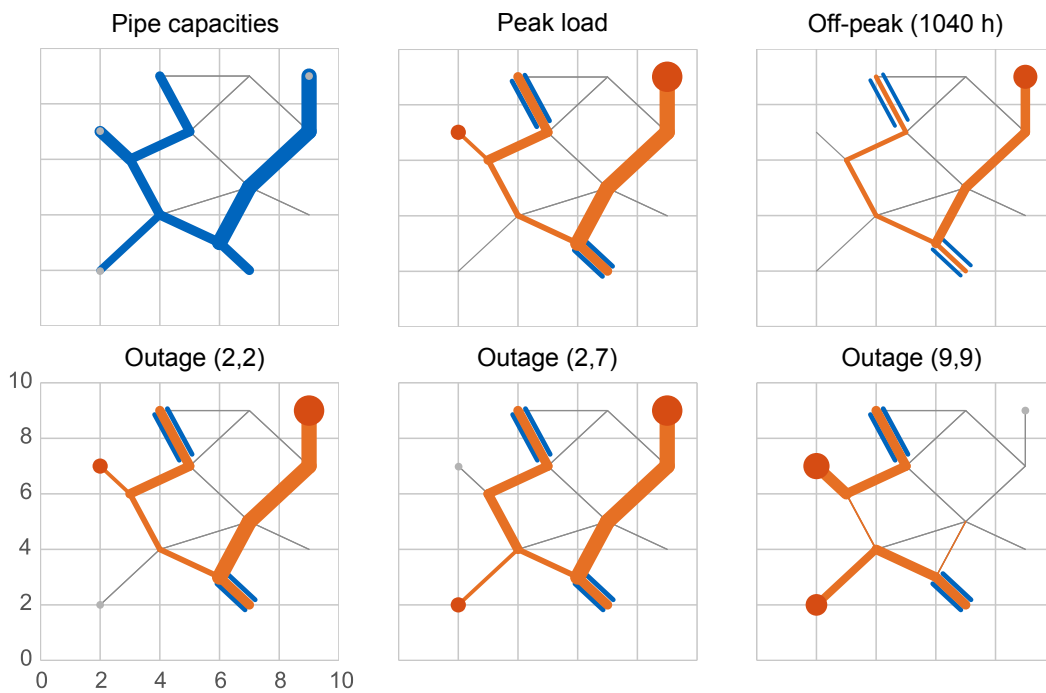


Figure 3.15: Input/output flow chart of model **dhmin** (seasonal). Differences to **DHMIN** in bold.










(a) Identical heat generation costs at all source vertices: only sources (2,2) and (2,7) are used.



(b) Source vertex (9,9) half generation costs, capacities of other sources reduced to less than total peak demand: source (9,9) becomes main supplier, other sources only used for outage replacement.

Figure 3.16: Example result with redundancy against any single source vertex outage. The first row shows pipe capacities (  ) and two least cost flow configurations (  ) for peak load (  ) and off-peak load with all three source locations (   ) operational. Each second row shows power flow configurations for peak load with one of the sources unavailable.

Costs and revenues are set so that a full demand satisfaction is the most economic solution. In order to ensure the reliability of any solution, three outage time steps are introduced, each one making one of the three source vertices unavailable. The capacity of each of the source vertices is sufficient to satisfy all loads, so without outages, only one of the sources would need to be connected to each load. However, in this example three *outage* time steps are introduced so that each of the three sources is not available for exactly one instance. Additionally, one off-peak time step models the duration of the heating period, while one peak load time step with all sources available shows the least cost heat generation configuration.

In the scenario depicted in figure 3.16(a), each of the source vertices has sufficient capacity to satisfy all loads on its own. Additionally, all three source vertices have identical heat generation costs. The optimal solution in this scenario only uses the two source vertices (2,2) and (2,7), but does not use source (9,9) at all, so no pipe capacity is connected to that vertex. With both used source vertices available, the shortest path from source to demand is used exclusively to reduce the transport losses. In the two outage time steps (2,2) and (2,7), the unaffected source vertex takes over the full supply task. Only for that purpose, the additional pipe capacity in edge (4,4) to (3,6) has been installed.

The lower half, figure 3.16(b), shows the result for a second scenario: in this scenario, the capacities of vertices (2,2) and (2,7) are reduced, so that one of the vertices can no longer satisfy all peak demand. Additionally the heat generation costs of vertex (9,9) are reduced by 50%. These changes result in source vertex (9,9) becoming the backbone of the whole network and supplying heat most of the time, except in case of its outage time step Outage (9,9). Pipe capacities are maximum from vertex (9,9) to the demand edge at (6,3). Interestingly, in outage time step (2,7), the most economical power flow includes heat generation from vertex (2,2), probably to reduce additional heat losses on the long distance from vertex (9,9).

### 3.4.4 Mathematical description

The presented model minimises the total costs  $\zeta$  of generating and distributing heat through a district heating network. The key result is the size and topology of the network, represented by the value of network capacity  $\bar{\pi}$  (vector of all individual arc capacities  $\bar{\pi}_{ij}$ ) in the optimal case:

$$\bar{\pi}^* = \underset{\bar{\pi}}{\operatorname{argmin}} \zeta$$

#### Sets

Like in the previous model, a district or city is represented as a graph of vertices and arcs. In most applications, this graph is usually derived from the street network. It should be derived in such a way to include all considerable locations for network pipes. The spatial resolution can be made as fine as required, but should rarely exceed the level of building blocks, i.e. a single street segment between two intersections.

This model, unlike the previous **DHMIN**, agrees in terminology with the rest of the thesis, by using the word *arc*  $a_{ij}$  to refer to a *directed* connection between two vertices  $v_i$  and  $v_j$ .

Let  $V$  be the set of vertices  $v_i$ , corresponding to connecting or terminating points of the graph. Set  $A$  of arcs then comprises ordered tuples of vertices  $a_{ij} = (v_i, v_j)$  with  $i \neq j$ .  $A$  is symmetric, that means either both or none of the pair  $a_{ij}$  and  $a_{ji}$  are elements of  $A$ . For readability, the subscript  $\circ_i$  is used to denote any parameter or variable that is defined over vertices  $v_i \in V$ , while  $\circ_{ij}$  is used to denote a quantity defined over arcs  $a_{ij} \in A$ .

The set  $V_0 \subseteq V$  defines so-called *source vertices*, which represent locations of possible heat sources. Usually, there is only a small number ( $\leq 10$ ) of source vertices.

The neighbour sets  $N_i$  of a vertex  $v_i$  are defined by the indices of all vertices that are connected to it by an arc:  $N_i = \{k \mid a_{ki} \in A\}$ .

Time is represented by a set  $T$  of discrete time steps  $t$ , which represent a small number of representative operational situations. This model is mathematically equivalent to the previous **DHMIN** if used with only a single time step, whose scaling parameter  $s_t$  is set to value 1 and whose time step weight is set to the length of the heating period  $w_t = T_{\text{flh}}$ . Please refer to the following paragraphs for a discussion of these two parameters.<sup>4</sup> The minimum viable number of time steps is two: one for peak demand with a duration of one to several hours, a second for the average annual load with a duration of the remaining year. A more comprehensive choice are three to four time steps: the third can refer to the all-year base load, while the fourth can be used to represent a common intermediate load level. Additional time steps need to be introduced if redundancy requirements are to be stated. Refer to the discussion of the parameter *availability* below for more details.

### Parameters

The numerical given facts for this model are grouped by their defining domains. First are vertex parameters, then arc parameters, then constant or global parameters, and finally time-dependent parameters.

**Vertices** Vertices have a single parameter: their maximum power capacity  $Q_i^{\text{max}}$ . It is the thermal output power given in kW for that location. All non-source vertices have this parameter set to 0.

**Arcs** Arcs have the three main attributes: their length  $l_{ij}$  (m), the thermal peak demand  $d_{ij}$  (kW) of their adjacent buildings and  $g_{ij}$  (binary), which states whether a pipe already exists in this arc.

As secondary parameter,  $C_{ij}^{\text{max}}$  indicates the maximum thermal power capacity (kW) in an arc, which can be derived from the maximum available pipe diameter. For arcs with existing pipes ( $g_{ij} = 1$ ), this value should be set according to the diameter of that existing pipe. As both arcs  $a_{ij}$  and  $a_{ji}$  refer to the same street segment, parameter values for both members are always identical.

**Global** Global parameters are technical and economic parameters that refer to the district heating system as a whole.

<sup>4</sup>Please note that  $T_{\text{flh}}$  is not used in model **dhmin**; it refers to the parameter from model **DHMIN**.

Economic parameters are all costs and revenues. The investment costs are split into a fixed part  $c_{\text{fix}}$  and a variable part  $c_{\text{var}}$ . The fixed part, given in €/m, captures all costs that are not dependent on the capacity of the pipe to be built, mainly earth works. The variable part, given in €/(kW m), refers to the capacity-dependent cost component of building a pipe. Both values must be tailored to the study area to compensate for differences in cost structure and availability of different pipe sizes. The same is true for the O&M cost parameter  $c_{\text{om}}$  in €/m. In contrast to the fixed investment cost term, it is also to be paid for existing pipes. The cost for generating heat  $c_{\text{heat}}$  is assumed to not depend on the source vertex.<sup>5</sup>

The letter  $c$  and a subscript denote *economic parameters*. These are investment costs for building the pipe network  $c_{\text{fix}}$  and  $c_{\text{var}}$ , maintenance  $c_{\text{om}}$ , costs of providing  $c_{\text{heat}}$  heat, and *revenue* for delivering  $c_{\text{rev}}$  heat to consumers. While  $c_{\text{fix}}$  contains the size-independent costs (mainly earth works),  $c_{\text{var}}$  contains costs that are dependent on the thermal capacity (diameter) of the pipe.

**Time-dependent** Parameters  $s_t$  and  $w_t$  represent scaling factor (dimensionless) and weight or duration (h) of a time step  $t$ . A value of  $s_t = 1$  refers to peak demand, while smaller values correspond to moments of partial load. Together, these two parameters encode a *discretised annual load curve*.

Redundancy requirements can be stated in the model by setting the binary availability parameter  $y_{it}$  to value 0 in certain time steps for  $1, 2, \dots$  source vertices. If one time step for each foreseen failure configuration is introduced, a full  $n - 1, n - 2, \dots$  failure safety (against heat source outages) can be enforced in the produced solution. Clarification: in the *conventional* time steps discussed above in paragraph time step set, the value  $y_{it} = 1$  is to be set for all source vertices  $v_i \in V_0$ .

## Variables

The main optimisation task is to find values for the *binary* decision variable  $\xi_{ij}$ . If its value is one, a pipe is built in the street segment corresponding to the arc  $a_{ij}$ . For each time step, the actual use of a given pipe is decided by setting the binary pipe usage decision variable  $\psi_{ijt}$ . If its value is one, the pipe in arc  $a_{ij}$  is used in direction from vertex  $v_i$  to vertex  $v_j$ . Consequently, there must be a power flow  $\pi_{ijt}^{\text{in}}$  into the pipe. Simultaneously, a value  $\xi_{ij} = 1$  requires that the demand  $d_{ij}$  of this arc has to be satisfied *at all times*. The power flow variable at the other end of the pipe is called  $\pi_{ijt}^{\text{out}}$ . Its value is determined by the ingoing power flow minus losses and demand.

The non-negative variable  $\rho_{it}$  represents thermal power output from a source vertex  $v_i \in V_0$  at time  $t$  in kW. It is limited in value by the source vertex capacity parameter  $Q_i^{\text{max}}$  and possibly by the availability parameter  $y_{it}$ . Table 3.13 summarises all model variables. The following line shows their domains, on which they are defined:

$$\zeta \in \mathbb{R} \quad \xi_{ij}, \psi_{ijt} \in \{0, 1\} \quad \bar{\pi}_{ij}, \pi_{ijt}^{\text{in}}, \pi_{ijt}^{\text{out}}, \rho_{it} \in \mathbb{R}_0^+.$$

<sup>5</sup>If the costs are different for different source vertices, this parameter must be made location dependent in equation (3.83).

Table 3.12: **dhmin** (seasonal) model parameters

Name	Unit	Description
$c_{\text{fix}}$	€/m	Fixed investment costs
$c_{\text{var}}$	€/(m kW)	Variable investment costs
$c_{\text{om}}$	€/(m a)	Operation & maintenance costs
$c_{\text{heat}}$	€/kWh	Heat generation costs
$c_{\text{rev}}$	€/kWh	<i>Revenue</i> for delivered heat
$w^{\text{fix}}$	kW/m	Fixed thermal losses
$w^{\text{var}}$	kW/(kW m)	Variable thermal losses
$b$	%	Concurrence effect
$q$	%	Connect quota
$u$	1/a	Annuity factor for investment costs
$l_{ij}$	m	Arc length
$d_{ij}$	kW	Arc peak demand
$g_{ij}$	—	Existence of a pipe (1=yes, 0=no)
$C_{ij}^{\text{max}}$	kW	Maximum pipe capacity
$Q_i^{\text{max}}$	kW	Source vertex capacity
$s_t$	1	scaling factor
$w_t$	h	weight/duration
$y_{it}$	—	availability (1=yes, 0=no)

Table 3.13: **dhmin** (seasonal) model variables

Name	Unit	Description
$\zeta$	€	Total system cost (Inv, O&M, Rev, Heat)
$\xi_{ij}$	—	Binary decision variable: 1 = build pipe
$\psi_{ijt}$	—	Binary decision variable: 1 = use pipe
$\bar{\pi}_{ij}$	kW	Thermal power flow capacity into arc $a_{ij}$
$\pi_{ijt}^{\text{in}}$	kW	Thermal power flow from $v_i$ into arc $a_{ij}$
$\pi_{ijt}^{\text{out}}$	kW	Thermal power flow out of arc $a_{ij}$ into $v_j$
$\rho_{it}$	kW	Heat generation power in source vertex $v_i$

## Equations

Equations fall into two categories: the first is the cost function, whose value is to be minimised. The second are constraints, that codify all the previously discussed rules in mathematical form. By that, they define the region of feasible solutions, under which the solver selects a (close to) cost optimal solution.

**Cost function** The value *total cost*  $\zeta$  is the sum of costs for supplying heat, minus revenue for that heat. Costs occur for building the network (annualised investment), maintenance costs and heat generation costs. The following *derived parameters* are introduced to shorten the definition of the cost function, whose definition is given below:  $k_{ij}^{\text{fix}}$  (€/a) and  $k_{ij}^{\text{var}}$  (€/(kW a)) cover network costs (investment, O&M).  $k_{ij}^{\text{heat}}$  (€/kWh) represents heat generation costs, while

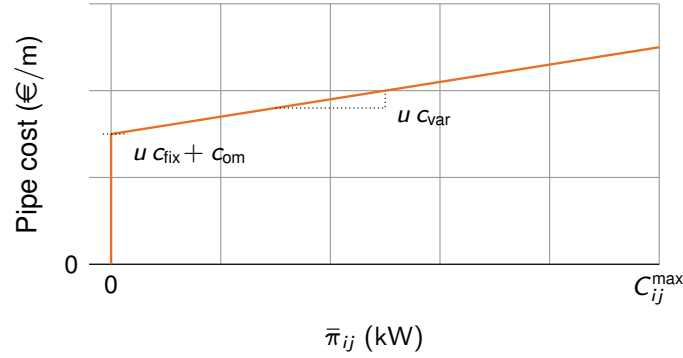


Figure 3.17: Investment and O&M costs for a single arc  $a_{ij}$  over pipe capacity  $\bar{\pi}_{ij}$

$r_{ij}^{\text{heat}}$  (€/h) represents revenue for delivered heat. With these, the cost function is equal to

$$\zeta = \sum_{a_{ij} \in A} \left( k_{ij}^{\text{fix}} \xi_{ij} + k_{ij}^{\text{var}} \bar{\pi}_{ij} \right) + \sum_{\substack{v_i \in V \\ t \in T}} k_i^{\text{heat}} w_t \rho_{it} - \sum_{\substack{a_{ij} \in A \\ t \in T}} r_{ij}^{\text{heat}} s_t w_t \xi_{ij}. \quad (3.79)$$

The first summand forms a piece-wise linear function that is depicted in figure 3.17. Like in model **DHMIN**, there is an unsteady jump from zero to the fixed cost  $u c_{\text{fix}} + c_{\text{om}}$  for any positive value of  $\bar{\pi}_{ij}$  for any non-existing ( $g_{ij} = 0$ ) pipe. The link between the binary decision variable  $\xi_{ij}$  and the continuous pipe capacity variable  $\bar{\pi}_{ij}$  is enforced through equations (3.85), (3.91) and (3.89) below.

These are the definitions for the four derived parameters used in the cost equation above. The division by value 2, common to all three arc parameters, compensates for the double-representation of one street segment  $\{v_i, v_j\}$  as a pair two directed arcs  $a_{ij}, a_{ji}$ . More about this issue can be found at the explanation of the symmetry constraints (3.92) and (3.93) below. The heating cost parameter  $k_i^{\text{heat}}$  is divided by the concurrence effect parameter  $b$ , to remove the load reduction effect that is introduced in equation (3.87). In other words, the power flow that needs to be satisfied from a source vertex is lower by  $b$  than the *energy* flow that needs to be delivered. The division by  $b$  removes that difference in terms of costs.

$$\forall a_{ij} \in A: \quad k_{ij}^{\text{fix}} = [c_{\text{fix}} l_{ij} u (1 - g_{ij}) + c_{\text{om}} l_{ij}] / 2 \quad (3.80)$$

$$\forall a_{ij} \in A: \quad k_{ij}^{\text{var}} = [c_{\text{var}} l_{ij} u (1 - g_{ij})] / 2 \quad (3.81)$$

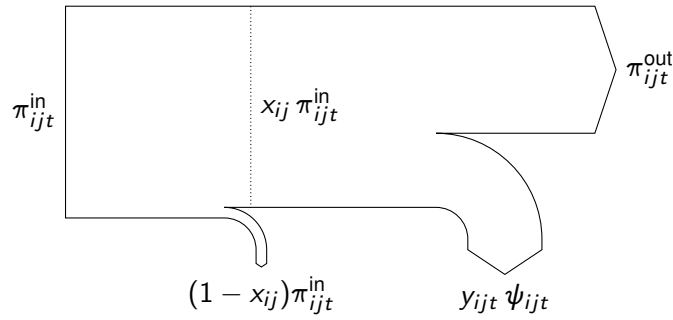
$$\forall v_i \in V: \quad k_i^{\text{heat}} = c_{\text{heat}} / b \quad (3.82)$$

$$\forall a_{ij} \in A: \quad r_{ij}^{\text{heat}} = c_{\text{rev}} d_{ij} q / 2 \quad (3.83)$$

**Constraints** The constraints formalise all physical laws and technical rules that need to be satisfied by a feasible solution. The first constraint concerns energy conservation:

$$\forall v_i \in V, t \in T: \quad \sum_{n \in N_i} (\pi_{int}^{\text{in}} - \pi_{nit}^{\text{out}}) \leq \rho_{it} \quad (3.84)$$

This inequality is thus a relaxed version of the law of energy conservation. Relaxed, because it allows for power to vanish at any vertex. As power does not come for free, any solution returned by the solver will satisfy this constraint to equality. For all non source vertices (i.e.

Figure 3.18: Sankey diagram of equation *demand satisfaction* (3.85)

$\rho_{it} = 0$ ), the difference between outgoing and incoming power must be smaller than or equal to zero. In source vertices, a positive difference may remain when it is met by an equal amount of input power  $\rho_{it}$  into the network at that vertex.

*Demand satisfaction* is the main *arc constraint*. It creates the logical connection between the discrete pipe usage decision  $\psi_{ijt}$  and the demand satisfaction. Like the cost equation, it relies on two derived parameters  $x_{ij}$  and  $y_{ijt}$ . This constraint is also graphically explained in figure 3.18.

$$\forall a_{ij} \in A, t \in T: \quad x_{ij} \pi_{ijt}^{\text{in}} - \pi_{ijt}^{\text{out}} = y_{ijt} \psi_{ijt} \quad (3.85)$$

Parameter  $x_{ij}$  refers to losses  $w^{\text{var}}$  that are proportional to the amount of power flow into the arc  $a_{ij}$ . Parameter  $y_{ijt}$  refers to fixed thermal losses  $w^{\text{fix}}$  that only depend on the pipe length, not on the power flow, and the time-dependent demand  $d_{ij} \cdot s_t$ , that is further reduced by the concurrence effect  $b$  and the connect quota parameter  $q$ . In notation:

$$\forall a_{ij} \in A: \quad x_{ij} = 1 - l_{ij} w^{\text{var}} \quad (3.86)$$

$$\forall a_{ij} \in A, t \in T: \quad y_{ijt} = b q d_{ij} s_t + l_{ij} w^{\text{fix}} \quad (3.87)$$

*Pipe capacity* is a technical constraint that limits the power flow through a pipe by the built capacity. This is accomplished by first limiting the ingoing power flow into a pipe  $\pi_{ijt}^{\text{in}}$  by the built pipe capacity  $\bar{\pi}_{ij}$ . The second constraint is required to set the pipe usage decision variable  $\psi_{ijt}$ . Its value is forced to 1 if a flow in direction from  $v_i$  to  $v_j$  at time step  $t$  is required. The reason for this seemingly artificial constraint is explained in the following constraint.

$$\forall a_{ij} \in A, t \in T: \quad \pi_{ijt}^{\text{in}} \leq \bar{\pi}_{ij} \quad (3.88)$$

$$\forall a_{ij} \in A, t \in T: \quad \pi_{ijt}^{\text{in}} \leq \psi_{ijt} C_{ij}^{\text{max}} \quad (3.89)$$

*Unidirectionality* of the power flow  $\pi_{ijt}^{\text{in}}$  is required, as otherwise the solver would happily use a pipe's capacity in both directions simultaneously, which would not be physically possible. Therefore, the pipe usage decision variable  $\psi_{ijt}$  may only have the value 1 in one direction at any given time step  $t$ . This way, the direction of flow may change between time steps, but any given pipe may only be used in one direction at a given time.

$$\forall a_{ij} \in A, t \in T: \quad \psi_{ijt} + \psi_{jit} \leq 1 \quad (3.90)$$

*Build capacity* limits the pipe capacity variables to the maximum available diameter, whose capacity in kW is represented by parameter  $C_{ij}^{\max}$ . At the same time, this constraint sets the value of the building decision variable  $\xi_{ij}$ , if a non-zero pipe capacity  $\bar{p}_{ij}$  is needed.

$$\forall a_{ij} \in A: \quad \bar{\pi}_{ij} \leq \xi_{ij} C_{ij}^{\max} \quad (3.91)$$

*Symmetry of building decision* are two constraints that enforce that the mathematically independent arcs  $a_{ij}$  and  $a_{ji}$  must have identical variable values for their pipe capacity  $\bar{\pi}_{ij}$  and building decision  $\xi_{ij}$ . This way, a pipe can then be used in both directions at different time steps.

$$\forall a_{ij} \in A: \quad \xi_{ij} = \xi_{ji} \quad (3.92)$$

$$\forall a_{ij} \in A: \quad \bar{\pi}_{ij} = \bar{\pi}_{ji} \quad (3.93)$$

*Use if built* is the last piece in the puzzle to link the building decision  $\xi_{ij}$  to the pipe usage decision  $\psi_{ijt}$ . Up to this point, nothing in the formulation requires that the heat demand of a customer is satisfied *at all times*. This constraint enforces exactly that, by requiring that the sum of  $\psi_{ijt} + \psi_{jit}$  is greater or equal to 1, if a pipe is built along that arc. Otherwise, the condition should not be enforced. This is done by calculating the expression  $(\xi_{ij} + \xi_{ji})/2$ . It is equal to 1 if  $\xi_{ij} = 1$ , as equation (3.92) requires  $\xi_{ij} = \xi_{ji}$ . Otherwise, its value is equal to 0. In other words: if an arc has a pipe, its demand must always be satisfied by a power flow from any of its two sides.

$$\forall a_{ij} \in A, t \in T: \quad \psi_{ijt} + \psi_{jit} \geq (\xi_{ij} + \xi_{ji})/2 \quad (3.94)$$

*Source vertices* is the last constraint. It limits the source vertex power flow  $\rho_{it}$  by its maximum allowed capacity  $Q_i^{\max}$ . The availability parameter  $y_{it}$  *usually* has value 1, except for special time steps in which the source vertex  $v_i$  is made unavailable by setting  $y_{it} = 0$ :

$$\forall v_i \in V, t \in T: \quad \rho_{it} \leq y_{it} Q_i^{\max} \quad (3.95)$$

Equations (3.79) to (3.95) together define the linear mixed-integer program **dhmin**, whose optimal solution is the desired vector of pipe capacities  $\bar{\pi}^*$ .



## 3.5 rivus

### 3.5.1 Motivation for model development

When using **DHMIN** for optimal district heating planning in Frankfurt, the good scaling behaviour for big problems with up to  $10^4$  vertices/edges became apparent. The linearised fixed cost formulation of the investment costs for network capacities are able to represent the real-world planning question at hand: What is – in principal – the optimal balance between network expansion and network concentration?

The remaining research question now is: how would an optimal district heating network look in a fully optimal network planning spanning all energy carriers? Figure 3.19 summarises the key model aspects. The focus lies – like in the previous model **dhmin** – on the spatial distribution of energy demand, but augmented with the possibility to discern between long duration base loads and short duration peak loads.

### 3.5.2 The main idea

**rivus** finds the cost-minimal distribution network to satisfy a set of demands for energy carriers. Here, these energy carriers are electricity and heat, but could for example include cooling, if feasibility of district cooling should be investigated. Like in model **DHMIN**, these demands are given as peak loads and discretised load duration curves. These peak loads are deduced from (estimated) building gross floor areas and then aggregated to street segments (edges) that form a network through nodes (vertices). Energy enters this network at pre-determined source vertices. In this thesis, these are given by the location of close electricity substations (electricity source vertex) and pipelines or gas terminals (natural gas source vertex). Source vertices could include geothermal extraction points, biomass or waste availability. Figure 3.20 graphically summarises this idea. Two source vertices, one for electricity (yellow diamond), one for gas (brown) and a distribution of peak demands (left column), are given. The maps in the right column are the optimisation results: location and size of conversion processes, as well as size and topology of all distribution networks. Depending on the available technology options, different networks and/or conversion processes are preferred.

Power can be converted from one carrier (called *commodity*) through conversion processes, simply called *processes* in the following. A process consumes an arbitrary number of input commodities in fixed proportions (given by parameter *input ratio*) of input commodities and produces an arbitrary number of output commodities in fixed proportions (given by parameter *output ratio*). Some commodities – here: electricity, heat and natural gas – are deemed transportable and thus may form a network of flows.

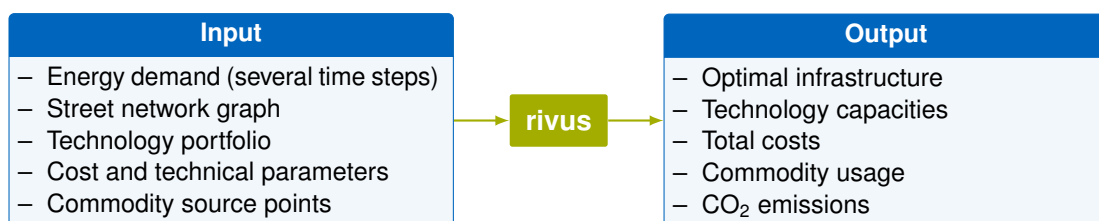


Figure 3.19: Input/output flow chart of model **rivus**

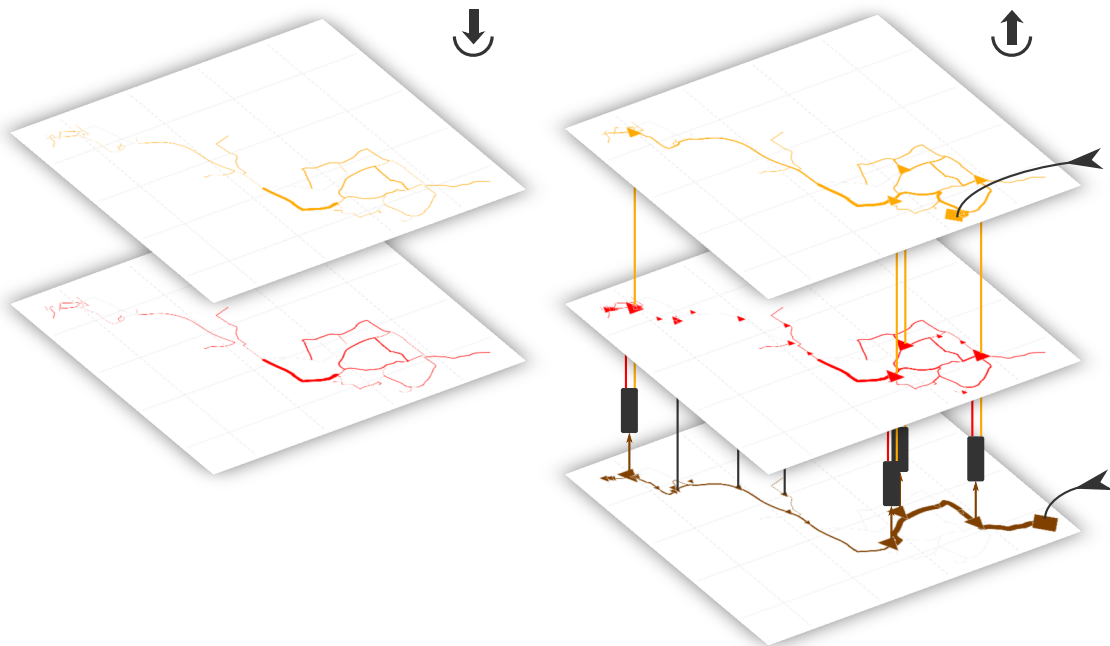


Figure 3.20: Street network of peak demands (left) and transport capacities (right) for commodities electricity, heat and natural gas (yellow, red, and brown). Two black arrows from the right mark the two source vertices.

Unlike in model **dhmin**, this model *must satisfy all* given demands. The result is a cost-minimal energy distribution network and set of installed process capacities, either centralised or distributed.

### 3.5.3 Mathematical description

This section defines all collections of model entities (sets), pre-determined numerical values (parameters) and the possible model decisions (variables), the rules to obey (constraints) and which quantity to minimise (the cost function).

#### Sets

Table 3.14 lists all model sets. Like in model **dhmin**, the street network is represented through a graph of vertices  $v \in V$  and undirected edges  $e \in E$ . For each edge  $\{v_1, v_2\} \in E$ , the pair  $(v_1, v_2)$  and  $(v_2, v_1)$  form the set of directed arcs  $a \in A$  of the graph.

The first model entity are commodities  $C$ . They mainly represent energy carriers, but are more general: they are any inputs and outputs of energy conversion processes, like fuels (natural gas), forms of energy (electricity, heat) or chemicals ( $\text{CO}_2$ ). To differentiate between those types, commodity type subsets are introduced. In contrast to model **urbs**, these type assignments are not mutually exclusive, but are assigned based on certain parameter values. Demand commodities  $C_d \subseteq C$  are those for which a non-zero peak demand occurs anywhere. Source commodities  $C_s \subseteq C$  are those which have a source vertex. Transportable commodities  $C_t \subseteq C$  are defined by having a positive maximum transport capacity. In the presented case

Table 3.14: rivus model sets

Set	Subsets	Description
$c \in C$	$C_d, C_s, C_t, C_e$	Commodities, (demand, source, transportable, limited) subsets
$p \in P$	$H$	Processes, Hubs
$v \in V$	—	Vertices
$e \in E$	—	Edge, unordered pair of vertices
$a \in A$	$A_v^p, A_v^s$	Arcs, incoming/outgoing arcs in vertex $v$
$t \in T$		Time steps, modelled time steps, initial/final time step

studies, electricity and heat are demand commodities, while electricity and natural gas are source commodities. All three commodities are transportable. Environmental commodities finally are those that have a maximum possible net generation set to a finite value. In the presented case study, CO<sub>2</sub> has this property set.

Then there is the set of processes  $P$ . It represents all means of converting energy of one form (commodity) to another. For each process, there are two sets of input and output commodities  $C_p^{\text{in}}$  and  $C_p^{\text{out}}$ . For example, a gas-fired district heating plant could have the input commodity set  $C_p^{\text{in}} = \{\text{Gas}\}$  and output commodity set  $C_p^{\text{out}} = \{\text{Electricity, Heat, CO}_2\}$ .

Hub processes  $H \subseteq P$  are defined on edges and represent small-scale energy conversion processes. These include all technologies that are installed within buildings, like boilers or small-scale CHP units. Mathematically, they are the subset of processes which satisfy the following four conditions:

1. Capacity-independent investment costs are zero:  $k_p^{\text{inv,fix}} = 0$
2. Minimum capacity is zero:  $K_p = 0$
3. Only one input commodity:  $|C_p^{\text{in}}| = 1$
4. Input ratio for that input commodity is one:  $r_{pc}^{\text{in}} = 1$

Consequently, hubs still have two associated sets of input and output commodities  $C_h^{\text{in}}$  and  $C_h^{\text{out}}$ , but there is no need to have an input ratio parameter  $r_{hc}^{\text{in}}$  (its value must be 1 due to conditions 3. and 4.), which is why it will be omitted in the following. The reason for the first two conditions is to allow hubs to be fully linear, whereas the fixed investment costs for central processes require binary variables.<sup>6</sup> Conditions three and four are not required to keep the formulations of hubs linear, but are rather a design decision to not allow too complex processes to be built both in vertices and edges.

A small set of time steps  $T$  is used to represent several time periods. By design, less than a dozen time steps are used to represent a full year by representative moments characterised by distinct demand situations. These at least include moments *peak load* and *base load*.

## Parameters

Table 3.15 lists all technical model parameters. The main parameter of the whole model is the demand  $d_{ec}$ . It represents the peak demand (kW) for commodity  $c$  in edge  $e$ . It is derived per

<sup>6</sup>To improve the scaling behaviour of rivus for large problems, one could limit processes to a subset  $V_p$  of possible locations, either per process or for all processes. For the case studies presented in this thesis, this step was not necessary.

edge  $e$  by adding all building peak demands  $d_{bc}$  along that edge:

$$d_{ec} = \sum_{b \in B_e} d_{bc} \quad (3.96)$$

Here,  $B_e$  is the set of buildings closest to edge  $e$ . To represent the actual demand in each time step  $t$ , these peak demands are multiplied with a commodity-dependent scaling factor  $s_{ct}$ . This product  $d_{ec}s_{ct}$  forms the right side of the edge constraint (3.101). Refer to section 4.4.1 for the procedure used to calculate  $d_{bc}$ , as it is independent from the model. Apart from the scaling factor  $s_{ct}$ , there is a second time-dependent quantity, the *time step weight*  $w_t$ . It refers to the length of that time step. By convention, the sum over all time steps is equal to a common (i.e. non-leap) year:  $\sum_{t \in T} w_t = 8760$ . These weights are used for scaling the variable costs for each time step to the appropriate portion of the year, as all costs (investment, fixed, variable) are scaled to a year.

Next are the technical parameters that govern the properties of the distribution networks. The edge length  $l_e$  is the real (i.e. not line-of-sight) length of edge  $e$  in metres. It is used to scale both the costs and the losses of a pipe/cable. Speaking of losses, they are governed by parameters  $w_c^{\text{fix}}$  (capacity-independent) and  $w_c^{\text{var}}$  (capacity-dependent). Note that there is no set of transmission processes like in model **urbs**. In other words, it is assumed that only a single technology is used per commodity.<sup>7</sup> The capacity limit  $\bar{K}_c$  (kW) is a per-edge upper bound for the transmission lines. Practically, it represents the biggest feasible pipe or cable diameter that could be used in a given study area. The parameter  $\bar{K}_{vc}$  (kW) is the source vertex capacity limit. It describes how much power may be pulled from the source at vertex  $v$  of commodity  $c$ , while paying only variable costs

*Processes* and *hubs* share the same data input and thus share similar parameters. Both have an upper bound for their allowed capacity  $\bar{K}_p, \bar{K}_h$ . Processes may also have a lower bound for their capacity  $\underline{K}_p$ , which is enforced wherever the building decision variable  $\phi_{vp}$  has value 1. Hubs may not have this feature to stay fully linear. Both processes and hubs have the output ratio parameters  $r_{pc}^{\text{out}}, r_{hc}^{\text{out}}$ , which are defined for all commodities in the respective commodity output sets  $C_p^{\text{out}}, C_h^{\text{out}}$ . Processes also have input ratios  $r_{pc}^{\text{in}}$  for their input commodities  $C_p^{\text{in}}$ . Hubs do not require these two because of their defining restrictions (cf. previous section **Sets**).

Table 3.16 summarises the economic parameters. *Economic* are the last group of parameters. They are defined for the transmission (and use) of commodities  $k_c$ , conversion in hubs  $k_h$  and processes  $k_p$ . They are categorised in investment, fixed and variable costs. The investment costs are represented by the two values  $k^{\text{inv,fix}}$  and  $k^{\text{inv,var}}$  for capacity-independent and capacity-dependent cost components. The fixed investment cost component is modelled using a binary decision variables  $\xi_{ec}$  for transmission and  $\phi_{vp}$  for processes. Like in the other models, fixed costs  $k^{\text{fix}}$  represent all operation-independent, but (roughly) capacity-dependent costs like maintenance. Variable costs for commodities  $k_c^{\text{var}}$  represent their fuel price, so to speak. They can be obtained for that price at all source vertices in unlimited quantity, only limited in power by the vertex capacity. Variable costs for processes  $k_p^{\text{var}}$  and hubs  $k_h^{\text{var}}$  finally represent all non-fuel operation-dependent costs like wear and tear.

<sup>7</sup>For large case studies, network levels like different gas pressures, could be represented by using one commodity per pressure level and conversion processes for substations. Introducing transmission technologies like set  $F$  in **urbs** could implemented alternatively.

Table 3.15: **rivus** technical model parameters

Name	Unit	Description
$w_t$	h	Time step weight
$s_{ct}$	—	Peak demand scaling factor
$d_{ec}$	kW	Peak power demand
$l_e$	m	Edge length
$w_c^{\text{fix}}$	kW/m	Fixed transport losses
$w_c^{\text{var}}$	kW/(kW m)	Variable transport losses
$\bar{K}_c$	kW	Upper bound transport capacity
$\bar{K}_{vc}$	kW	Upper bound source vertex capacity
$\underline{K}_p, \bar{K}_p$	kW	Lower and upper bound process capacity
$r_{pc}^{\text{in}}, r_{pc}^{\text{out}}$	—	Process input/output ratio
$\bar{K}_{ch}$	kW	Upper bound hub capacity
$r_{hc}^{\text{out}}$	—	Hub output ratio

Table 3.16: **rivus** economic model parameters

$k_c^{\text{inv,fix}}$	€/m	Capacity-independent transport investment costs
$k_c^{\text{inv,var}}$	€/(kW m)	Capacity-dependent transport investment costs
$k_c^{\text{fix}}$	€/(kW m)	Capacity-dependent transport maintenance costs
$k_c^{\text{var}}$	€/kWh	Commodity “fuel” costs
$k_h^{\text{inv}}$	€/kW	Hub capacity investment costs
$k_h^{\text{fix}}$	€/kW	Hub capacity fixed costs
$k_h^{\text{var}}$	€/kWh	Hub usage variable costs
$k_p^{\text{inv,fix}}$	€	Capacity-independent process investment costs
$k_p^{\text{inv,var}}$	€/kW	Capacity-dependent process investment costs
$k_p^{\text{fix}}$	€/kW	Capacity-dependent process fixed costs
$k_p^{\text{var}}$	€/kWh	Process usage variable costs

## Variables

Table 3.17 summarises all model variables. They are grouped by the sets they are defined over. Vertices are locations to source vertices and processes. In edges, demand, transmission and hubs are located. The last group are four cost variables.

**Vertices** The source stream variable  $\rho_{vct}$  (kW) represents the flow of a commodity  $c \in C_s$  from a source vertex  $v$  at time  $t$ . It is expected that the number of source vertices per commodity is small compared to the number of all vertices  $|V|$ .

**Arcs & Edges** The key difference to model **urbs** are the properties of the transmission processes along edges. The supply variable  $\sigma_{ect}$  (kW) is the amount of power that is drawn from the power flow *through* that edge to supply the local demand/hubs. This available power flow is represented by the arc incoming flow variable  $\pi_{act}^{\text{in}}$  (kW). What remains of the flow at

the end of the arc is contained in the outgoing flow variable  $\pi_{act}^{out}$  (kW). To limit the use of an arc to only one direction per time step, the binary decision variable  $\psi_{act}$  is needed. The capacity for power flow in an edge is represented by variable  $\kappa_{ec}$  (kW). The binary building decision of a transmission cable/pipeline for a commodity in an edge is called  $\xi_{ec}$ .

**Hubs** The capacity  $\kappa_{eh}$  (kW) of a hub process  $h$  in an edge  $e$  determines its cost and limits its maximum activity. The activity or throughput  $\epsilon_{eht}$  (kW) of a hub process  $h$  in an edge  $e$  at a given time  $t$  determines the output via the output ratio parameter  $r_{hc}^{out}$  as mentioned before.

**Processes** Similarly, the capacity  $\kappa_{vp}$  (kW) of a process  $p$  in a vertex  $v$  determines its cost and limits its maximum activity. For the capacity-independent investment costs, the binary variable  $\phi_{vp}$  represents the building decision of that process  $p$  in vertex  $v$ . The activity of the process is represented by the throughput variable  $\tau_{vpt}$  (kW). Via the input and output ratio parameters, the variables for input  $\epsilon_{vpc}^{in}$  and output  $\epsilon_{vpc}^{out}$  are linked with that throughput, cf. equations (3.115) and (3.116).

**Costs** All system costs are represented by variable  $\zeta$ . Costs by type are  $\zeta_{inv}$ ,  $\zeta_{fix}$ ,  $\zeta_{var}$ . Investment costs occur for capacities of hubs and processes, as well as for transport capacities in edges. Fixed costs are operation-independent but capacity-dependent, like maintenance. Variable costs are due for activity of hubs and processes, and for supply of commodities with source vertices. For simplicity, this model does not include annuity factors in the model equations. Instead, the cost parameters  $k^{inv}$  are assumed to be already annualised.

## Equations

**Cost function** The objective function of the optimisation problem is to reduce the total costs  $\zeta$  for satisfying the given energy demand. The costs consist of three parts:

Investment costs  $\zeta_{inv}$  are to be paid for all capacities in the model. Process capacities  $\kappa_{vp}$  may have a fixed investment cost part that is triggered by the building decision variable  $\phi_{vp}$ . Hub capacities  $\kappa_{eh}$  are similar but without the possibility for a fixed cost part. Energy transmission technologies are priced by length  $l_e$  of the edge along which they are installed. Like processes, they have a variable investment cost part proportional to their transmission capacity  $\kappa_{ec}$  and a fixed investment triggered by their building decision variable  $\xi_{ec}$ . Fixed costs  $\zeta_{fix}$  are proportional to the capacities for processes, hub processes and transmission.

Variable costs  $\zeta_{var}$  occur for the activity of hub processes, central processes and the procurement of commodities at source vertices. In the previous models, this term was separate in fuel costs. The hub process activity is measured by input  $\epsilon_{eht}$ . Central conversion processes are modelled by their throughput  $\tau_{vpt}$ . Source vertices can provide supply of a commodity through source term  $\rho_{vct}$ . These costs are weighted by time step weight  $w_t$  and summed to an annual value.

$$\zeta = \zeta_{inv} + \zeta_{fix} + \zeta_{var} \quad (3.97)$$

Table 3.17: rivus model variables

Name	Unit	Range	Description
$\rho_{vct}$	kW	$\mathbb{R}_0^+$	Commodity source power flow
$\pi_{act}^{in}$	kW	$\mathbb{R}_0^+$	Commodity flow into arc
$\pi_{act}^{out}$	kW	$\mathbb{R}_0^+$	Commodity flow out of arc
$\psi_{act}$	—	$\mathbb{Z}_1$	1 if arc is used at $t$ , 0 else
$\sigma_{ect}$	kW	$\mathbb{R}_0^+$	Supply in edge at time
$\kappa_{ec}$	kW	$\mathbb{R}_0^+$	Power flow capacity
$\xi_{ec}$	—	$\mathbb{Z}_1$	1 if edge is used anytime, 0 else
$\kappa_{eh}$	kW	$\mathbb{R}_0^+$	Hub process capacity
$\epsilon_{eht}$	kW	$\mathbb{R}_0^+$	Hub process activity
$\kappa_{vp}$	kW	$\mathbb{R}_0^+$	Process throughput capacity
$\phi_{vp}$	—	$\mathbb{Z}_1$	1 if process has non-zero capacity, 0 else
$\tau_{vpt}$	kW	$\mathbb{R}_0^+$	Process power throughput
$\epsilon_{vpct}^{in}$	kW	$\mathbb{R}_0^+$	Process power input flow
$\epsilon_{vpct}^{out}$	kW	$\mathbb{R}_0^+$	Process power output flow
$\zeta$	€	$\mathbb{R}$	Total system costs
$\zeta_{inv}$	€	$\mathbb{R}$	Investment costs
$\zeta_{fix}$	€	$\mathbb{R}$	Fixed costs (e.g. maintenance)
$\zeta_{var}$	€	$\mathbb{R}$	Variable costs (activity dependent)

$$\begin{aligned} \zeta_{inv} = & \sum_{\substack{v \in V \\ p \in P}} \left( \kappa_{vp} k_p^{inv,var} + \phi_{vp} k_p^{inv,fix} \right) + \sum_{\substack{e \in E \\ h \in H}} \kappa_{eh} k_h^{inv} \\ & + \sum_{\substack{e \in E \\ c \in C_t}} l_e \left( \kappa_{ec} k_c^{inv,var} + \xi_{ec} k_c^{inv,fix} \right) \end{aligned} \quad (3.98)$$

$$\zeta_{fix} = \sum_{\substack{v \in V \\ p \in P}} \kappa_{vp} k_p^{fix} + \sum_{\substack{e \in E \\ h \in H}} \kappa_{eh} k_h^{fix} + \sum_{\substack{e \in E \\ c \in C_t}} l_e \kappa_{ec} k_c^{fix} \quad (3.99)$$

$$\zeta_{var} = \sum_{t \in T} w_t \left[ \sum_{\substack{e \in E \\ h \in H}} \epsilon_{eht} k_h^{var} + \sum_{\substack{v \in V \\ p \in P}} \tau_{vpt} k_p^{var} + \sum_{\substack{v \in V_0 \\ c \in C_s}} \rho_{vct} k_c^{var} \right] \quad (3.100)$$

**Edges/arcs** The the first group of constraints describe the behaviour of the flows through the graph. Equation *peak satisfaction* is the the equation that assures that the instantaneous demand  $d_{ec} s_{ct}$  is always satisfied. The first summand  $\sigma_{ect}$  is the supply of commodity  $c$  provided by an ingoing power flow from either direction, as defined in the following constraint (3.102). The two sums add to or take from that supply either by output or input from hubs. Note that the output ratio  $r_{hc}^{out}$  is multiplied only to the output side, because hubs must have an input ratio of one:

$$\forall e \in E, c \in C_d, t \in T: \quad \sigma_{ect} + \sum_{h|c \in C_h^{out}} \epsilon_{eht} r_{hc}^{out} - \sum_{h|c \in C_h^{in}} \epsilon_{eht} \geq d_{ec} s_{ct} \quad (3.101)$$

Equation *edge* is linking energy flows  $\pi$  through arcs with the supply term  $\sigma$  in edges. While the flow is directional and potentially different per time step and commodity, the supply term is not directed. The supply is determined by the difference between incoming and outgoing power flow into one of the two consisting arcs  $a$  and  $a'$ . As only one of  $\psi_{act}$  and  $\psi_{a'ct}$  may have a non-zero value per time step, power flow is also either in direction  $a$  or  $a'$ , but never simultaneous in both directions. The formulation as an inequality instead of an equality is believed to be more friendly to the solver, but no thorough tests were performed. In optimality, the constraint is always active, i.e. equality is assured.

$$\forall e \in E, c \in C_t, t \in T: \quad \sigma_{ect} \leq (1 - w_{ec}^{\text{var}} l_e)(\pi_{act}^{\text{in}} + \pi_{a'ct}^{\text{in}}) - (\pi_{act}^{\text{out}} + \pi_{a'ct}^{\text{out}}) - (\psi_{act} + \psi_{a'ct}) w_{ec}^{\text{fix}} l_e \quad (3.102)$$

*Arc flow by capacity* limits the power flow per arc to the edge capacity variable. The notation  $a \in e$ , though not strictly correct, means that arc  $a$  and edge  $e$  must connect the same vertices. Equation (2.26) gives a more formal definition of that intuition:

$$\forall a \in A, c \in C_t, t \in T, a \in e: \quad \pi_{act}^{\text{in}} \leq \kappa_{ec} \quad (3.103)$$

*Arc flow unidirectionality* assures that only one direction is used per time step, using the additional arc usage decision variable  $\psi_{act}$ :

$$\forall a \in A, c \in C_t, t \in T: \quad \pi_{act}^{\text{in}} \leq \psi_{act} \bar{K}_c \quad (3.104)$$

$$\forall a \in A, c \in C_t, t \in T: \quad \psi_{act} + \psi_{a'ct} \leq 1 \quad (3.105)$$

*Edge capacity* finally ensures that the built capacity  $\kappa_{ec}$  in an edge does not exceed the capacity limit  $\bar{K}_c$  allowed for that commodity. As a side effect, the building decision variable  $\xi_{ec}$  is forced to have the value 1 in case the edge capacity is non-zero, leading to the fixed investment costs  $k_c^{\text{inv,fix}}$  occurring in the cost function.

$$\forall e \in E, c \in C_t: \quad \kappa_{ec} \leq \xi_{ec} \bar{K}_c \quad (3.106)$$

**Hubs** Local energy conversion happens in hub processes. They are governed by three constraints. *Hub supply* describes the relationship between the supply term  $\sigma_{ect}$  and the consumption of hub processes:

$$\forall e \in E, c \in C, t \in T: \quad \sum_{h|c \in C_h^{\text{in}}} \epsilon_{eht} - \sum_{h|c \in C_h^{\text{out}}} \epsilon_{eht} l_h^{\text{out}} \leq \sigma_{ect} \quad (3.107)$$

*Hub output by capacity* limits the activity of hub processes  $\epsilon_{eht}$  by their corresponding capacity variable:

$$\forall e \in E, h \in H, t \in T: \quad \epsilon_{eht} \leq \kappa_{eh} \quad (3.108)$$

*Hub capacity* finally limits the hub process capacity variable by the prescribed maximum capacity limit:

$$\forall e \in E, h \in H: \quad \kappa_{eh} \leq \bar{K}_h \quad (3.109)$$



**Vertex** The following constraint handles energy conservation in vertices, balancing process input/output, transport input/output and possibly a source term  $\rho_{vct}$  in case of source vertices. While the first constraint states the energy balance equation, the second constraint then limits this source term value by the limit of the vertex  $\bar{K}_{vc}$ . The energy balance is actually formulated as an inequality for performance reasons. The fact that  $\rho_{vct}$  causes non-zero costs in the cost function automatically forces equality for any optimal solution:

$$\forall v \in V, c \in C, t \in T: \quad \rho_{vct} \geq \sum_{a \in A_v^s} \pi_{act}^{\text{in}} - \sum_{a \in A_v^p} \pi_{act}^{\text{out}} \quad (3.110)$$

$$+ \sum_{p|c \in C_p^{\text{in}}} \epsilon_{vpct}^{\text{in}} - \sum_{p|c \in C_p^{\text{out}}} \epsilon_{vpct}^{\text{out}}$$

$$\forall v \in V, c \in C, t \in T: \quad \rho_{vct} \leq \bar{K}_{vc} \quad (3.111)$$

**Commodity** The single constraint here is specific to the environmental commodity subset. Equation *commodity maximum* limits the *net* generation of environmental commodities  $c \in C_e$  to the value set by parameter  $\bar{L}_c$ . The first line sums the net generation of a commodity in processes, the second line in hub processes. The time step weight  $w_t$  scales the mass/power flow values to an annual value:

$$\forall c \in C_e: \quad \sum_{t \in T} w_t \left( \sum_{v \in V} \left( \sum_{p|c \in C_p^{\text{out}}} \epsilon_{vpct}^{\text{out}} - \sum_{p|c \in C_p^{\text{in}}} \epsilon_{vpct}^{\text{in}} \right) \right. \\ \left. + \sum_{e \in E} \left( \sum_{h|c \in C_h^{\text{out}}} \epsilon_{eht} r_h^{\text{out}} - \sum_{h|c \in C_h^{\text{in}}} \epsilon_{eht} \right) \right) \leq \bar{L}_c \quad (3.112)$$

**Processes** Central energy conversion processes are defined through three time-dependent variables: their commodity input flow  $\epsilon_{vcpt}^{\text{in}}$ , the commodity output flow  $\epsilon_{vcpt}^{\text{out}}$  and, defined as the weighted sum of all input, the process throughput  $\tau_{vpt}$ . Equation *process throughput by capacity* limits the throughput to the process capacity:

$$\forall v \in V, p \in P, t \in T: \quad \tau_{vpt} \leq \kappa_{vp} \quad (3.113)$$

*Capacity borders* ensure that, if a process is built somewhere ( $\phi_{vp} = 1$ ), its capacity is within the range between  $\underline{K}_p$  and  $\bar{K}_p$ . Otherwise ( $\phi_{vp} = 0$ ) the process capacity must be zero.

$$\forall v \in V, p \in P: \quad \phi_{vp} \underline{K}_p \leq \kappa_{vp} \leq \phi_{vp} \bar{K}_p \quad (3.114)$$

*Input and output equations* relate the throughput to the input and output of commodities through the *input/output ratio* parameters:

$$\forall v \in V, p \in P, t \in T, c \in C_p^{\text{in}}: \quad \epsilon_{vcpt}^{\text{in}} = \tau_{vpt} r_{cp}^{\text{in}} \quad (3.115)$$

$$\forall v \in V, p \in P, t \in T, c \in C_p^{\text{out}}: \quad \epsilon_{vcpt}^{\text{out}} = \tau_{vpt} r_{cp}^{\text{out}} \quad (3.116)$$

Equations (3.97) to (3.116) together form the optimisation model **rivus**, whose optimal solution yields the minimum cost  $\zeta^*$  and the corresponding infrastructure network capacities and operation pattern.

### 3.5.4 Possible model extensions

**Basic principle** For increased simplicity, all parameters only have the minimum amount of domain sets. For example, all cost parameters only depend on the commodity  $c$  and process  $p$ , but not on location  $e$ . Any additional dependency, like a vertex-dependent peak demand scaling factor  $s_{vct}$  can be readily added, if only one can provide sensible data that exhibits significant differences.

**Process locations** Processes may be located in any vertex in the presented form. For the case study in this thesis, this complexity is not prohibitive and allows to discover “phase changes” from central to local heat generation more easily. For more large scale model applications, a constraining vertex subset of  $V_p$  per process (or one set for all processes) could be easily added, for which the capacity expansion variable  $\psi_{vp}$  may assume a non-zero value (or is defined at all). That way, the model should scale to higher number of vertices –  $|V| > 10^3$  – on desktop hardware.

**Redundancy** By including availability parameters for various model components like in model **dhmin**, outages of processes, sources or (critical) transmission edges could be modelled. This would allow for cross-sectoral infrastructure planning with proven redundancy against the specified outage types.

## Chapter 4

# Case studies

*The purpose of mathematical programming is insight, not numbers.*

---

Arthur M. Geoffrion (in 1976)

This chapter describes a series of case studies that were executed over the course of several years, using the models described in the previous chapter. Figure 4.1 shows their locations on a map within the outline of Germany. Each location and study highlights a different aspect of modelling energy systems, starting from a single-commodity (district heating) network and then moving towards multiple commodities. Due to the use of optimisation models, all studies have in common the (assumed) existence of a planning agent. A planning agent implies that all actions and decisions of the model are coordinated in a way to accomplish a globally optimal outcome, even if individual parties would suffer disadvantages compared to today's situation. How and whether the behaviour of such an entity can be realised, either through regulatory or deregulatory measures, is beyond the scope of this thesis. In contrast, these case studies serve the purpose of gauging whether *under idealised conditions*, such planning could uncover substantial benefits to justify more detailed (and complicated) investigations. This page lists a short summary of all case studies, followed by a detailed discussion of each case study in the following sections.

In Frankfurt, district heating network expansion planning was conducted using model **DHMIN**. In this study, the model was first applied on a large dataset. A key feature of this study was the measured gas consumption data. The consumption data allowed to calibrate the baseline estimation for the city's total heat demand. Based on this calibrated demand, several future demand scenarios were derived. Through collaboration with RWTH Aachen, network planning for all three sectors gas, electricity and district heating were performed in parallel. The study allowed the city utilities to independently verify the viability of their in-house network planning for the next decades.

In Munich, first the western region (Pasing, Westkreuz) was investigated with model **DHMIN**, as a reference plan had been published by SWM for that area. This study had the goal to test how close an optimisation model, only equipped with a roughly estimated heat demand distribution, could match the published reference in terms of network topology and coverage.

The next study then investigated the economic potential of using district *cooling* in the



Figure 4.1: Location of case studies in Germany

city centre of Munich using model **dhmin**. This model allowed to construct the cost-minimal network topology, while being fail-safe against the outage of any single cooling location, thus being (n-1)-redundant. Another remarkable property of this study is that by simply changing technical and cost parameters, a different energy carrier could be modelled.

This insight is then generalised by the development of model **rivus**, which is able to plan an arbitrary number of energy carrier networks simultaneously. The last presented case study in Haag in Oberbayern (Haag i. OB) finally shows the application of two models in combination to optimise both planning and operation of an energy system. Haag i. OB is the location for the project EEBatt's field test of Energy Neighbor, a 200 kWh battery storage container. To assess the viability of storage technologies like batteries in foreseeable future infrastructures, two models are used in combination. Model **rivus** is used similarly to the study in Frankfurt, but with much broader scenarios to create a portfolio of very different infrastructure types; different from the Frankfurt case study, all three sector networks are planned simultaneously. For each scenario, the utility of storage technologies can be assessed and compared. This is accomplished by coupling two models: first, **rivus** is used to *plan* the capacities for transmission and energy conversion processes in the study region with a spatial resolution. Then, model **urbs** uses these capacities as fixed parameters and *operates* optimally sized storage at a lower spatial resolution. The outcome is then summarised to gain insight on the different ways storage could play a role in a future energy system.

## 4.1 District heating network (Frankfurt)

From January 2012 to October 2013, Frankfurt based network company *Netzdienste Rhein-Main* (NRM) ordered the Institute of Power Systems and Power Economics (RWTH Aachen) and the Chair of Energy Economy and Application Technology (Technical University of Munich) to perform a target system study (German: *Zielnetzplanung*) for Frankfurt's gas, electricity and district heating networks. RWTH investigated gas and electricity, while TUM's task was concerned with district heating. As the three sectors (gas, heat, electricity) are becoming more and more connected, interactions among the sectors needed to be considered as well. Obvious examples are combined heat and power plants (central) and cogeneration units (distributed) that generate both heat and power, mainly from natural gas. The main results described here were also published by Schönsteiner et al. [107].

The district network planning was conducted using model **DHMIN**. In the case of electricity and gas network, every customer must be connected to both grids if requested. In the case of district heating networks, the supplier may decide whether to develop a given district or not. Therefore, the district heating network offers the highest flexibility (and uncertainty) in the future planning of the network structure. The key insight that allowed model **DHMIN** to be used on a full city was the roughly affine relationship between thermal pipe capacity (in kW) and the specific investment costs for its installation (in €/m). Before, each possible pipe diameter had been modelled with a separate discrete decision step, which multiplies not only problem size, but also cuts the solver from the possibility to exploit the well-known structure encoded in the energy conservation constraints to derive viable minimum pipe sizes. By introducing a continuous pipe capacity variable, the number of decision variables comes down to one per edge, greatly reducing problem size and computation time (cf. section 3.3).

### 4.1.1 Input data

In a first step, the official cadastral map of Frankfurt was used to map partly measured, partly estimated district heating, gas and electricity consumption data (referenced by address points) to physical buildings. This data assured that all planning decisions were derived from the best available knowledge about the demand side.

Secondly, the official street register was processed and converted to a street graph. This conversion was in part executed by using the street skeletonisation algorithm described in section 2.4.3. However, in the historic city centre, the automatic processing lead to an undesirable *honeycomb* effect. That means, that the street graph edges resembled rather a hexagonal grid than the actual street lines. The objecting region was then manually cleaned, so that the street graph closely followed the street pattern. This process lead to a final street graph with 9471 vertices and 13 805 edges.

As a third step, the building demands were aggregated to the nearest street graph edge to reduce the spatial resolution from over 100 000 buildings to the spatial resolution of the street graph. The resulting street graph had a complete length of 1591 km and a raw summed heat peak demand of over 4 GW. With an approximate heating period of about 1800 h per year this equals 7.2 TWh of annual heat demand. This amount and distribution form the main input to the following scenario development.

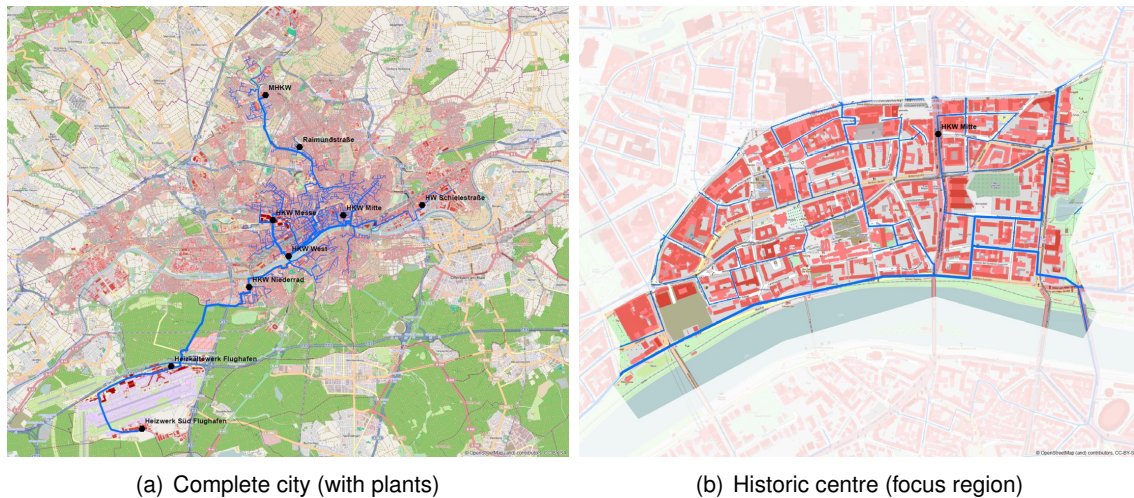


Figure 4.2: Target district heating network in Frankfurt for scenario CON in the year 2025

#### 4.1.2 Scenario development

For the two years 2025 and 2050, three different demand scenarios each were investigated. The first scenario was based on a business-as-usual extrapolation of today's (partly measured) demand distribution, called *conventional* or CON for short. In it, foreseen growth trends for the building stock were extrapolated linearly. The two other scenarios, called ECO+ and ECO++, made more and more progressive assumptions on energy related actions of the general public: these include the adoption of distributed heat generation from small cogeneration plants or domestic units, use of electric heat pumps, increased thermal insulation of buildings, and finally an uptake of electric vehicles.

Each of these demand reductions was quantified, in part based on existing studies, in part by conducting own analyses. This work was conducted by colleagues Karl Schönsteiner and Markus Wagner at the Institute for Energy Economy and Application Technology, and thus is not described here further. The following discussion instead focusses on district heating and especially aspects of model development.

#### 4.1.3 Results

Figure 4.2 shows resulting optimal target networks for the scenario CON, together with the location of existing and planned district heating plants for the target year 2025. In total, 9 heating plants are located within the study region. Three plants are located near the city centre, two plants at the airport. The remaining four plants are spread over the other districts suitable for district heating. The location and size of the major pipelines resembles today's structure. All heat generation plants are used to their full capacity, i.e. the heat generation capacity is the limiting factor for the heating network size, not the lack of development area. The plant locations are connected via a backbone network with higher capacity than necessary for a simple supply of a plant's nearest demands. Instead, this backbone has two benefits: first, it serves as a capacity reserve, allowing other plants further away to satisfy off-peak load.

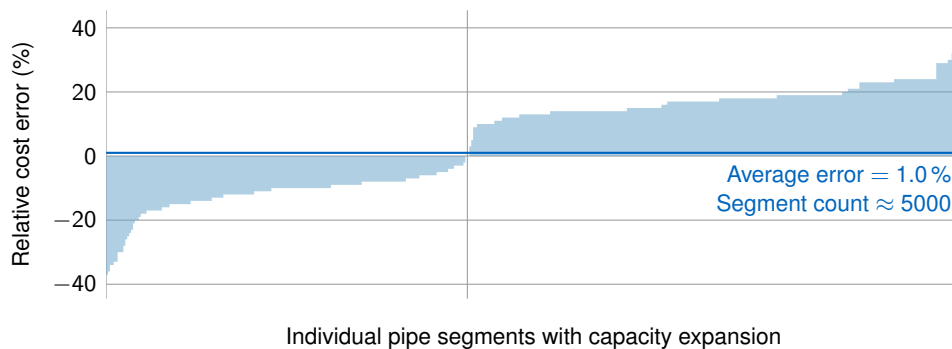


Figure 4.3: Relative cost error for individual pipe segments, compared to closest discrete available pipe diameter. Plot shows individual error for each pipe segment, sorted ascending by error value. While individual segments have considerable error ( $\pm 10\%$  or more), the overall error cancels out almost perfectly (1%).

Secondly, it allows for using more efficient plants in base load operation, while less efficient plants can be used only in times of high load.

The sensitivity of the target network against the foreseen changes is small. In all investigated scenarios, the same major network structure remained cost-optimal. Only the extent of the minor distribution lines changed slightly. For lower demand densities in scenarios ECO+ and ECO++ in the year 2025, a slightly larger area around the city centre has to be developed in order to fully utilise the heat generation plants. The only exception to this observation is scenario ECO++ in 2050. There, the projected heat demand density is reduced so far that not enough viable development area is left to use up all producible heat.

This case study also gave the opportunity to quantify the error that was introduced by linearising the actually discrete cost function. Figure 4.3 shows the result of an error analysis performed on the level of individual pipe segments that were to be built due to one of the optimisation results.

The main insights gained from this project are two-fold: first, the linear investment cost function of model **DHMIN** is precise enough to mimic the planning behaviour of human expert planners, when equipped with the same demand data as a basis. The second insight is that lack of sector coupling limited the conclusions that could be drawn from the individual models. For example, the district heating planning was done purely from a cost-revenue perspective for that single energy carrier, with pre-defined locations of heat generators. While this hypothesis (fixed heat generation locations) is quite valid for existing heating networks or in short- to mid-term projections, a long term planning should be able to determine favourable size and locations of energy conversion processes. However, this requires that all relevant energy carriers (electricity, heat, natural gas) are included in a single model.

This desire to model all energy carriers with a single model in high spatial resolution was the motivation for model **rivus**, whose application is shown later in case study Haag in section 4.4.



## 4.2 District heating network (Munich west)

While the previous case study showed the suitability of the model for large input data sets, it could not be used to strictly validate the planning decisions made by the optimisation model due to lack of an independent reference plan. This case study, originally published as a paper in 2014 [24], has the benefit of such an independent reference. Additionally, this study only relies on publicly available information for preparing its input data.

Munich has a huge district heating network with a total length of about 800 km. The network delivers about 4 TWh of thermal energy per year according to the city's municipal utility company Stadtwerke München (SWM) [113]. Due to a large area with high building density, development of district heating is expected to remain profitable for decades to come. Therefore, expansion projects for the district heating network in the outer districts have been and are being planned and realised.

SWM had published an extension plan for their district heating network in the west of Munich [112]. Figure 4.4 shows the location of the extension region within Munich, highlighted in red. Unfortunately, the original plan is no longer available online. However, figure 4.5(a) shows a reproduction of the original extension plan. The three marked *source vertices* are (A) the connection to the remaining district heating network, (B) the CHP plant of public baths and (C) an unlabelled heat source. That last heat source is used for scenario WEST. With this reference in place, the question to be answered in this case study is: can a simple optimisation model like **DHMIN** reproduce the same or at least similar decisions as the published plan, when equipped with an only crudely estimated heat demand?

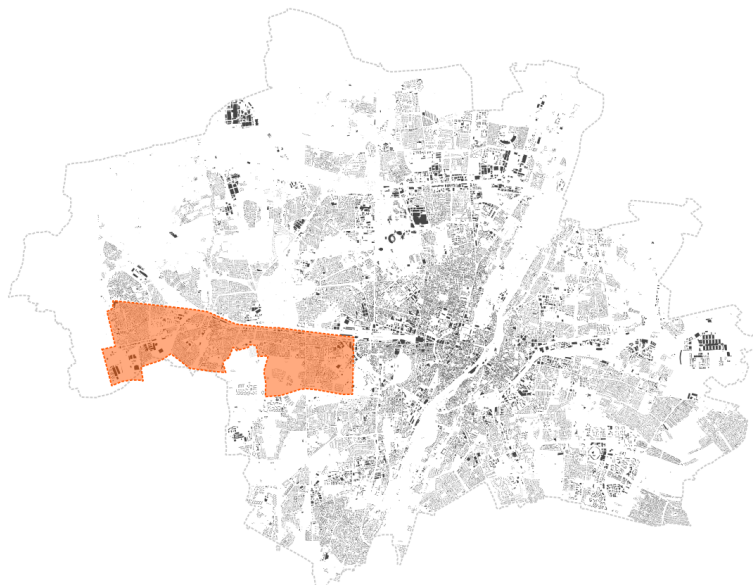


Figure 4.4: Area of investigation for case study Munich west within Munich. Data: OSM [91]



### 4.2.1 Data preparation

#### Heat demand

The main determinant for the optimisation model is the density of heat demand. This section describes the process for estimating this density from publically available data only. The spatial basis is a database export from the OSM project. Within the study area, it contains a total of 9662 building outlines. Building use is assigned from larger landuse polygons, if no per-building classification is already present. The resulting building typology (with over 20 types) is then reduced to six use types (residential, commercial, retail, industrial, school, hospital) with significant total building area. Figure 4.5(b) shows the distribution of buildings and their type.

While the building outline is a rough approximation for the gross floor area, the height of a building is obviously needed as well to estimate the total area to be heated. These buildings heights were visually assessed in manually drawn areas, shown in figure 4.5(c). During this step, obviously wrong landuse information of some buildings were corrected. Table 4.1 shows the resulting counts, cumulated area and area-specific heat demand per building type. The heat demands were derived from average demand statistics for private households [28] and the commercial sector [60]. Given the high uncertainty in estimating the building areas, these values are sufficiently accurate.

The building heights are not yet mapped with sufficient coverage in OSM. Therefore, the number of floors per building  $n_b$  was visually assessed from aerial images that can be accessed on Google Maps [37]. This attribute was not assigned individually to each building, but by identifying regions of approximately homogeneous building types. These regions were then traced manually and assigned a representative building height. The resulting division is shown in figure 4.5(c).

If the total heated area  $A_b$  of a building would now be calculated by multiplying the area of the building outline  $A_b^{\text{poly}}$  with the number of floors, the resulting area would in general overestimate the real area because it includes exterior walls, interior walls, and unheated rooms. Therefore, a correction factor  $z_b$  is introduced to correct for that bias. Its definition is drawn from an article about remote sensing for estimating heat demand [32]. Its value depends on the number of floors of that building:

$$z_b = \begin{cases} 0.75 & n_b \in \{1, 2, 3\} \\ 0.80 & n_b \in \{4, 5, 6\} \\ 0.85 & n_b \in \{7, 8, 9, 10\} \end{cases} \quad (4.1)$$

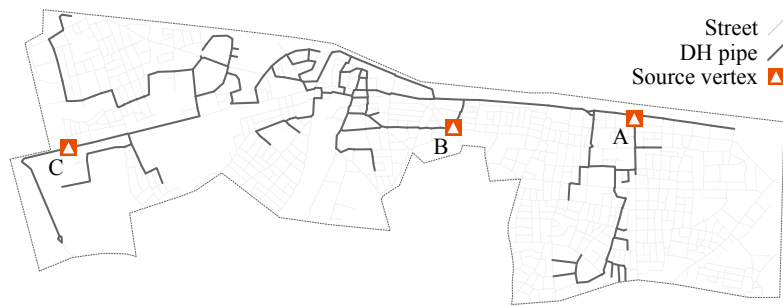
Thus, the total heated area  $A_b$  for a building  $b$  is given by the expression

$$A_b = A_b^{\text{poly}} n_b z_b. \quad (4.2)$$

The resulting heat demand density distribution is shown in figure 4.5(d).

#### Street network

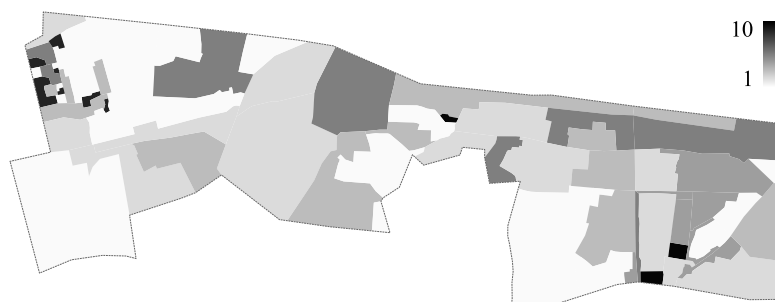
The street network is the geographic basis for the planning decisions. The underlying graph was also extracted from the OSM database export. The raw way segments were first filtered



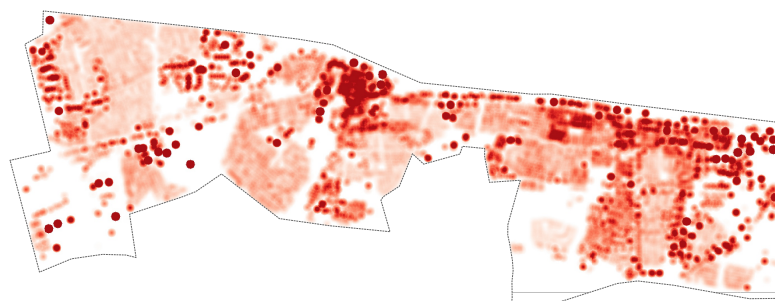
(a) Reference plan. Data: SWM [112]



(b) Building outlines with usage type. Data: OSM [91]



(c) No. of building levels. Data: Google Maps [37]



(d) Heat demand density, estimated

Figure 4.5: Input data for case study in Munich West

Table 4.1: Building Types and Heat Demand Parameters

Building type	Count	Area 10 <sup>3</sup> m <sup>2</sup>	$h_k$ W/m <sup>2</sup>	$H_k$ kWh/(m <sup>2</sup> a)
Residential	9263	6502	80	140
Commercial	211	661	80	100
Retail	59	308	100	100
Industrial	79	370	80	80
School	17	70	80	100
Hospital	8	58	100	200

by the value of the attribute `highway` and restricted to the values `motorway`, `trunk`, `primary`, `secondary`, `tertiary`, `residential`, or one of the corresponding types with the suffix `_link`.

The filtered street segments were then processed by the skeletonisation algorithm described in section 2.4.3. In this study, the buffer distance was set to 20 m, and the simplification tolerance to 15 m. The resulting graph consists of 1112 vertices and 1630·2 edges.<sup>1</sup>

Finally, the estimated per-building heat demand is aggregated to the skeletonised street graph by nearest distance. The set  $N_{ij}$  refers to the buildings  $b$  which are closest to that edge:

$$N_{ij} = \{b \in B \mid b \text{ is closest to } e_{ij}\}$$

The per-building areas  $A_b$  are then weighted with the corresponding specific heat demand parameters from table 4.1 for peak  $h_{k_b}$  and for annual  $H_{k_b}$  demand are aggregated to per-edge demand values.  $k_b$  here refers to the building type  $k$  of a certain building  $b$ .

$$d_{ij} = \sum_{b \in N_{ij}} A_b h_{k_b} \quad (4.3)$$

$$D_{ij} = \sum_{b \in N_{ij}} A_b H_{k_b} \quad (4.4)$$

The total annual heat demand in the case study area amounts to 1129 GWh of thermal energy. It is distributed over the street graph with a total length of 183 km. As roughly indicated in figure 4.5(d), this distribution is not uniform, but decreasing exponentially. Over 21 km of street segments have no associated heat demand.

## 4.2.2 Scenarios

Table 4.2 sets the technical and economic parameters used for both of the two investigated scenarios BASE and WEST. Investment cost parameters were estimated from experience: fixed investment costs of 1000 €/m and variable investment costs of 0.05 €/(m kW) are estimated to be rather high, in order to err on the side of a more conservative planning result. The annuity factor  $u$  is derived from an interest rate of 6% and a depreciation duration of 40 years, a typical conservative estimate for the commercial lifetime of DH pipes. Heat generation costs of 3.5 ct/kWh and a revenue of 7.5 ct/kWh are representative price assumptions.

<sup>1</sup>The factor two is caused by using *directed* edges in model **DHMIN**.

Table 4.2: Parameter values for the case study in Munich west

Name	Unit	Value
$c_{fix}$	€/m	1000
$c_{var}$	€/(m kW)	0.05
$c_{om}$	€/(m a)	10
$c_{heat}$	€/kWh	0.035
$c_{rev}$	€/kWh	0.075
$u$	1/a	0.067
$w_{fix}$	W/m	20
$w_{var}$	W/(kW m)	0.0002
$T_{filh}$	h/a	1800
$b$	1	0.7
$q$	1	0.7

In scenario *BASE*, only source vertices A and B from figure 4.5(a) are available. Vertex A represents the connection to the city centre and has a maximum capacity of 600 MW, vertex B is a public bath's heating plant with a capacity of 9 MW and vertex C has a capacity of 0 MW

In scenario *WEST*, the third source vertex C is changed to 250 MW to demonstrate the effect of a large local heat plant in the West of Munich.

### 4.2.3 Results

Table 4.3 summarises the main results for both investigated scenario *BASE*. 601.2 GWh of heat are supplied, which equals about 76 % of the connectible heat demand. In scenario *WEST*, this value increases to 661 GWh or to about 84 % of the connectible heat demand.

The street graph has a total length of 183 km. In scenario *BASE*, the total network size is 85 km or 46 % of the total length. However, it must be remembered that over 12 % of the total street graph length have no associated heat demand. In scenario *WEST*, the total network size increases to 95 km or 52 % of the street graph length. Not surprisingly, this expansion of the network size concentrates in the western part of the the study area, since this is where the new source vertex is located.

Figure 4.6 shows the network structure for both scenarios graphically. In scenario *BASE*, a tree network with root in source vertex A connects to the demand clusters in the central north of the study area, before branching out to connect the high-rise buildings in the north-west (cf. figure 4.5(c)).

The thermal network losses increase from 7 % to over 10 %. This increase is caused by the higher share of pipes with small diameters (capacities) in scenario *WEST*; with the chosen values for the loss parameters  $w_{fix}$  and  $w_{var}$ , the shorter average transport distance is not able to compensate the – on average – smaller distance from heat source to demand.

A visual comparison of the cost-optimal pipe network in scenario base shown in figure 4.6(a) to the published reference planning in figure 4.5(a) shows: the major topology of the optimised network, i.e. the locations pipes with capacities of 10 MW and higher, resembles the reference plan very well, with the exception of Neuaubing in the south-west of the study area. However, this difference can be lead back to the lack of coverage in the building dataset and the difficulty

Table 4.3: Summarised optimisation results

Name	Unit	Scenario	
		BASE	WEST
Total pipe length	km	85.2	95.4
Relative thermal loss	%	7.0	10.1
Supplied heat	GWh	601.2	660.8
Annual invest cost	M€	8.0	8.3
Operation & maint.	M€	1.8	1.9
Heat generation cost	M€	22.5	25.5
Heat revenue	M€	-45.1	-49.6
<b>Cost balance (z)</b>	<b>M€</b>	<b>-12.7</b>	<b>-13.9</b>

of correctly estimating heat demand for the commercial and industrial sites in that area. Areas with good coverage in the building dataset match quite well.

To investigate the impact of the connection quota parameter  $q$  on the BASE scenario results, a sensitivity analysis was conducted. For ten values of  $q \in \{0.1, 0.2 \dots 1.0\}$  the network optimisation was performed and the total supplied heat tabulated against this parameter. The resulting graph is shown in figure 4.7. It can be seen that the relationship is nearly linear, with next to no economic district heating for a connection quota of  $q = 0.1$ . The diagonal line shows the connectible heat demand for each connection quota value, the horizontal line the total heat demand.

#### 4.2.4 Discussion

In the previous case study in Frankfurt, model **DHMIN** was provided with (confidential, but) accurate data on the heat demand situation in a city. In this case study, the heat demand was crudely estimated solely on public data. Nevertheless, like before, the heating network returned by the solver closely matched a reference. In other words, a simple MILP model can be used to generate a first approximation for a given planning task with surprising accuracy. Despite the underlying optimisation problem being NP-hard, state-of-the-art solvers handle the sparse problem instance quite well and generate reasonable layouts within minutes even for thousands of discrete decision variables. It can be concluded that model **DHMIN** is useful to identify (or rule out) regions for possible network expansion plans.

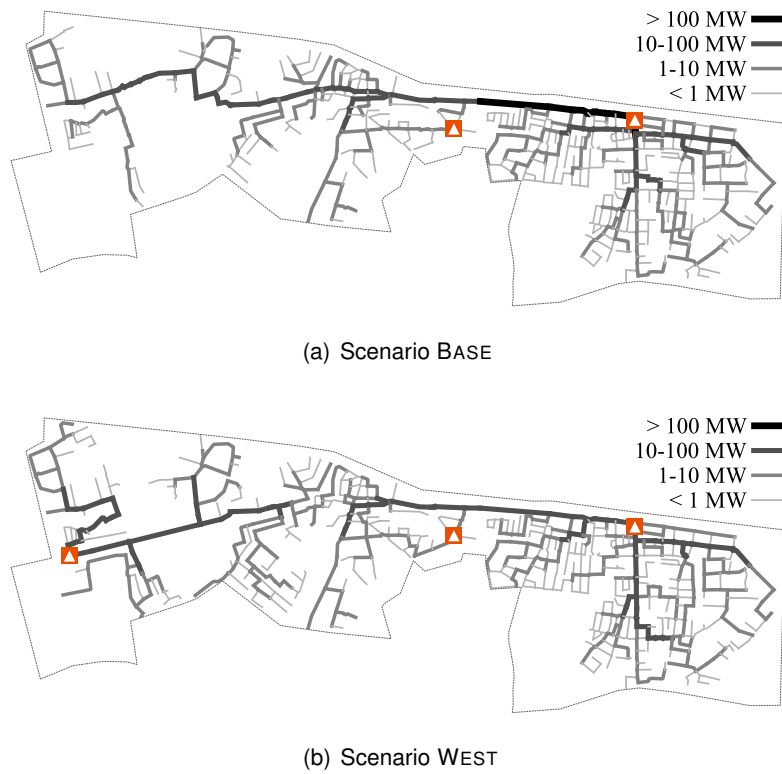


Figure 4.6: Optimal district heating network topologies  $\bar{\pi}_{ij}$  in Munich west

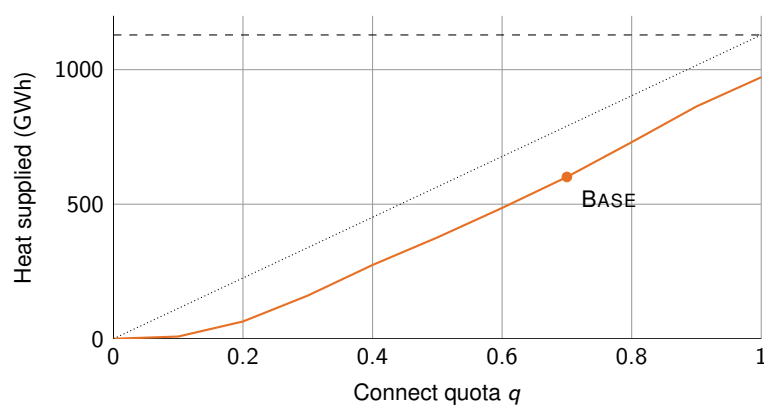


Figure 4.7: Total supplied heat (—) for different connect quotas. The diagonal line (⋯) shows the connectible portion of the total heat demand (---).

### 4.3 District cooling network (Munich centre)

This case study has been conducted in collaboration with Patrick Krystallas from SWM, Munich's municipal utility and published as an article in *EuroHeat & Power* [63]. Here, the additional features of model **dhmin** (compared to **DHMIN**) allow to investigate redundancy requirements for the availability of cooling stations. Economic analysis is also improved, as the trade off between peak load satisfaction and annual demand revenue is more precisely represented. The redundancy requirement allows for planning a district cooling network that can satisfy almost all cooling demands even when any single cooling station is non-operational at a given instant.

It must be pointed out that a model developed for district *heating* is now used for planning a district *cooling* network, just by adjusting its parametrisation for costs, revenues and losses.

As this study was conducted with partially sensitive data, all data presented in this chapter is modified. This measure is to protect sensitive information, while keeping the central conclusions intact.

#### 4.3.1 Input data

The point of this study is contrary from the previous one on district heating, not only in temperature. Instead of publicly accessible building data, planning assumptions were used as input to the model. The underlying spatial representation was digitised from a hand-drawing of coarse locations of large cooling demands. The supply side is represented by a list of existing (Stachus) and planned (e.g. Isar) cooling plants. Internal production costs for cold water, together with expected retail prices for cooling energy were used as cost parameters.

The study area is located between the following points of interest (POIs): Central station in the west, University (LMU) in the north, river Isar in the east, and Sendlinger Tor in the south. The total assumed cooling demand is estimated about 100 MW and the maximum capacity of all cooling stations around 125 MW.

The locations of cooling plants (blue and green points) and cooling loads (red lines) are displayed in figure 4.8. There are seven potential cooling stations considered in total. The loads are peak cooling demands of potential customers (measured in MW), aggregated to low spatial resolution. These consumers are mainly commercial entities, as space cooling is not yet common in the residential sector. These peak loads are expected to occur only during the hottest days of the year, analogous to the peak electricity or peak heating demand during winter. As with all central infrastructure, the network capacities must be planned in order to satisfy peak load. Note that *none* of the cooling loads must be satisfied. Instead, the optimisation identifies profitable consumers for the given cost and revenue assumptions.

The white and red lines form the arcs  $A$  of the graph of possible pipe routing options, while all intersections and cooling stations form the vertex set  $V$ . Lengths are calculated as direct point to point distances. This level of detail is sufficient, as the main source of insecurity lies in the parametrisation of the pipe investment costs and the generation costs for cooling energy. These are calculated by the sum of running OPEX and depreciated CAPEX using the annuity factor from equation (2.34).



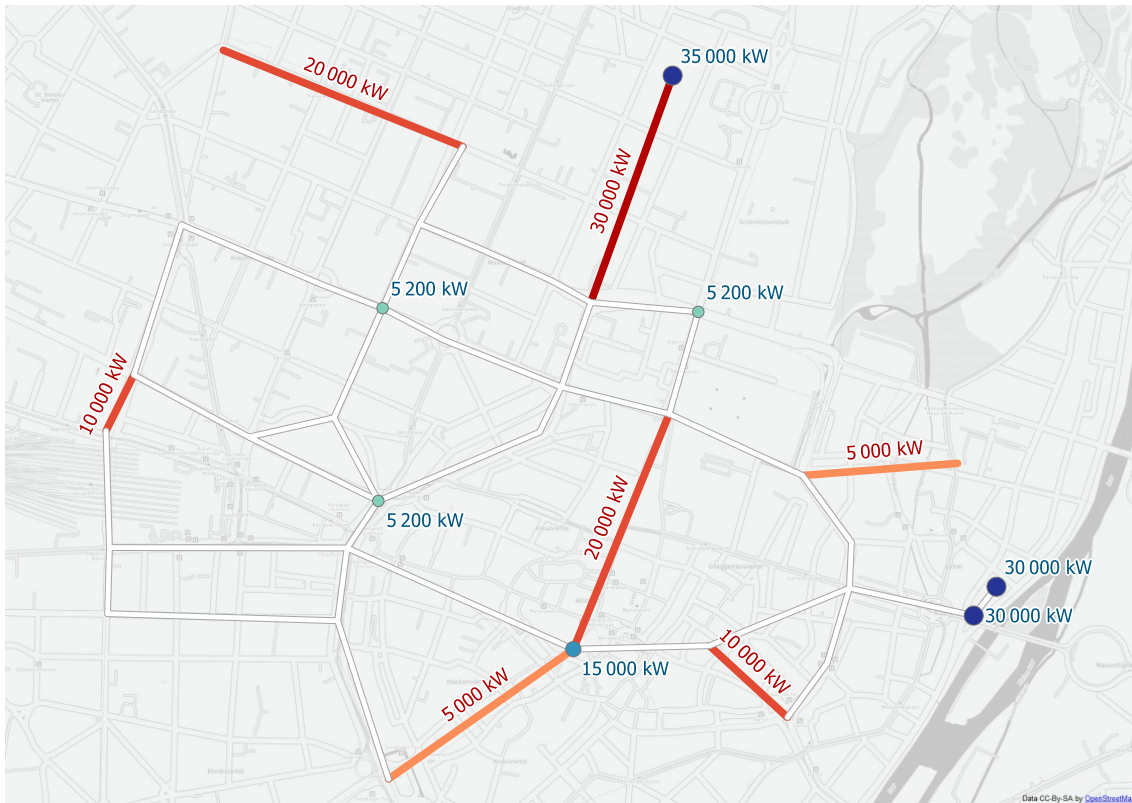


Figure 4.8: Locations of possible cooling stations (●) and cooling demands (—) in case study area (—, labels in kW) Munich centre. Background map: OSM [92]

Concerning the costs of cooling energy: three of the seven considered cooling stations have watercourse access: the existing plant at Stachus with a capacity of 5 MW and a hypothetical cooling plant near river Isar with 60 MW of maximum cooling power (modelled as two sources with half the capacity each) have lower production costs. The running watercourse can either be used for direct cooling, especially during winter, or as a heat sink for more efficient re-cooling. The other four cooling stations are assumed to use air for cooling the working fluid of the cooling stations. The sites were pre-selected considering technical (space) and practical (noise) constraints.

Operational and investment costs are chosen according to experience with the existing cooling network around the existing cooling station at Stachus. Due to the central location, fixed investment costs for pipe networks are unsurprisingly expected to be much higher than outside the city centre (as assumed in the previous case study in Munich west).

Technical parameters are assumed as follows: the connect quota  $q$  is set equal to 1, or 100%. That means, any single demand location must either be satisfied completely, or not at all. Concerning the concurrence effect parameter  $b$ , it is set to a value of 0.9, higher than typical in district heating. This is due to the bigger size of a single customer and the higher temporal correlation of peak cooling demands. The maximum possible pipe capacity  $C_{ij}^{\max}$  is set high enough to not interfere with the planning decision. No existing pipe network is



Table 4.4: Time steps in case study Munich centre

Name	Scaling $s_t$	Length $w_t$ (h)	Comment
<i>Peak</i>	1.0	1	Peak demand, full availability
<i>High</i>	0.8	1000	High demand, full availability
<i>Half</i>	0.5	1200	Half of peak demand, full availability
<i>Outage</i> ( $\times 7$ )	1.0	1	Peak demand, $a_{it} = 0$ for one source

Table 4.5: Summary of optimisation results in case study Munich centre

Indicator	Value
Total pipe length	12.5 km
Supplied cooling peak load	100 MW
Supplied cooling energy	140 GWh
Relative thermal network losses	<1 %

assumed, i.e.  $g_{ij} = 0$  for all arcs.

Time steps are summarised in table 4.4. Time step *peak* has a full scaling factor  $s_t = 1$  and thus ensures that all considered loads can be satisfied simultaneously. Its low duration  $w_t = 1$  ensures that this moment does not contribute to the running costs. This is the task of the two time steps *high* and *half*. They have lower scaling factors of 0.8 and 0.5, but durations of over 1000 h each. Together, they sum to a cooling full-load duration  $\sum_{t \in T} s_t w_t$  of approximately 1400 h. The scaling factor and weight of the partial load time steps has a purpose: in each time step, a different configuration of operational sources can be chosen. The last time step *outage* actually represents seven more time steps. For each of them, one of the seven source's availability parameter  $a_{it}$  is set to zero, so that the other sources must be able to satisfy all connected demands. These time steps are what forces the model to create a network structure that has a  $(n - 1)$ -redundancy with respect to the cooling stations. This requires that there are enough cooling station capacities available; otherwise, the model would become infeasible.

### 4.3.2 Results

Table 4.5 summarises key indicators of the optimisation result. With the present cost assumptions, it is profitable to satisfy all of the given cooling loads of 100 MW or 140 GWh. The total pipe length of 12.5 km does only include the distribution network, not connections to end consumers that would have to be planned individually.

Figure 4.9 shows the optimal locations of district heating pipes and their capacity. Again, all demand locations are connected to a tree-like distribution network. The pipe capacities are sized sufficiently large so that all cooling loads can be met, even in case a single cooling station is not available. The main backbone of the network are the two cooling stations in the east of the study area. Their operation costs are slightly cheaper due to the availability of a watercourse. The most northern cooling station with a capacity of 35 MW is connected with a pipe capacity of 30.5 MW. It is only used as a fallback option if one of the two eastern cooling stations has an outage. This result shows that the capacity-independent investment

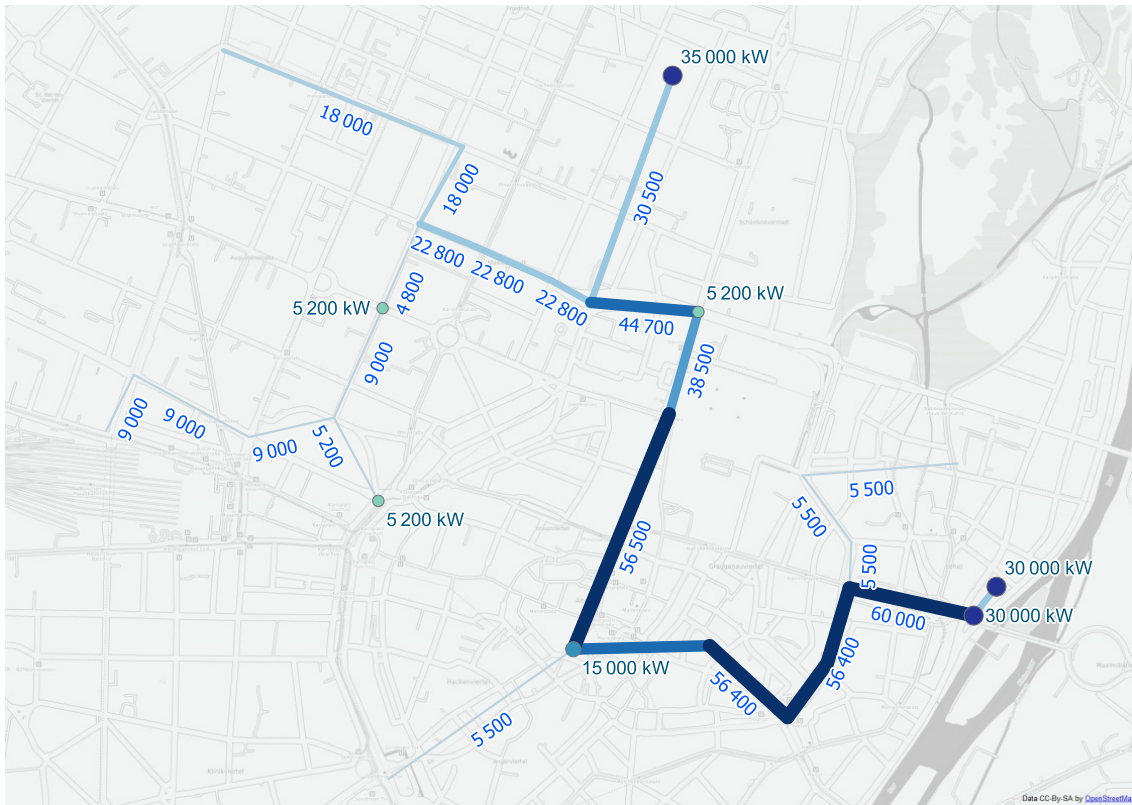


Figure 4.9: Optimal district cooling pipe capacities (—, labels in kW) in Munich centre: Cheapest cooling source at Isar (east) forms backbone of minimum spanning tree to all loads. Background map: OSM [92]

costs for pipe construction are the dominant factor when compared to the capacity-dependent cost component. The network structure also indicates that thermal losses are not much of a problem at the distances faced in this case study. This fact will become even more obvious in the next paragraphs, which compare the thermal power flow through this pipe network at two different instances.

Figures 4.10 and 4.11 show the actual power flow in two of the ten modelled time steps. The first image shows the operation of cooling stations when all plants are available: the main cooling power (60 MW of 90.1 MW) is provided by the eastern Isar stations. In case the reader is wondering why a total cooling supply of just over 90 MW is sufficient to satisfy 100 MW of cooling peak load: the concurrency effect (parameter  $b$  in model **dhmin**) is assumed to reduce the peak load “seen” by the network by 10% to just 90 MW.

In figure 4.10, the northern cooling station is not used at all due to its higher costs for re-cooling, even though thermal losses would be lower due to the proximity of the adjacent load of 30 MW. Only when one of the two Isar cooling stations is not available, as shown in figure 4.11, the northern cooling station is activated.

It goes without saying that this case study cannot replace a hydraulic simulation or a detailed operational planning of cooling station equipment. However, this model is suitable for identifying (or, in this case, confirming) a cost-optimal network structure with respect to

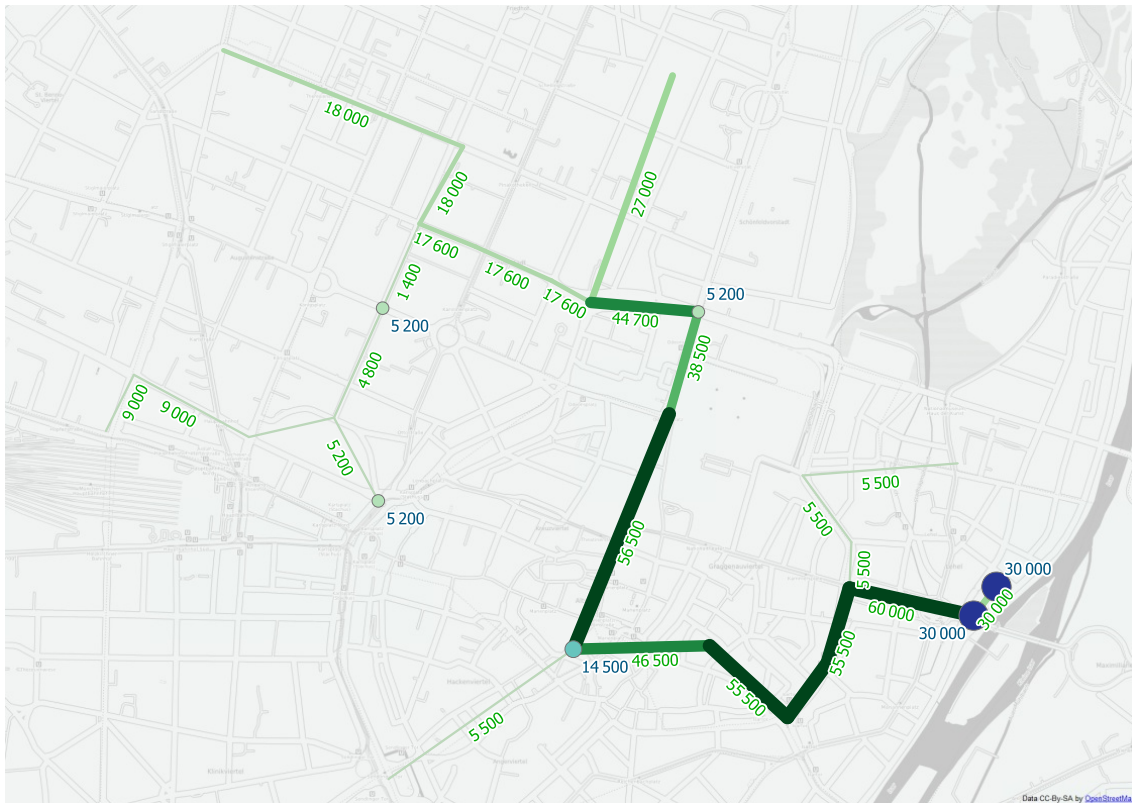




Figure 4.10: Optimal district cooling power flow in time step *All sources working* (   , labels in kW): cheapest sources at Isar (east) and Stachus (west) fully used. Background map: OSM [92]

revenue, while satisfying both basic technical and advanced reliability requirements.

### 4.3.3 Discussion

All in all, this case study shows two main results: first, development of district cooling in Munich's centre is profitable, especially in regions with high cooling demands from commercial buildings. Secondly, the model shows that a combination of different re-cooling options (both water and conventional air cooling) is needed in order to have both cost-efficient and reliable cooling supply throughout the supply area.

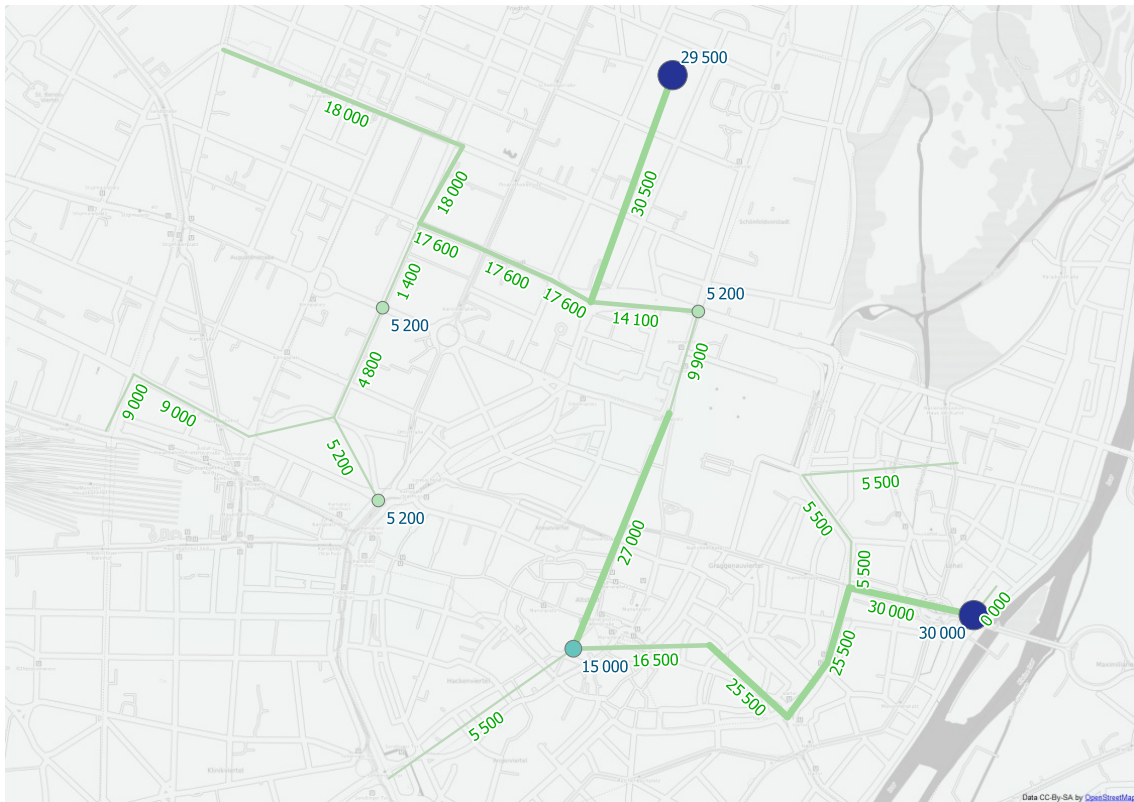



Figure 4.11: Optimal district cooling power flow in time step *Outage of an Isar source* ( , labels in kW): fallback capacity at university (north) used. Background map: OSM [92]

## 4.4 Multi-commodity network (Haag)

Haag in Oberbayern (Haag i. OB) is a market town in Upper Bavaria about 50 km east of Munich and has about 6000 inhabitants. It was chosen for this case study because of EEBatt [116], an interdisciplinary research project at TUM concerned with energy storage. The field test *Energy Neighbor* of its prototype stationary battery storage system is being conducted in Moosham, a small village located north-west of Haag. This case study is to investigate how a widespread use of such batteries could influence the planning and operation in possible future energy systems. For this type of investigation, the idealisations inherent to optimisation models are beneficial, as they allow to establish an outer bound of feasibility for the combination of the investigated technologies.

This case study combines the two models **rivus** and **urbs** on the same study area. Figure 4.12 shows how both models are coupled. First, model **rivus** is used to determine different infrastructure topologies for a variety of scenarios. The optimisation result yields capacities for energy conversion processes and transmission capacities. Then, **urbs** uses these capacities as parameters in aggregated form. Using hourly load data, and time series for intermittent renewable energy sources, it is then investigated under which economic and technical conditions *energy storage* changes the use of the given capacities. In scenarios in which storage use changes the use of transmission capacities significantly, this information is then propagated back as a change in *peak load* that must be satisfied by the distribution network. Finally, **rivus** is used with that *reduced* peak load to quantify the possible cost saving and compare it to the additional investment on the side of storage technologies. Depending on the cost assumptions, one can either estimate lower or upper bounds on when storage would be beneficial.

This case study puts emphasis on the principle of coupling two optimisation models together and (solving) the complications that arise from such an endeavour. The parameter values used in these scenarios however are mostly only indicative values, so any results should not be interpreted as if they would allow factual conclusions about desirable design choices for the actual energy system of Haag. To allow such recommendations, much more work has to be put into properly calibrating all parameter values and technology choices.

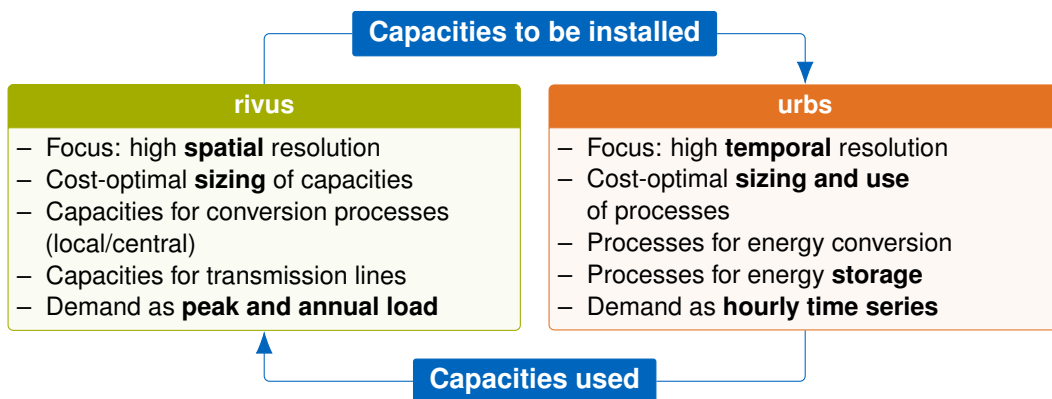


Figure 4.12: Model coupling of **rivus** (network sizing) and **urbs** (storage scheduling)



The use of two models with combined high temporal and high spatial resolution implies that also input data – mainly electricity and heat demand – must be prepared in both high temporal and high spatial resolution.

Using buildings as the finest spatial granularity easily exceeds the resolution of **rivus** in the spatial dimension, as it is designed to operate on street segments, i.e. the edges  $e \in E$  of the street graph. Therefore, using statistical estimates of the per-building consumption are considered sufficiently accurate to synthesise electricity and heat demand at street resolution.

To generate demand time series for using in **urbs**, building load curves by usage types are required with a resolution of about one hour. The best way to synthesise these load curves would have been to implement a simple thermal building model and running it with local climate data. Instead, combined load curves for buildings were required, as isolated heating or electricity load curves are not sufficient to capture the temporal correlation between both demands. Therefore, certain lengths had to be gone in order to find a repository of suitable load curves and identify a set of time series that could be representative for Haag i. OB. For the lower spatial resolution, the study area has been manually divided into 12 clusters, whose definition is given later in section 4.4.3. First, generating the spatial demand pattern for Haag i. OB is discussed.

#### 4.4.1 Data preparation for rivus

Like in previous case studies, the OpenStreetMap database [91] serves as the foundation for estimating the spatial energy demand distribution in the study area. Figure 4.13 shows the location and sizes of all mapped buildings in Haag i. OB as of January 5th, 2015. The outward villages Diezmanning and Moosham are shown in the map inserts. The total number of mapped buildings in the study area amounts to 1738. The two diamonds show the (assumed) locations of sources for electricity and natural gas. For electricity (yellow), the nearest point to the next mapped substation is marked, while for natural gas (brown) the nearest mapped pipeline determines the source position.

##### Street graph

Like in the case study 4.2, the original street network as shown in figure 4.13 is first reduced in detail to an abstract street graph, using the skeletonisation procedure described in 2.4.3. The parameters used here were determined empirically and are 30 m for the buffer distance  $l_{\text{buf}}$ , 15 m for the dissolve length  $l_{\text{dis}}$  and 30 m for the simplification threshold  $l_{\text{sim}}$ .<sup>2</sup> The resulting street graph can be seen later in part in figures 4.15, 4.17, and completely in 4.18. It contains 153 vertices and 203 edges with a total lengths of approximately 36.2 km. With this graph in place, the next step is to estimate the spatial energy demand distribution.

##### Demand density

The peak power demand  $d_{ec}$  per commodity  $c$  and per edge  $e$  is determined as follows. The most important determinant of the peak demand is certainly the gross floor area of adjacent buildings. For a rough overview, an aggregate value of building areas per *cluster* is given in

<sup>2</sup>The full procedure is documented in file `data/haag15/streets_to_edge.py` in repository **rivus**.

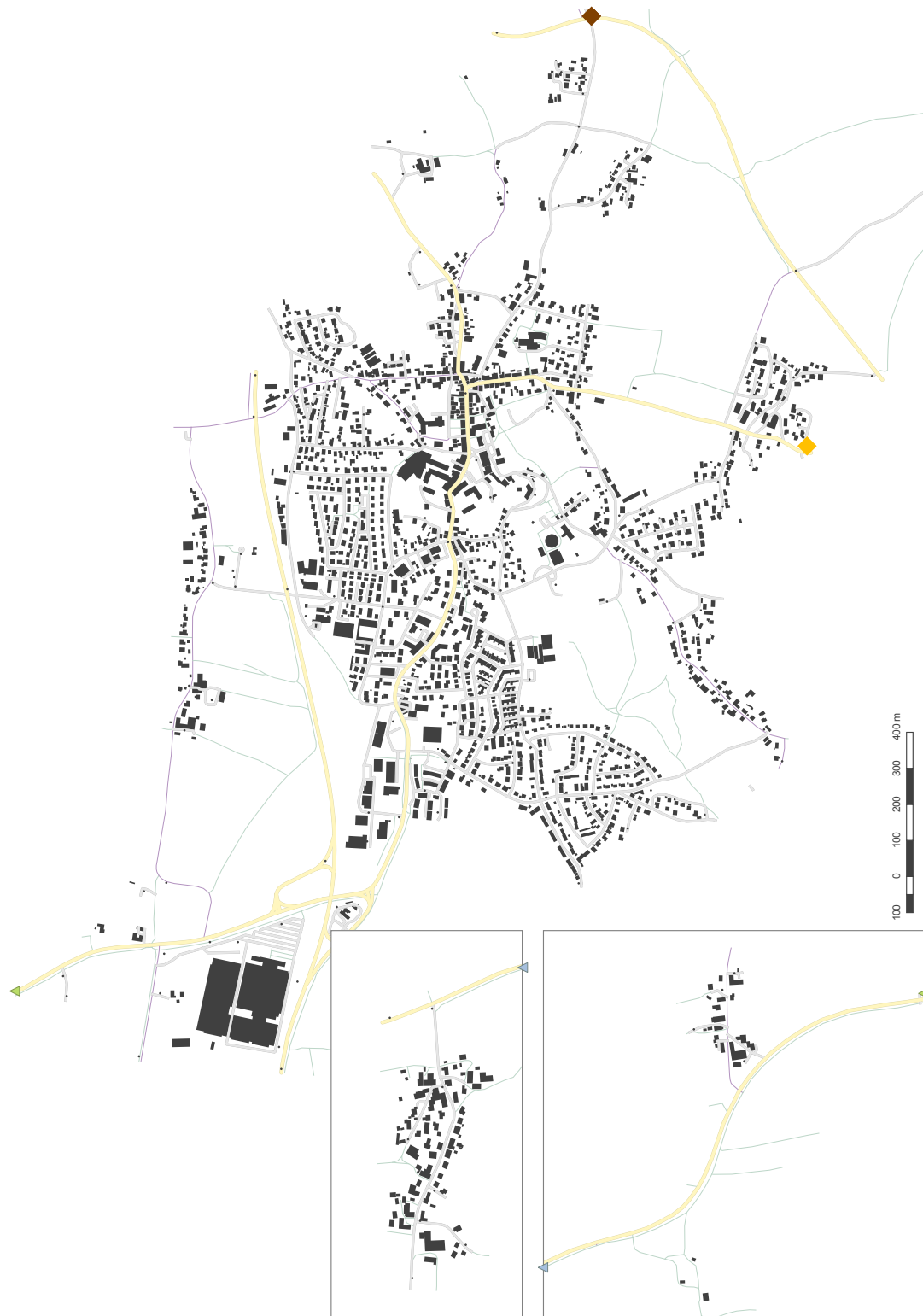


Figure 4.13: Map of mapped buildings in Haag i. OB. The two map inserts are located in the north west of the main part and connected by the green/blue triangle symbols.

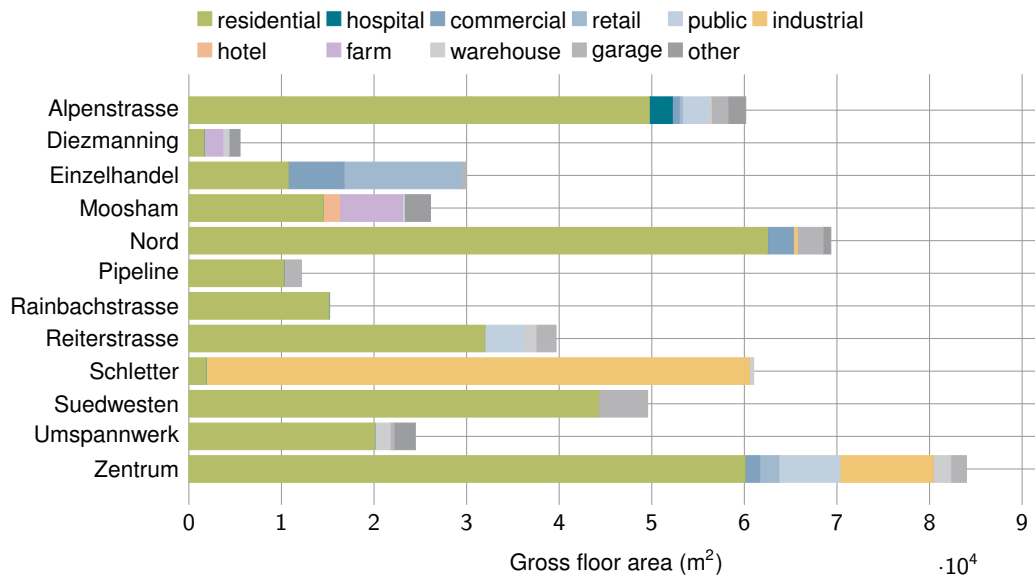


Figure 4.14: Aggregated building areas per cluster and OSM building type in Haag i. OB

figure 4.14. The clusters are manually defined districts whose definition is shown in figure 4.18 and will be used later as the spatial resolution in model **urbs**. As can be seen from the building area statistics, residential and industrial areas clearly dominate in terms of quantity. Each individual building's gross floor area  $A_b$  is multiplied by a specific heat demand  $h_{kC}$ , depending on the building type and the demand commodity. Table 4.6(a) lists the specific values that were assumed here. Here, they are stylized values that roughly correspond to the characteristics of the building type. So for example, *residential* has a two to one ratio between thermal and electrical peak demand, while *industrial* has almost equal peak demand for both thermal and electrical energy. That said, the error on individual buildings will still be in the order of 10% to 100% for residential [17] or even higher for the individual industrial sites. It is beyond of the scope of this case study to improve upon the estimated demand data, but only use roughly indicative values. The gist from case study Munich west in section 4.2 was that it is not about absolute precision, but getting the scale and proportions of demand distribution roughly right in order to make reasonable planning decisions.

Although **rivus** does not put much emphasis on temporal resolution, a small amount of time steps with durations (weights) and scaling factors must be provided nevertheless. Using the method of discretising multiple load duration curves, as discussed in section 2.4.2, the time series of site *Zentrum* is reduced to five clusters that represent the most representative operation points for the network. They are summarised in table 4.6(b). *Peak electricity* and *peak heat* represent the (probably overestimated) durations of peak demand for the respective commodity. The three time steps *cold*, *dark* and *summer* reflect very typical load situations in a very stylised way. They capture the frequency patterns of how often which combination of loads roughly occur together. The table is not designed to represent actual load situations, but rather to correctly reflect the total power/energy ratios that occur over the course of a year. Therefore the scaling factors should *not* be confused with *typical* operation points of generation units, which might deviate from those values.



Table 4.6: Parameters for estimating spatial electricity and heat demand of Haag i. OB.  $h_{kC}$  is the specific peak demand per building type. Temporal duration  $w_t$  and magnitude  $s_{ct}$  scale the peak demands.

(a) Specific peak demands $h_{kC}$			(b) Peak scaling factor $s_{ct}$ , duration $w_t$		
Building type	Elec	Heat	Time step	Weight	Scaling $s_{ct}$
$k$	$h_{kC}$ (kW/m <sup>2</sup> )		$t$	$w_t$	Elec Heat
commercial	0.060	0.080	Cold	3055	0.41 0.42
farm	0.060	0.080	Dark	1339	0.63 0.10
garage	0.000	0.000	Summer	3708	0.32 0.07
hospital	0.110	0.110	Peak electricity	508	1.00 0.42
hotel	0.110	0.110	Peak heat	150	0.46 1.00
house	0.060	0.100			
industrial	0.095	0.110			
office	0.060	0.080			
other	0.040	0.040			
public	0.060	0.080			
residential	0.050	0.110			
restaurant	0.090	0.080			
retail	0.090	0.080			
school	0.060	0.080			

In total, those weights and scaling factors preserve both peak load and annual demand. The peak load is preserved thanks to the maximum scaling factor equal to one per commodity for each commodity. The annual demand is preserved by the values of the time step weights  $w_t$ , which in sum are equal to the original load curve.

When the specific peak demands are assigned to all buildings in the study area and then grouped to their nearest edge in the simplified street graph, the cumulated peak load for electricity and heat as shown in figures 4.15 and 4.16 follows. (Diezmanning, the small village between Haag and Moosham cannot be seen in these figures, though.) The total summed peak demand amounts to 26.7 MW for electricity and 47.6 MW for heat.

#### 4.4.2 Results model rivus

Figure 4.17(a) contains the optimal distribution network and conversion locations for a conservative scenario labelled TODAY, in which no electric heating and no heat pumps may be installed. The resulting topology resembles current infrastructure design patterns, in that electricity demand is mainly satisfied by a central electricity grid, while heat demand is satisfied by fossil fuels.

In contrast, figure 4.17(b) shows the infrastructure that results from allowing both heat pumps and electric heating. One can notice that in this scenario, domestic heat pumps the sole provider of thermal energy. At total capacity of almost 16 MW is installed to satisfy all heat demand. An artefact of the high ratio between gas and electricity price of 0.01 €/kWh to 0.3 €/kWh, together with an attractive low variable investment cost part of 60 €/kW (on top of a base price of 30 000 € per unit) leads to the construction of a 5 MW (thermal) gas power





Figure 4.16: Map of peak demands (kW) for electricity and heat in Moosham

plant for base load operation. It continuously outputs 1.3 MW to 1.8 MW electricity in all five time steps.

When heat pumps are excluded from the technology portfolio, as done in scenario NO HEAT PUMP, the result can be observed in figure 4.17(c). Still, electric heating, even if assumed to have *much* higher investment costs than seen today, is the cheaper option.

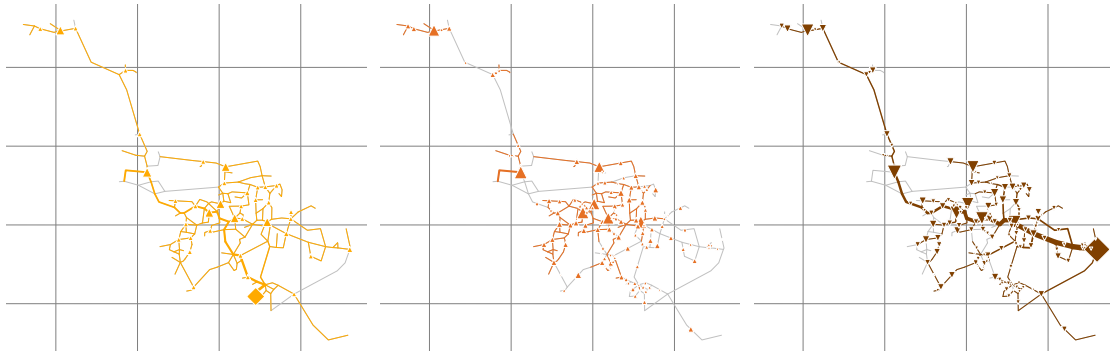
Once observed, the reason for this result becomes obvious: saving the investment costs for a second layer or third layer of distribution network is huge. Therefore, the observed results stay stable for a wide variety of cost assumptions (both commodity prices and investment cost changes). Electricity is used as the main energy carrier for transmission, electricity and heat supply for a wide range of cost parameters. The result is particularly stable if heat pumps are available, even though their coefficient of performance (COP) is assumed to be very low ( $< 2$ ).

Some caveats when interpreting this result: It completely assumes that electricity “is available” in sufficient quantity and quality (as in acceptable GHG emission). This might or might not be a realistic scenario in the future. Either way, these results are *not* directly applicable to real-world planning tasks, as they assume an omnipotent, all-knowing actor. They neglect personal preferences of consumers and any special requirements or existing infrastructure. District heating for example, if available already on site, still is an efficient use of power plant waste heat. Gas heating, if used with efficient equipment, still is on par with a much more expensive heat pump, that only marginally improves upon emissions, but for much higher costs. However, at the same rate that electricity becomes less GHG-intense, all power to heat technologies become more attractive.

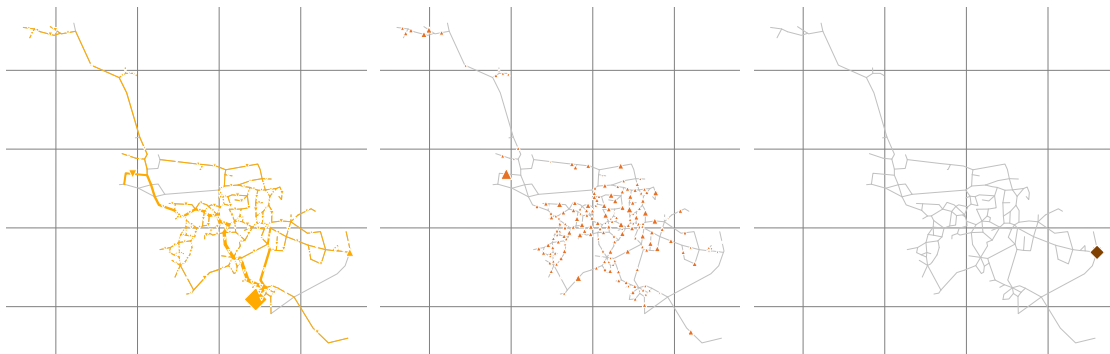
In brief, this result could motivate a more detailed rethinking of which kinds of energy infrastructure are to be installed in future developments of neighbourhoods and districts. Perhaps, a single (fortified for future demand increase) electric network might be sufficient for future developments.

#### 4.4.3 Data preparation for urbs

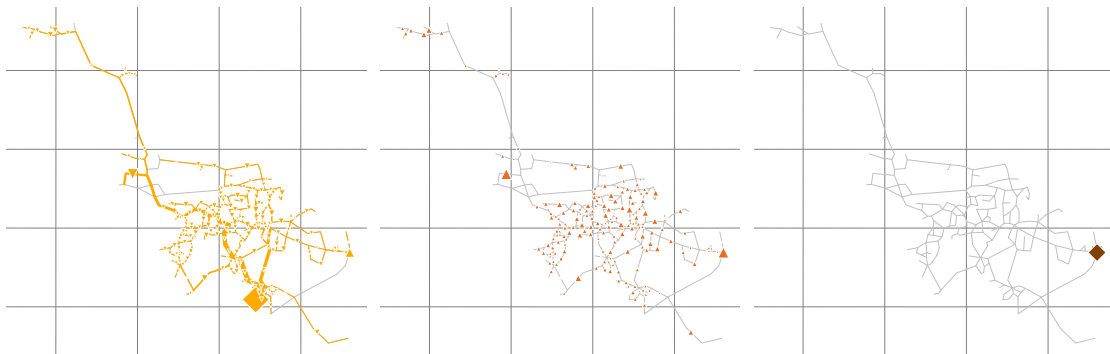
The focus of **urbs** is on temporal resolution, not spatial resolution. In order to transfer the capacities planned by **rivus**, they must be aggregated. Therefore, the installed capacities, as well as energy demand, must be reduced to a small number of sites. In this case study, the aggregation target is manually defined by assigning regions of more strongly connected edges to in total 12 so-called *clusters*. These clusters are shown in figure 4.18. If such an analysis were to be repeated multiple times, similar clusters could have also been derived automatically, either using a traditional cluster analysis or a minimum-cut graph-theoretic approach. The following paragraphs now describe how energy demand load curves, intermittent supply time series and installed capacities are assigned to those clusters.



(a) Scenario TODAY: Full electric grid, satisfying electricity demand; no electric heating is allowed. To satisfy heat demand, a full gas network is built, used both for local heating gas used for local heating and short range district heating.



(b) Scenario FUTURE: Heat pumps are allowed. Consequence: no gas network or district heating network are built. Instead, a stronger electric grid with local heat pumps are used to satisfy all demand. Gas is only used at the source.



(c) Scenario NO HEAT PUMP: identical to FUTURE, except that less efficient conventional electric heating is used. Gas is not used except at the source.

Figure 4.17: Transmission and conversion capacities in Haag for three scenarios



Figure 4.18: Map of 12 manually defined clusters for Haag i. OB. Vertices at edge borders form the boundaries between clusters.

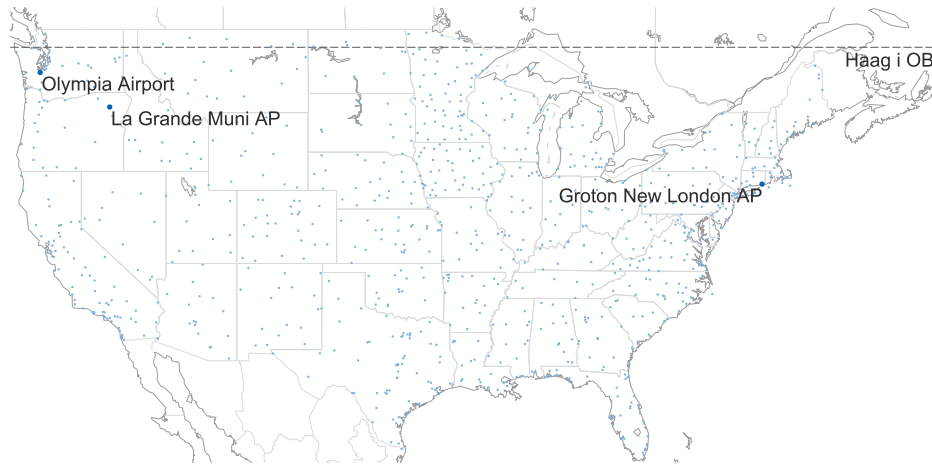


Figure 4.19: Three best matching TMY3 station locations (labelled dots) among all 1020 weather stations (unlabelled dots). The latitude of Haag i. OB (DE) is denoted by the dashed parallel line.

### Load curves

In Germany, the so-called *Standardlastprofile* are an established set load curves for various that could in principle used for electricity demand. Eight types for commercial loads (called G0 to G7), three types for agricultural loads (L0 to L2) and one type for residential loads are available. Unfortunately, for the purpose of this thesis, a simultaneous matching pair of electricity *and* heat load curves are needed. One way to proceed could have been to synthesise a thermal load curve using a simplified thermal simulation of a building. However, this was deemed to be beyond the scope of this thesis. Instead, another source for representative load profiles for electricity and heat demand was found:

The American Department of Energy (DOE) has published a huge dataset with load profiles [120] for 17 building types (16 commercial, 1 residential), synthesised for all the 1020 locations of the weather stations providing the time series for the TMY3 [85] dataset. Figure 4.19 shows these locations as dots on the map, as well as the latitude of Haag i. OB as a dashed line for reference. The idea now is to identify location(s), whose meteorological profile is most similar to that of Haag i. OB. The winner is location *Olympia Airport*, capital of Washington state, with United States Air Force (USAF) station number 727920.

Figure 4.20 shows an excerpt of the selected load curves for 12 of the 17 available building types. Table 4.7 shows which building area types (identical to those in peak demand table 4.6(a)) from the Haag dataset is assigned to which DOE load profile type. This mapping is called  $D_k$ , where  $k \in K$  is an element of the building type set  $K$ .  $K$  contains all OSM types, while  $D_k$  returns the corresponding DOE type. For example, for  $k = \text{"retail"}$ ,  $D_k$  returns the corresponding DOE type "supermarket". Only 7 of the 17 available types are actually used. The biggest issue with this whole process lies in the fact that typical buildings in Europe and USA are built from different materials and tend to have different appliances that are used differently. But as the focus of this study is not about using real, measured (and thus confidential) load time series, but rather indicative loads that exhibit the general trends of a building in a comparable

Table 4.7: Matching of OSM building types to DOE load types

OSM type $k$	→	DOE type $D_k$
commercial		medium office
farm		warehouse
garage		—
hospital		hospital
hotel		small hotel
industrial		large office
other		—
public		medium office
residential		residential
retail		supermarket
warehouse		warehouse

climate, this problem seems acceptable at this point. To summarise: to retrieve a single DOE load curve value for a given building type  $k$ , commodity  $c$  at time step  $t$ , the notation  $LC(D_k, c, t)$  is used in the following.

The next step is to combine and scale these load curves for multiple building types so that they correspond in size to the annual loads derived for model **rivus**. Mathematically speaking, this process aims to prepare data for the demand parameter  $d_{vct}$  for the commodities electricity and heat, with the following sets:

$$V = \{\text{Alpenstrasse}, \dots, \text{Zentrum}\} \quad (4.5)$$

$$C_d = \{\text{Elec}, \text{Heat}\} \quad (4.6)$$

$$T = \{1, \dots, 8760\}. \quad (4.7)$$

To calculate a single value of the demand time series  $d_{vct}$ , first some more parameters must be introduced: the area of all buildings within a cluster  $v$  of a certain (OSM) building type  $k$  is referred to as  $A_{vk}$ . The specific areal peak demand of that building type for a commodity is called  $h_{kc}$  (W/m<sup>2</sup>) and listed in table 4.6(a).

$$\forall v \in V, c \in C, t \in T: \quad d_{vct} = \sum_{k \in K} A_{vk} h_{kc} LC(D_k, c, t) \quad (4.8)$$

### Fluctuating renewable energy sources

In order to not only see the processes that were planned by **rivus** satisfying the demands they were sized to satisfy, **urbs** is allowed to invest in renewable energy sources, notably photovoltaics and wind parks. Photovoltaics are allowed in all clusters, while wind parks may only be constructed in the outer clusters Pipeline, Rainbachstrasse, Suedwesten and Umspannwerk. The used time series are shown figure 4.21, provided for the location from a global dataset of capacity factor time series [54], as described in section 2.6.3. For Haag, the solar time series has (conservative) annual full load hours of 750 h, while the wind time series has progressive 2700 h and thus overestimates the wind potential available to a town in Bavaria. The maximum allowed capacities for photovoltaic plants and wind turbine are



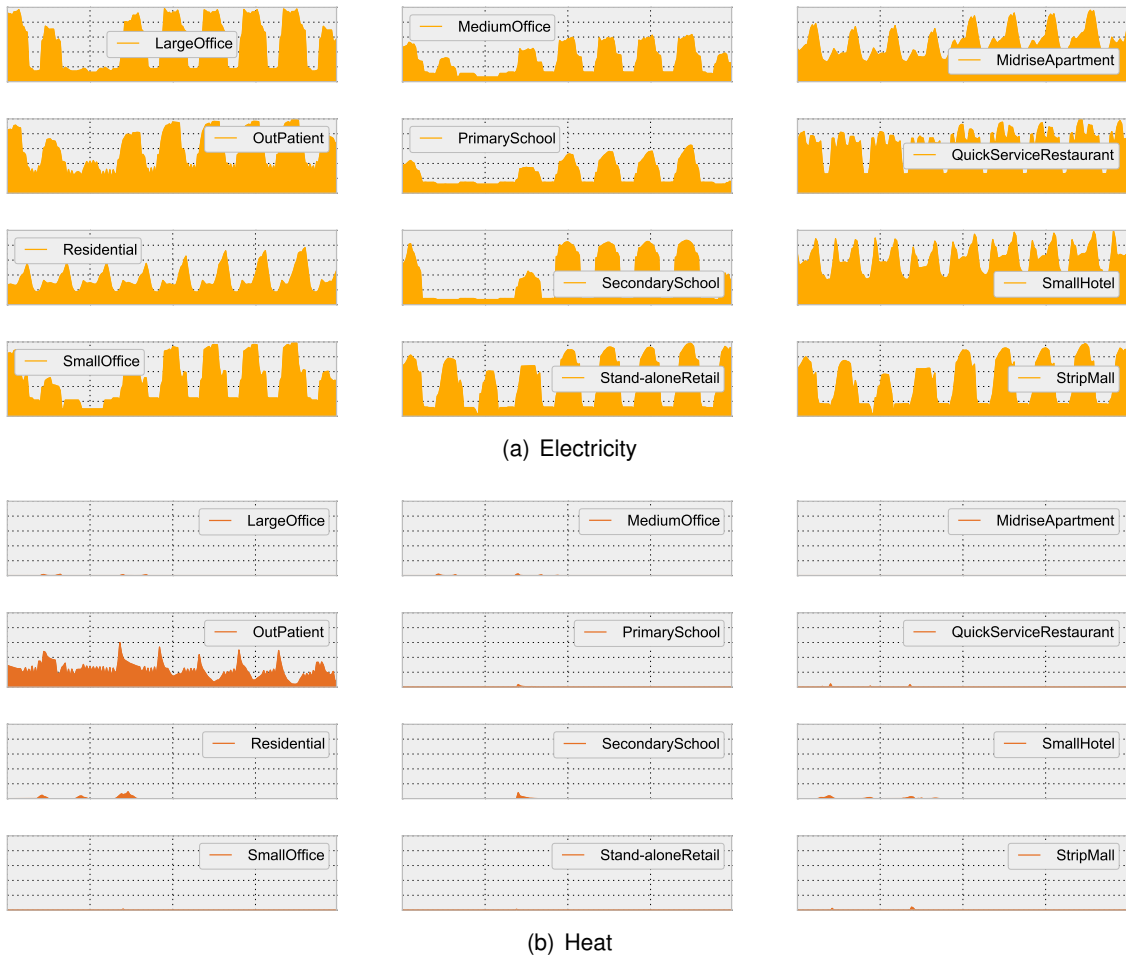


Figure 4.20: Exemplary normalised load curves for TMY3 station Olympia Airport from Jul 28 to Aug 5, plotted in the range 0 to 1 for 12 of the 17 DOE building types

shown in 4.8. The values for photovoltaics are rough estimates from a visual inspection of the different clusters. The values for wind turbines are set to (arguably) high 2 MW to 5 MW at the outwards clusters of Haag; it is not expected that Haag will deploy that much wind power. The assumption is still made to maximise the possible benefit that a storage system could yield, by increasing the allowed amount of fluctuating electricity generation.

### Process & transmission capacities

The installed capacities of conversion processes and transmission between clusters must be aggregated from the street graph to the cluster level. For conversion processes, the sum of all capacities on edges/vertices within that cluster are aggregated. Due to the definition of the clusters being based on edges, there is ambiguity towards the assignment of capacities constructed on those vertices. A deterministic assignment can be made based on the number of edges that start/terminate at that vertex per cluster. If this criterion does not yield a winner, an arbitrary (but deterministic) choice is made.



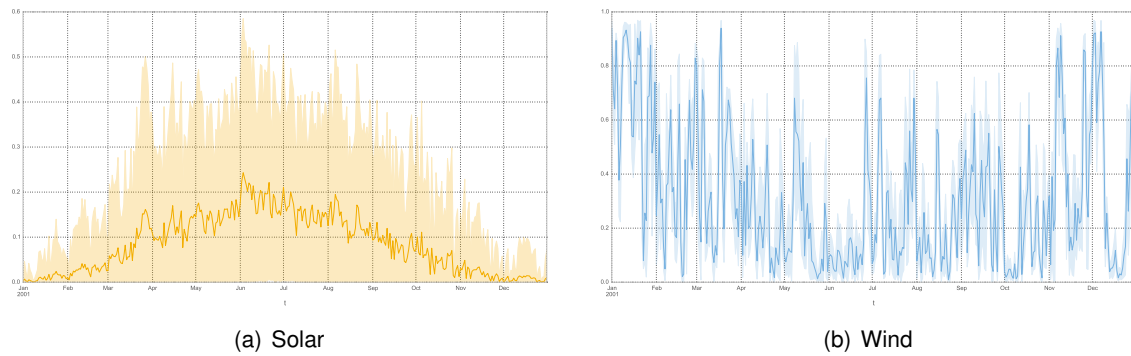


Figure 4.21: Capacity factor time series of intermittent renewable energy sources (normalised to installed capacity). Plot shows daily average (line) and minimum/maximum (area) values, while the raw data has hourly resolution.

Table 4.8: Maximum allowed fluctuating renewable capacities (kW) for Haag i. OB

Cluster	Photovoltaics	Wind turbine
Alpenstrasse	300	0
Diezmanning	50	0
Einzelhandel	200	0
Moosham	100	5000
Nord	200	0
Pipeline	300	5000
Rainbachstrasse	400	2000
Reiterstrasse	200	0
Schletter	100	0
Suedwesten	200	3500
Umspannwerk	100	5000
Zentrum	800	0

For the transmission capacities, the sum of all edge capacities that *cross* the boundary between neighbouring clusters<sup>3</sup> Because it contains the lowest capacities for electricity transmission, scenario TODAY is used as the basis for both scenarios whose results are presented in the next section.

With these assumptions in place<sup>4</sup>, model **urbs** is run on the aggregated input data set. This discussion only covers two of the finally six scenarios<sup>5</sup> that were kept in the final scenario selection.

<sup>3</sup>The full analysis script is available as an IPython notebook [rivus-cluster-aggregation.ipynb](#) with inline documentation on <https://gist.github.com/ojdo>.

<sup>4</sup>The input parameters are documented in file [ojdo/urbs/haag15.xlsx](#) of branch *haag15*.

<sup>5</sup>The full scenario definitions are documented as code in file [ojdo/urbs/rivhg15.py](#) of branch *haag15*.

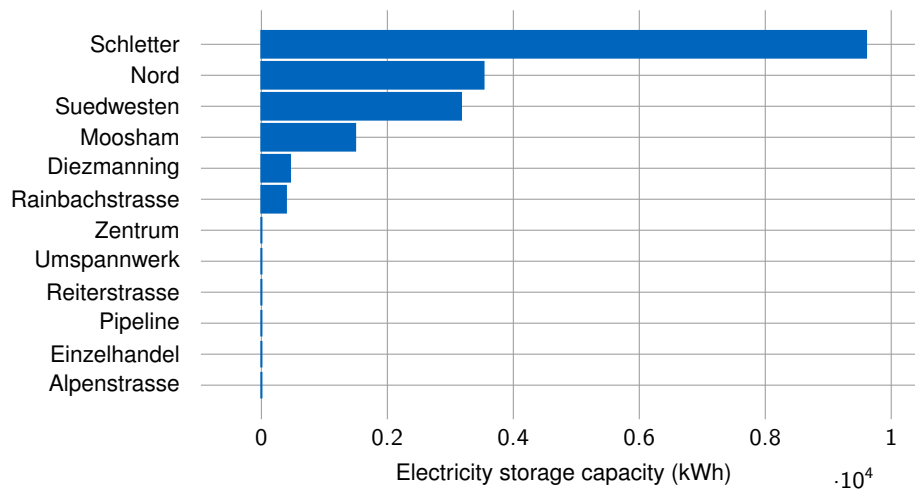


Figure 4.22: Installed battery storage capacity by cluster in scenario CHEAP BATTERY

#### 4.4.4 Results model urbs

Table 4.9 shows the tabulated energy generation and consumption for the scenarios BASE and BATTERY CHEAP. Both tables are grouped by cluster (columns) and process (rows), together with energy transmission among and storage within clusters. The red part of the diagram deals with heat, the yellow part of the diagram with electricity. Due to the low investment cost for renewable energies, the maximum allowed capacities for both photovoltaic and wind energy are exhausted, leading to a significant share of fluctuating available energy production. This becomes important when assessing the use of storage. Common to both tables, the first three rows show the annual summed consumption for those commodities. Consumption appears for demand of electricity and heat. Electricity has a second consumer, *heat pumps*. The next block then describes the summed *creation* of both demands from conversion processes. For heat, the main generators are district heating plants, followed by heat pump plants.

In both scenarios, the district heating plants installed near the centre of Haag (cf. figure 4.17(a)) generate most heat. In the other clusters, heat pumps are the main process for heat generation. Cluster *Zentrum* also contains the generation for four of its five neighbouring clusters.

In scenario BASE, the large amount of fluctuating renewable electricity leads to noticeable amount of unused electricity generation, as the case study model does not include the possibility to sell excess energy back to an external market.<sup>6</sup>

An illustrative excerpt of the original form of the results is given in appendix C. It shows the hourly composition of energy conversion, transmission and storage is shown for all twelve clusters, both electricity and heat during a single week, or 196 time steps. The full optimisation duration is a full year, or 8760 time steps.

Figure 4.22 shows the installed battery storage capacity by cluster in scenario CHEAP BATTERY, in which investment costs for battery storage is reduced by 95% compared to the base scenario assumption of 1500 €/kWh (annualised with  $n = 15$  a,  $i = 5$  %).

<sup>6</sup> Another model user has contributed this feature in the meantime, now documented under the name *Buy-Sell*. It allows to define price time series for buying or selling commodities.



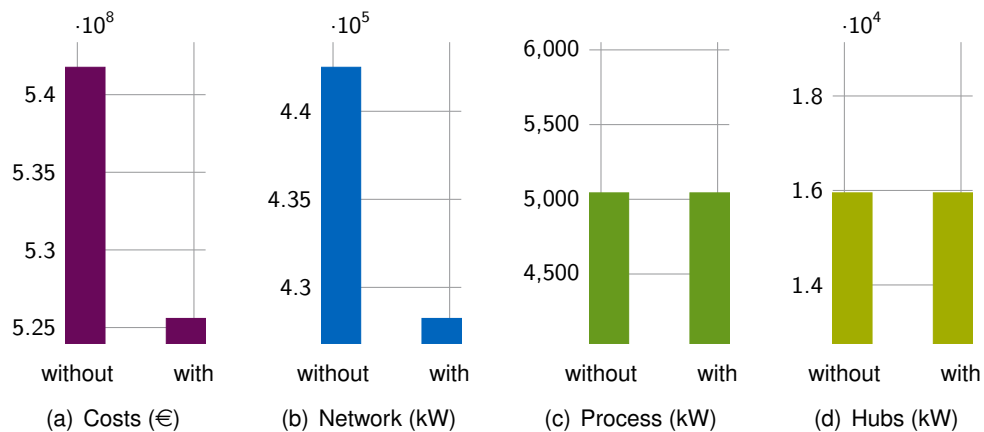


Figure 4.23: Comparison of cumulated installed capacities with and without peak reduction

#### 4.4.5 Closing the loop

The use of battery capacities installed in scenario CHEAP BATTERY are now compared to the load in the respective cluster. By calculating the average usage of power retrieved from battery storage relative to the simultaneous electric load<sup>7</sup>, an upper bound for the peak demand reduction can be derived. In the analysed scenario, storage leads to an average reduction of the load in the range from 5 % to 10 % per cluster. A cursory investigation of the time series yields that this reduction happens mainly in times of high demand, but not always. If one optimistically assumes that the average reduction equals an identical figure in terms of peak demand reduction, the original peak demand estimated for model **rivus** can be reduced, the optimisation be relaunched and a possible reduction in network expansion costs be quantified.

With those values, the per-edge peak demand is reduced according to the containing cluster. In the investigated scenario, the battery storage use manages to reduce the system cost of the system planned by **rivus** by 3.0%. This happens by reducing the cumulated installed pipe capacity by 14.3 MW (3.2%), as shown in figure 4.23.

#### 4.4.6 Sensitivity analysis: battery investment costs

To give more context how sensitive the installation of battery storage capacity is to changes of investment costs, an additional investigation is performed. To better reflect the (lack of) potential flexibility of distributed heat generation, the additional feature *proportional processes* are implemented as a case study specific model modification in model **urbs**. The following section first outlines the change and its implementation, then the results of the sensitivity analysis are presented.

##### Proportional processes

This section describes a case-study specific modification that was performed on model **urbs** after the co-optimisation study described above. The change is related to the local processes

<sup>7</sup>The exact analysis and result is documented in the IPython notebook [urbs peak demand reduction.ipynb](https://gist.github.com/ojdo) on <https://gist.github.com/ojdo>.

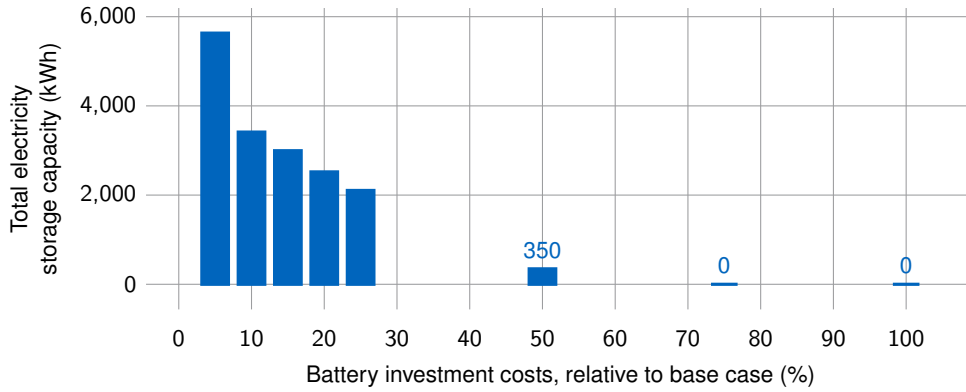


Figure 4.24: Installed battery storage capacity by investment costs for 5 % to 100 % of base case investment costs.

coined *hub processes* in the context of model **rivus**. Hub processes represent small, distributed energy conversion units like domestic room heating or domestic heat pumps. The unmodified model **urbs** cannot discern big, central units whose operation can be controlled from small, distributed processes whose operation cannot be planned by a central entity (at least for the moment). Instead, these units are assumed to always meet “their” fraction of the aggregated demand curve of the cluster (vertex) they are located in.

For the three processes *Elec heating domestic*, *Heat pump domestic*, *Gas heating domestic* their main output, i.e. commodity *Heat* is deemed directly following the respective load in each cluster *proportionally*. These processes are summarised in the set  $P_{\text{prop}}$ . However, which *fraction* of the load is met by the process can be freely chosen by the solver and is captured by the additional continuous variable  $\delta_{vcp} \in [0, 1]$ . This rule is equivalent to the following additional constraint, which is added to function `create_model`<sup>8</sup>:

$$\forall t \in T_m, v \in V, c \in C_{\text{dem}}, p \in P_{\text{prop}}: \quad \epsilon_{vcpt}^{\text{out}} = d_{vct} \delta_{vcp} \quad (4.9)$$

With this new constraint in place, the operation of any installed domestic heat generation units is fully determined by the associated demand time series, similar to how the operation of processes with *intermittent* input commodities is determined by their capacity factor time series.

## Result

Figure 4.24 shows the cumulated installed battery capacities over the relative change in battery investment costs. The bar at value 5 % correspond to the previous BATTERY CHEAP scenario. One can notice that the total installed capacity is reduced from over 20 000 kWh to just under 6000 kWh. This is caused by the reduced flexibility of the system, triggered by the less flexible decentralised proportional processes. As expected, higher investment costs lead to a monotonous decrease in installed storage capacity. At 75% of the base case costs, no more capacity is installed. Before that, the remaining installation more and more concentrates in the isolated cluster *Moosham* and, below 20 %, *Suedwesten*.

<sup>8</sup>The full implementation is available in branch *haag15*, commit 0257631 in repository *ojdo/urbs* on GitHub.

Concerning the storage usage, the change to proportional processes has significant implications. The remaining storage capacity is only used as backup, leading to very low number ( $< 10$ ) of annual charging cycles. However, this finding is highly depending on the less firm assumptions of this case study (fluctuations in weather time series, amount and type of renewable generation), and thus warrants more investigation for an independent case study with fewer free parameters.

#### 4.4.7 Discussion

In this case study, an optimisation case study for Haag i. OB has been presented. A hybrid approach demonstrating the combined use of two models has been demonstrated. First, energy conversion and transmission technologies were sized using model **rivus** with high spatial resolution. Then, using these results as input for model **urbs**, energy storage technologies were sized and scheduled with high temporal resolution.

The most limiting factor for the conclusions to be drawn for the *actual* case study region is the large amount of assumptions that needed to be taken in order to generate only the demand data. If that data would be known, the remaining sensitivities (mainly concerning technology costs and efficiencies) could be handled. Additionally, the presented scenarios are selected to demonstrate the maximum effect that certain scenario assumptions could make.

That said, one result remains robust against all investigated changes: if green-field planning is allowed and the availability of (off-site) electricity is sufficient, the additional effort of creating a second or third “layer” of infrastructure merely for room heating seems not to be favourable from a planner’s perspective. Even if natural gas is made cheaper, it just increases the amount local electricity generation. In other words: the status quo of routinely installing gas lines and electricity networks is more a result of the retail prices for electricity compared to gas rather than an economic optimum.

## Chapter 5

# Conclusion

*The conclusions of most good operations research studies are obvious.*

---

Robert E. Machol (1917–1998)

This thesis has presented in total five related optimisation models for planning future and operating current energy systems for minimum costs. Four case studies have demonstrated that these models can be used for various planning tasks and identify *stable solutions* in the design space of potential future energy systems.

### 5.1 Summary

The central insight is that conceptually simple optimisation models can be pushed very far, with the benefit of being able to analyse interconnected systems with many small decisions, which – by themselves – are modelled with the absolute minimum complexity. By not taking a single optimisation result of such a model at face value, but by abstracting the results over a range of inputs (*scenarios*) and drawing conclusions over result properties that remain stable (or unstable) over those ranges, more reliable conclusions can be drawn concerning the properties of the design space at large.

The first two case studies in Frankfurt and the west of Munich have shown that central heat generation can be cost-efficient when planned individually without considering the other energy carrier networks. Furthermore, they demonstrate that simple MILP models based on estimated heat demands are sufficient to draft network plans that resemble those created by expert planners. The author conjectures that such optimisation models could play an important role in the pre-planning phase of the planning process for urban infrastructure development.

The result of the final study in Haag allows one tentative conclusion: once renewable electricity becomes available in large quantities, electric heat generation through heat pumps could render all other energy distribution networks superfluous. While the economics look different in today's existing building stock and with today's electricity mix, a greenfield planning scenario with CO<sub>2</sub>-lean electricity mix is hard to convince to plan more than one energy distribution network. Under such a regime, only very cheap electricity storage technologies could prevail against the ubiquitous option of “storing” electricity in form of thermal energy.

Again, this result is only valid under the anticipated move away from fossil fuels in electricity generation and where no gas network is pre-existing.

## 5.2 Limitations

Naturally, conclusions drawn from highly simplified models do not come without caveats. Foremost, commodities are modelled as **opaque energy carriers** without technical properties: no voltage or current for the electricity grid, no temperature or fluid dynamics for district heating pipes, no pressure drops and flow speeds for gas networks are modelled. Adding energy carrier specific equations to these models could greatly increase applicability (and complexity) of the resulting models. The focus of this thesis was to create openly available, well documented and *generic* energy models, so that these tasks could be carried out on a stable foundation.<sup>1</sup>

Another limitation for the planning decisions drawn by the presented models concerns the omission of **inter-temporal optimisation** for investment decisions. In this thesis, only optimal snapshots for target system configurations are shown. Sampling points for identifying ideal technology investment *pathways* to reach those targets are left open.

## 5.3 Outlook

The two published models **urbs** and **rivus**, as well as the documented modelling workflow and data analysis scripts hope to make a contribution to the energy system modelling community. The author hopes that these models will find their place in the growing ecosystem of open source optimisation models. The development of other open source energy optimisation models in organisations like **openmod** will play an important role in increasing the reproducibility and maintainability. By making it easier for following researchers to independently verify current modellers' findings, the standards for future research can improve in the long run.

---

<sup>1</sup>Linearised DC load flow equations for power networks [19] are currently being implemented as an optional feature for transmission technologies in **urbs** (issue #41).



# Appendix A

## Common model structure

This appendix presents the underlying structure that is shared by the three models **dhmin**, **urbs**, and **rivus**. This structure consists of the directory layout, common file names, code structure, input data format and function names in the model code.

### A.1 File structure

This is the minimum file structure shared by all models. The word `modelname` is replaced by the actual model name. Model **urbs** uses the minimum form of this template with only three files. It consists of an input data spreadsheet, the model itself and a run script:

---

```
1 urbs/  
2   example.xlsx  
3   urbs.py  
4   runexample.py
```

---

The two models **dhmin** and **rivus** consist of more files. They store their geographic input in a subdirectory `data/location` with two files `vertex.shp`, `edge.shp`<sup>1</sup>. **rivus** also has the additional `building.shp`, which is used for aggregating the energy demand from building to edge. The tabular data file `data.xlsx` is consequently moved to the same location, resulting in the following structure:

---

```
1 dhmin/  
2   data/mucwest/  
3     data.xlsx  
4     edge.shp  
5     vertex.shp  
6   dhmin.py  
7   runmucwest.py
```

---

---

```
1 rivus/  
2   data/haag/  
3     building.shp  
4     data.xlsx  
5     edge.shp  
6     vertex.shp  
7   rivus.py  
8   runhaag.py
```

---

---

<sup>1</sup>Actually, the shape files consist of multiple physical files each. See section 2.6 for more information.

Table A.1: Attribute-value system example

Domain 1	...	Domain $M$	attribute 1	...	attribute $N$
South		Stock	25		70
South	...	Env	—	...	999
South		Demand	256		33
Mid		Stock	30		65
Mid	...	Env	—	...	888
Mid		Demand	1024		12
North		Stock	30		65
North	...	Env	—	...	44
North		Demand	512		18

## A.2 Data format

All tabular data in the spreadsheet `data.xlsx` is represented in all models as multiple sheets. Each sheet has the same form, called **attribute-value system**. Its general structure is demonstrated in table A.1.

$M$  domain columns form unique combinations of tuples, e.g. (South,..., Stock), that address unique rows or *objects*, for which then up to  $N$  attribute values may be defined. Depending on the tool chain, attribute values must either be numeric or can have arbitrary types (strings, blobs). By convention, the domain columns start with an uppercase character, while the attribute columns start with a lowercase character. The background colours are used for easier visual identification, but have no function and thus can be changed at will..

This tabular form is used because it is closely related to normalised forms of database tables, making it easier to switch the storage format from spreadsheet files to a database system, should the necessity arise. Additionally it can be represented and manipulated easily in most spreadsheet applications. Even attribute tables of shapefiles (stored in the file ending `.dbf`) can be made to conform to this format, by identifying some of the column names as domain columns.

In order to better scale to larger data sizes, switching to HDF as a storage backend could be a favourable move. First, it features compression, fast read and write performance even for gigabyte-sized datasets. Second, the format enjoys good support across most scientific computing environments. For example, Matlab's own `.mat` format switched to using HDF5 for its storage backend for performance and compatibility reasons. Refer to section 2.5.2, more alternatives are discussed.

## A.3 Model

All three models file share the code structure shown in listing A.1. A function `read_input` (or `read_excel`) reads all input data from a given input filename, whose structure conforms to the data format described in the previous section. This function always returns one or multiple pandas DataFrame objects. These input data structures then can be either passed unmodified to the following `create_model` function, or modified to create scenarios.

This leads to the following minimal life cycle for the optimisation problem object `prob`, here exemplified for model `urbs`:

---

```
1 import urbs
2 data = urbs.read_input('example.xlsx')
3 model = urbs.create_model(data)
4 prob = model.create()
5 result = optim.solve(prob)
6 pro.load(result)
7
8 costs = urbs.get_entity(prob, 'costs')
9 cap_pro = urbs.get_entity(prob, 'cap_pro')
```

---

The last two lines demonstrate how to retrieve optimisation results as a DataFrame. In order to simplify this work, most models also implement additional `get_*` functions like `get_constants` for costs and capacities, and `get_timeseries` for time-dependent variables.

For retrieving multiple entities with identical domains, helper function `get_entities` is provided. It takes a list of set, parameter, variable or constraint names and returns a consolidated DataFrame. In order to discover all valid entity (i.e., set, parameter, variable, objective, constraint) names for a model, helper function `list_entities` is provided. It returns a list of names, domains and docstrings for the given entity type.

For archiving and restoring a problem instance before or after solving, the serialisation and deserialisation functions `save` and `load` are provided. They rely on Pickle, the standard Python protocol for object serialisation. To reduce file size, the files are transparently compressed and decompressed using the GZip library. Note that this feature is not optimised for speed. A model-specific data storing technique, possibly relying on HDF to save only relevant variable values, could reduce optimisation runtime considerably. However, as using Pyomo itself also is not optimal for highest performance, no effort was spent here to optimise this particular feature.

## A.4 Run script

While the previous model structure is rather stable and only to be changed for model developers, this section describes the common structure of a user run script, that catches the particulars of a specific case study. First it defines the input file name, scenarios (here implemented as functions, but other options are possible as well) and the desired result post-processing steps. An exemplary run script for model `urbs` is presented in listing A.2.

Listing A.1: Common structure of model files `dhmin.py`, `urbs.py`, `rivus.py`


---

```

1 import coopr.pyomo as pyomo
2
3 def read_input(filename):
4     ''' reads input file and possible pre-process steps '''
5     # e.g. calculation of a derived attribute value
6     data['commodity']['annuity-factor'] = annuity_factor(...)
7     return data
8
9 def create_model(data, **options):
10    ''' creates a pyomo optimisation problem instance '''
11    model = pyomo.ConcreteModel()
12    model.time = pyomo.Set(initialize=data['timesteps'].index)
13    model.com = pyomo.Set(initialize=data['commodity'][0].index)
14    model.weight = pyomo.Param(model.time, initialize=data['weights'].values)
15
16    model.x = pyomo.Var(model.time, within=pyomo.Reals)
17    model.obj = pyomo.Objective(model.time, model.com, rule=def_objective_rule)
18    # [more declarations]
19    return model
20
21 def def_objective_rule(m, t):
22    ''' example objective function '''
23    return m.obj = sum(m.weight[t] * m.x[t] for t in m.time)
24
25 def res_some_constraint_rule(m, t, c):
26    ''' example constraint rule '''
27    # dataframes can be used in constraints directly, without
28    # declaring them as Pyomo parameters
29    return (m.x[t, c] <= data['upper_limit'].loc[t, c])
30
31 # [more constraint rule functions]
32
33 # technical helper functions (identical for all models)
34 def list_entities(problem, entity_type)
35 def get_entity(problem, name)
36 def get_entities(problem, names)
37 def save()
38 def load()
39
40 # canonical helper functions (custom for each model)
41 def plot()
42 def report()
43 def get_constants()
44 def get_timeseries()

```

---

Listing A.2: Exemplary run script `runexample.py`


---

```

1 import urbs
2 from coopr.base import SolverFactory
3
4 def scenario_base(data):
5     ''' base scenario: don't change input '''
6     return data
7
8 def scenario_cheap_solar(data):
9     ''' cheap solar: pv investment -50% '''
10    data['process'].ix['pv', 'inv-cost'] *= .5
11    return data
12
13
14 def run_scenario(filename, scenario, timesteps):
15     ''' perform optimisation for given scenario '''
16
17     # prepare input data
18     data = urbs.read_input(filename)
19     data = scenario(data)
20
21     # create problem, run solver, read output
22     model = urbs.create_model(data, timesteps)
23     prob = model.create()
24     optim = SolverFactory('glpk')
25     result = optim.solve(prob, tee='True')
26     prob.load(result)
27
28     # reporting and plotting
29     sce = scenario.__name__
30     urbs.report(prob, '{}-{}.xlsx'.format('report', sce))
31     for ext in ['png', 'pdf']:
32         urbs.plot(prob, '{}-{}.{}'.format('plot', sce, ext))
33     urbs.save(prob, '{}-{}.pgz'.format('problem', sce))
34
35     return prob
36
37 for __name__ == '__main__':
38     filename = 'example.xlsx'
39     timesteps = range(0,8760)
40
41     scenarios = [scenario_base, scenario_cheap_solar]
42
43     for scenario in scenarios:
44         prob = run_scenario(filename, scenario)

```

---



## Appendix B

# MathProg storage sizing model

This chapter presents the full text of the abridged listing 2.2 in the presentation of modelling languages in section 2.5.1. It has been used as an introductory model on teaching basics of linear programming and energy system model in two lectures.

The model determines optimal size and operation of a generic energy storage technology (think: battery). The model assumes a given electricity demand curve, as well as an optional fluctuating supply (think: photovoltaics) of electricity. Additional power can be bought any time for a time-varying, yet pre-defined electricity price. The formulation implicitly assumes perfect foresight on the side of the battery owner. Solutions thus are always best-case feasibility boundaries which – depending on the insecurity of the price forecast – can either be reached almost or not even close.

The full code, including artificial demand, supply and price data for a single day in hourly resolution is given in listing B.1 on the following pages.

Listing B.1: MathProg example model for storage sizing `soforsg.mod`

```

1 # S0forSG: Storage Optimization for Smart Grids
2 # Last updated: 14 November 2013
3 # Author: johannes.dorfner@tum.de
4 #
5 # This model optimizes size (storage_capacity) and operation (storage_level[t])
6 # of a hypothetical lossless storage technology for electric energy. A given
7 # electricity demand[t] must be satisfied from a) a cost-free (renewable) energy
8 # supply[t] with intermittent characteristic or from b) electricity_purchase[t],
9 # i.e. buying of electricity from the grid for a time-dependent
10 # electricity_price[t].
11 # SETS & PARAMETERS
12 set time;
13 param demand{time} >= 0; # (kWh/h)
14 param supply{time} >= 0; # (kWh/h)
15 param electricity_price{time}; # (EUR/kWh)
16 param storage_cost; # (EUR/kWh)
17 param selling_price_ratio; # (1) for sold energy, relative to electricity_price
18
19 # VARIABLES
20 var energy_balance{time}; # (kWh)
21 var storage_capacity >= 0; # (kWh)
22 var storage_level{time} >= 0; # (kWh)
23 var energy_purchase{time} >= 0; # (kWh)
24 var energy_sold{time} >= 0; # (kWh)
25 var costs;
26
27 # OBJECTIVE
28 minimize obj: costs;
29
30 # CONSTRAINTS
31 # total costs = investment for storage + purchased electricity
32 s.t. def_costs:
33     costs =
34     storage_cost * storage_capacity +
35     sum{t in time} electricity_price[t] * energy_purchase[t] -
36     sum{t in time} electricity_price[t] * energy_sold[t] *
37     selling_price_ratio;
38 # balance = supply - demand + purchase - sold
39 s.t. def_balance{t in time}:
40     energy_balance[t] = supply[t] - demand[t] +
41     energy_purchase[t] - energy_sold[t];
42
43 # new storage level = old storage level + balance
44 s.t. def_storage_state{t in time: t>1}:
45     storage_level[t] = storage_level[t-1] + energy_balance[t];
46
47 # storage is filled 50% at beginning

```



```

48 s.t. def_storage_initial{t in time: t=1}:
49     storage_level[t] = 0.5 * storage_capacity;
50
51 # storage must be filled at least 50% in the end
52 s.t. res_storage_final{t in time: t=card(time)}:
53     storage_level[t] >= 0.5 * storage_capacity;
54
55 # storage may be filled at most to storage capacity
56 s.t. res_storage_capacity{t in time}:
57     storage_level[t] <= storage_capacity;
58
59 # limit sold energy to prevent unbounded model
60 s.t. res_energy_sold{t in time}:
61     energy_sold[t] <= 999;
62
63 # SOLVE
64 solve;
65
66 # OUTPUT
67 printf "RESULT\n\n";
68 printf "Costs: %+5.1f EUR\n", costs;
69 printf "      (%+5.1f EUR for %g kWh storage at %g EUR/kWh, \n",
70     storage_cost*storage_capacity, storage_capacity, storage_cost;
71 printf "      %+5.1f EUR for purchasing %g kWh,\n", sum{t in time}
72     electricity_price[t] * energy_purchase[t], sum{t in time}
73     energy_purchase[t];
74 printf "      %+5.1f EUR from selling %g kWh)\n\n", - sum{t in time}
75     selling_price_ratio * electricity_price[t] * energy_sold[t], sum{t in time}
76     energy_sold[t];
77 printf "%2s:\t%6s\t%6s\t%5s | %5s\t%5s\t%5s\n",
78     "t", "demand", "supply", "price", "Level", "Purch", "Sold";
79 printf "-----+-----\n";
80 printf{t in time}: "%2i:\t%6g\t%6g\t%5g | %5g\t%5g\t%5g\n",
81     t, demand[t], supply[t], electricity_price[t], storage_level[t],
82     energy_purchase[t], energy_sold[t];
83 printf "-----+-----\n";
84 printf "%s:\t%6g\t%6g\t%5s | %5s\t%5g\t%5g\n", "Sum",
85     sum{t in time} demand[t], sum{t in time} supply[t], "---", "---",
86     sum{t in time} energy_purchase[t], sum{t in time} energy_sold[t];
87 printf "%s:\t%6s\t%6s\t%5.1f | %5.1f\t%5s\t%5s\n", "Avg",
88     "---", "---", sum{t in time} electricity_price[t]/card(time),
89     sum{t in time} storage_level[t]/card(time), "---", "---";
90 printf "%s:\t%6i\t%6i\t%5i | %5i\t%5i\t%5i\n", "Min",
91     min{t in time} demand[t], min{t in time} supply[t],
92     min{t in time} electricity_price[t], min{t in time} storage_level[t],
93     min{t in time} energy_purchase[t], min{t in time} energy_sold[t];
94 printf "%s:\t%6i\t%6i\t%5i | %5i\t%5i\t%5i\n", "Max",
95     max{t in time} demand[t], max{t in time} supply[t],
96     max{t in time} electricity_price[t], max{t in time} storage_level[t],
97     max{t in time} energy_purchase[t], max{t in time} energy_sold[t];

```

```
93 printf "\n\n";
94
95 # DATA
96 data;
97
98 param storage_cost := 3; # storage capacity cost (EUR/kWh)
99 param selling_price_ratio := 0.5; # ratio of electricity price (1) for sold
    energy
100
101 param: time: supply demand electricity_price :=
102     1   0   0   0
103     2   0   1   2
104     3   0   1   2
105     4   0   1   2
106     5   0   1   2
107     6   0   1   2
108     7   1   2   1 # supply begins
109     8   3   5   2
110     9   6   2   2
111    10   5   1   1
112    11   8   1   1
113    12   9   5   2
114    13   9   0   1
115    14   6   0   1
116    15   8   0   1
117    16   7   0   1
118    17   6   6   5 # demand peak, price peak begins
119    18   3   9   5
120    19   2   6   5 # price peak ends
121    20   1   3   1 # supply ends
122    21   0   4   1
123    22   0   4   1
124    23   0   1   1
125    24   0   0   1;
126 end;
```

---

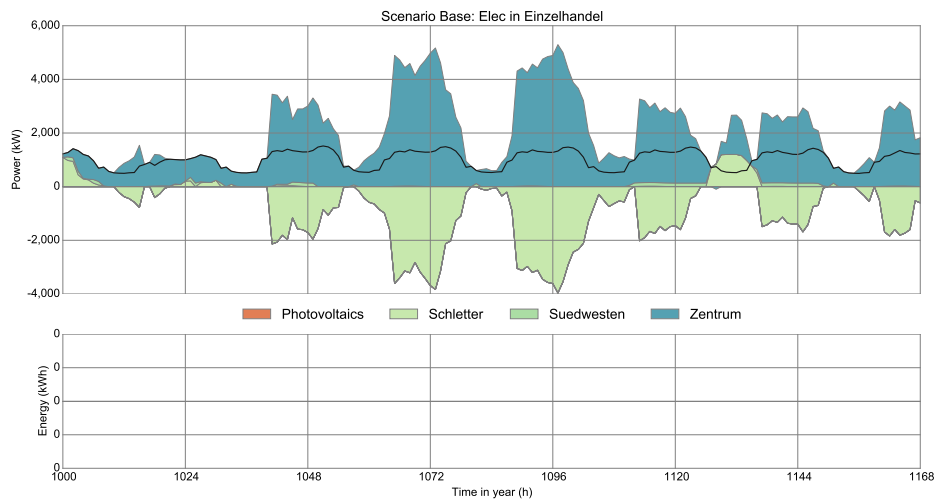
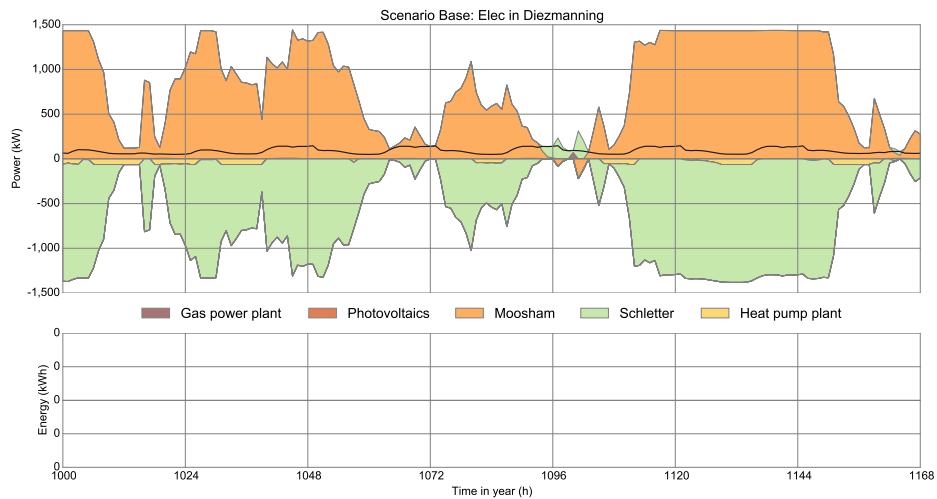
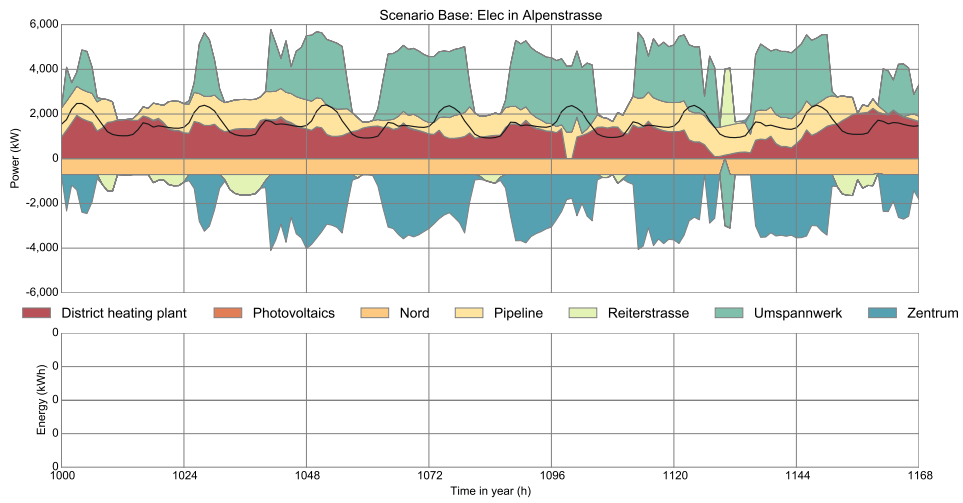
## Appendix C

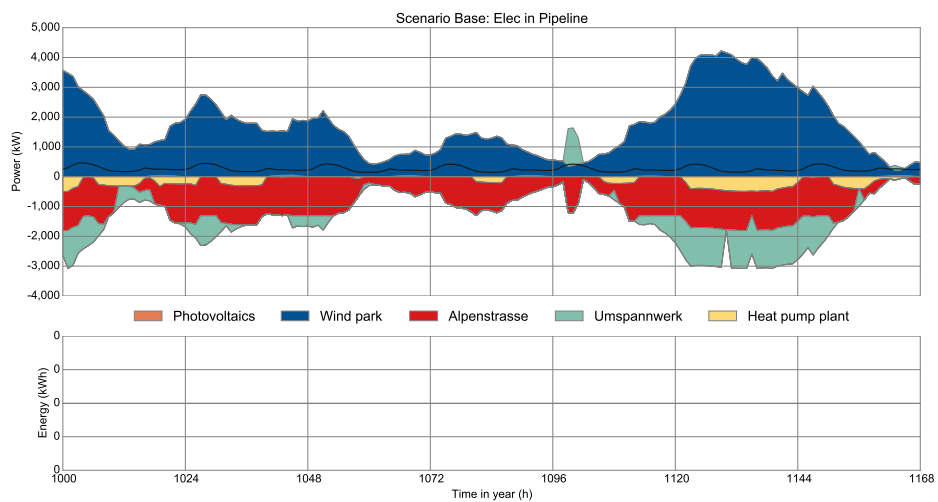
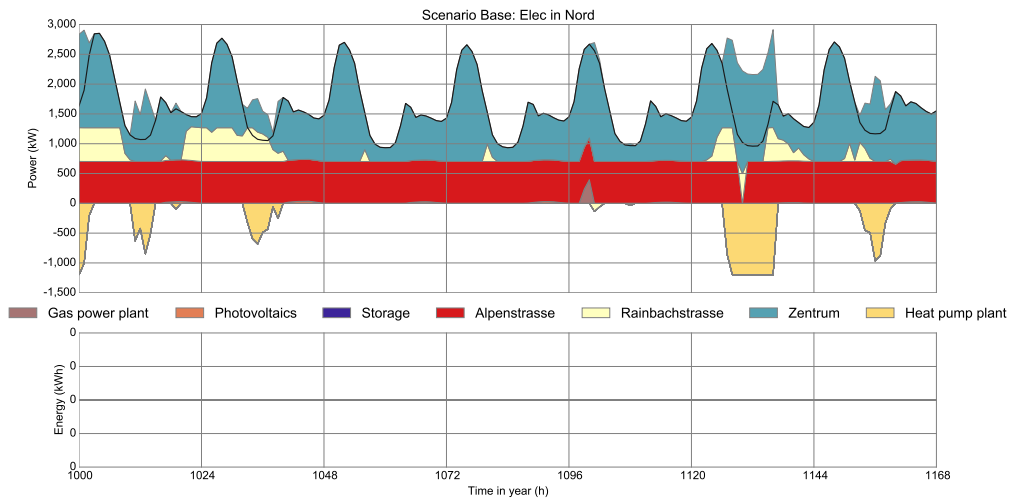
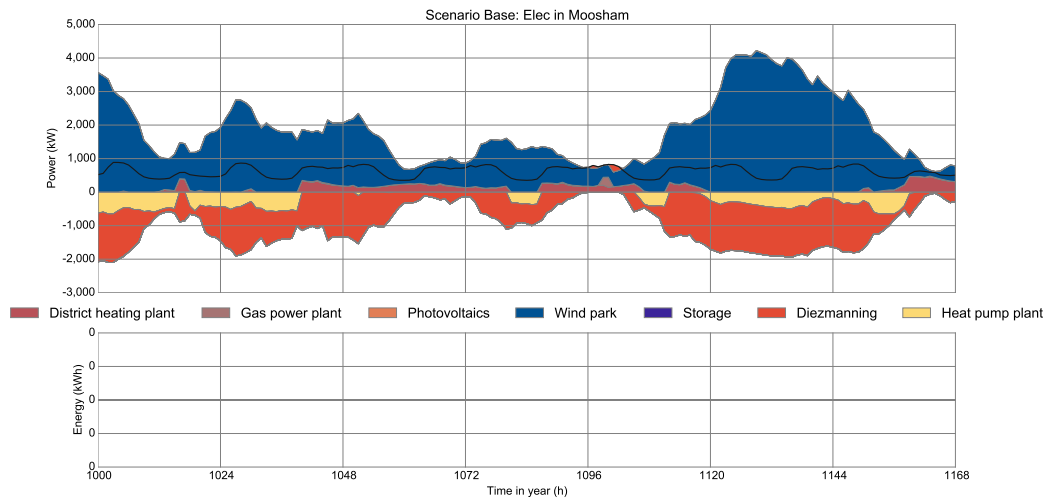
# Time series plots for Haag i. OB

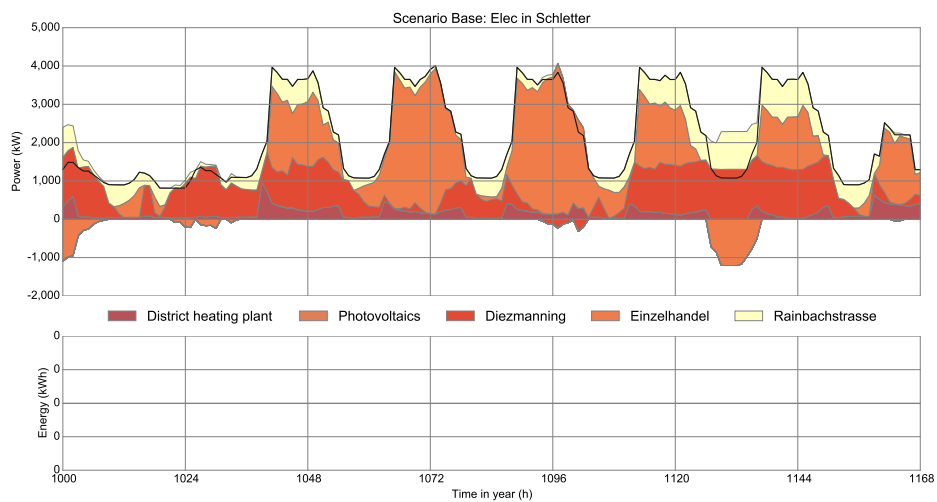
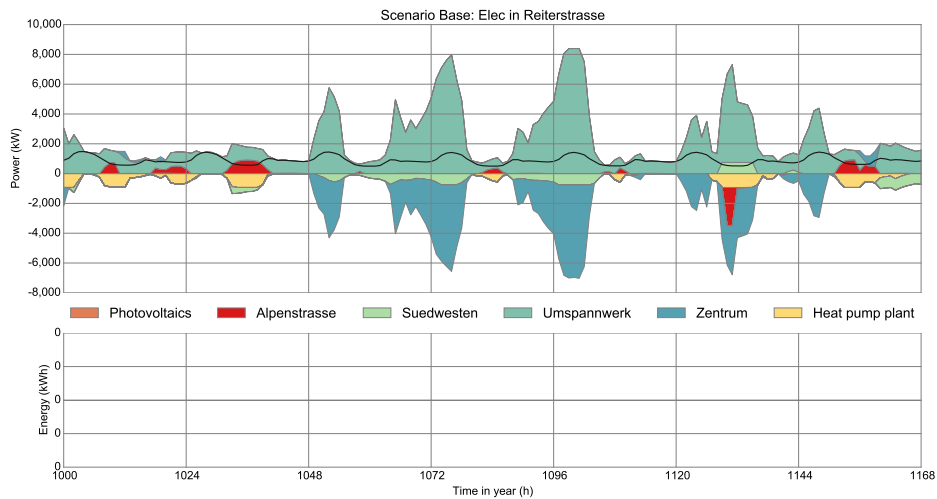
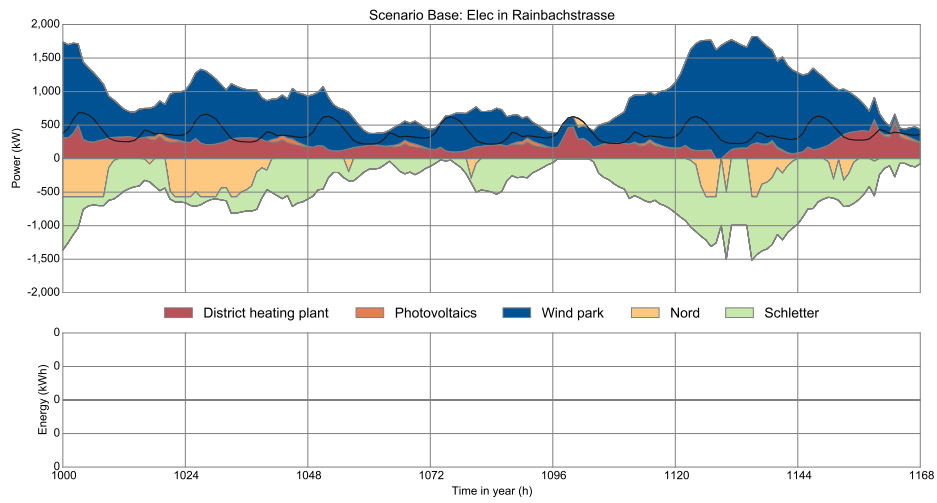
This appendix shows the result of model **urbs** for one (of 52) modelled weeks for two of the scenarios that were investigated in case study Haag i. OB. The complete case study is described in section 4.4. The shown week ranges from time step 1000 to 1168 and shows the energy balances for both demand commodities electricity and heat. Both balances are shown for all the twelve clusters (districts) within Haag.

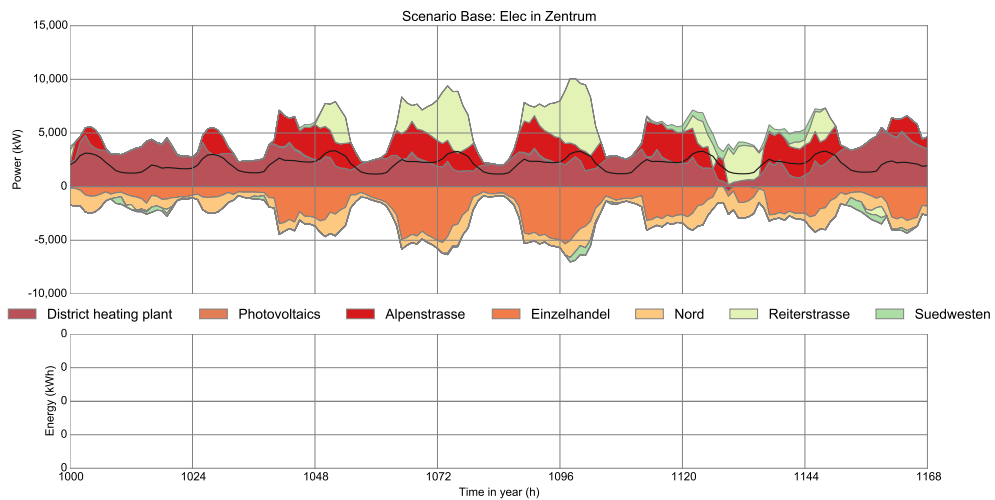
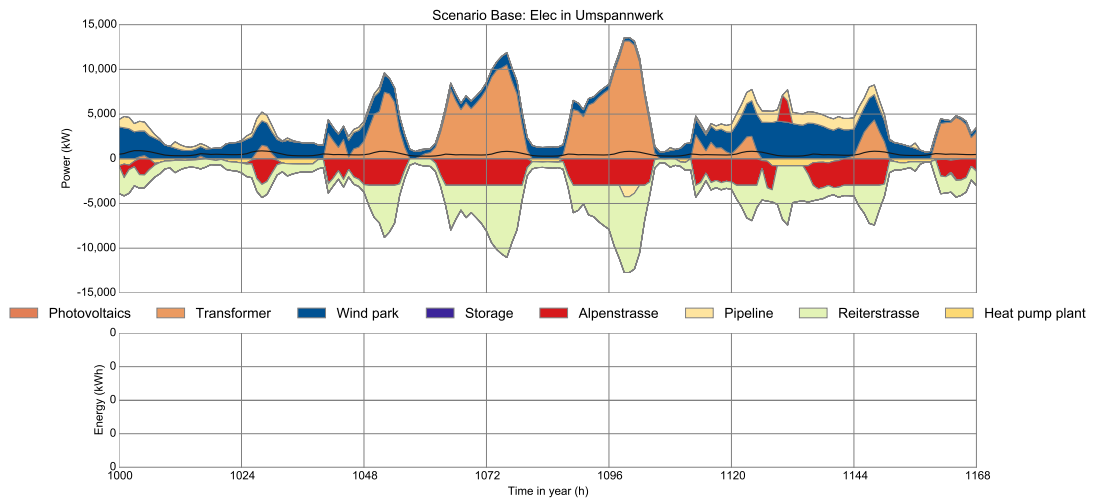
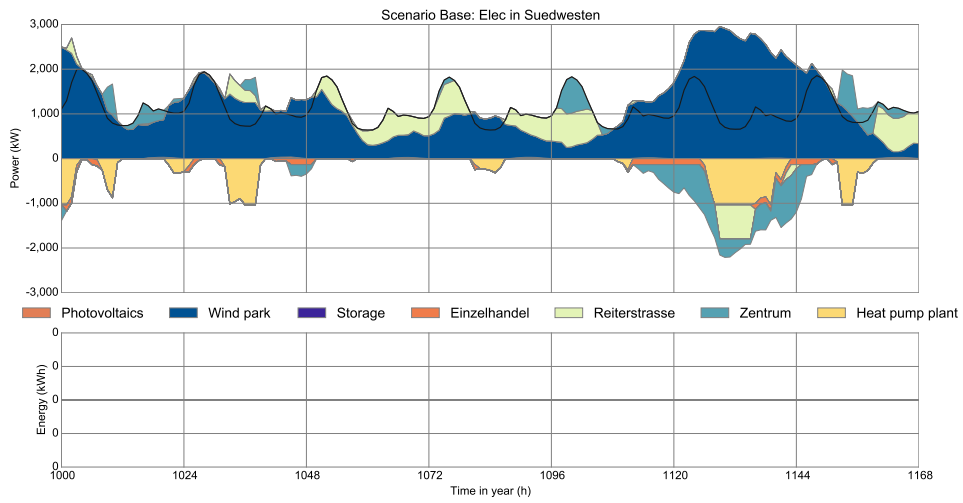
### C.1 Scenario BASE

#### C.1.1 Electricity



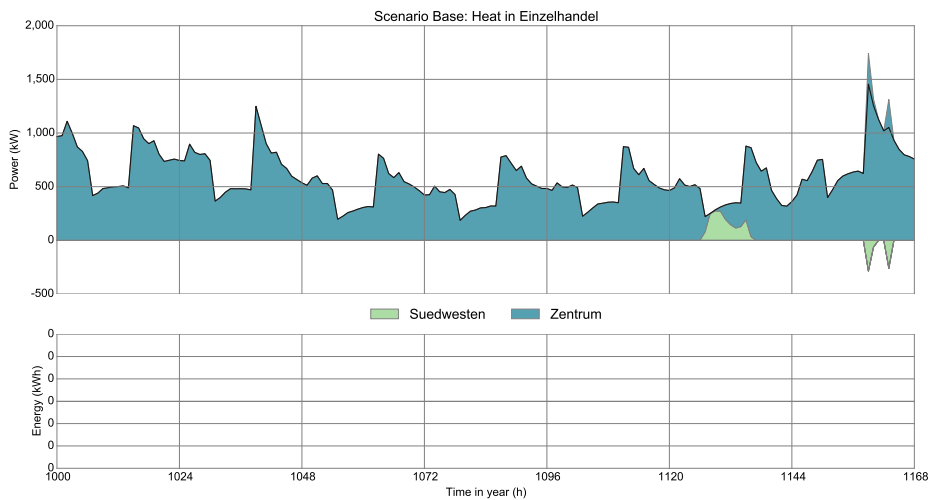
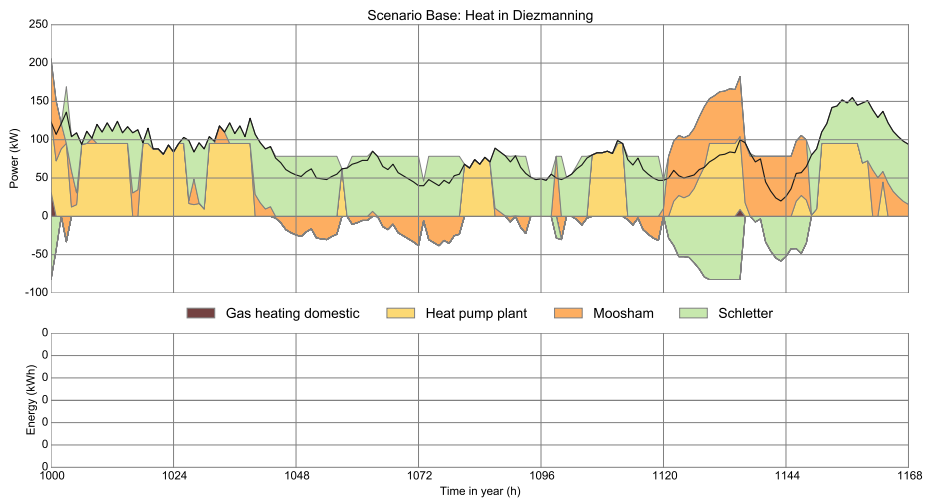
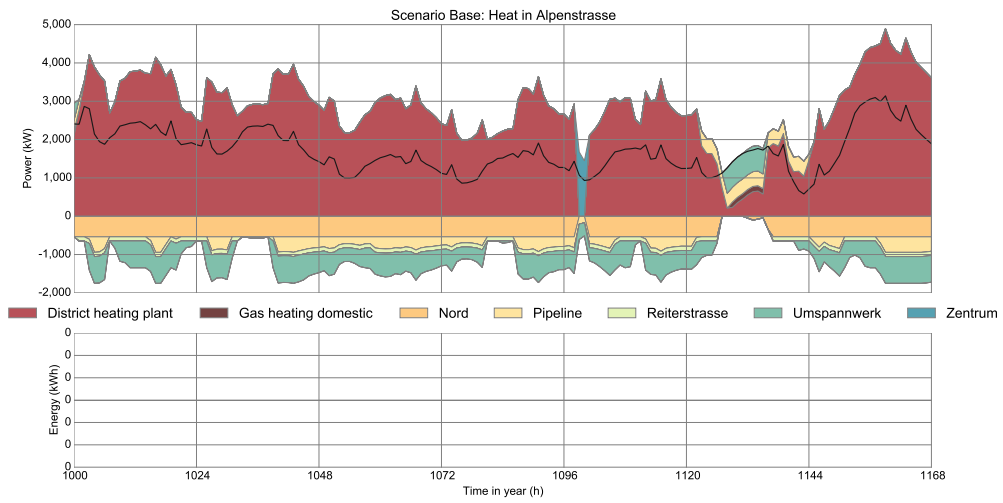


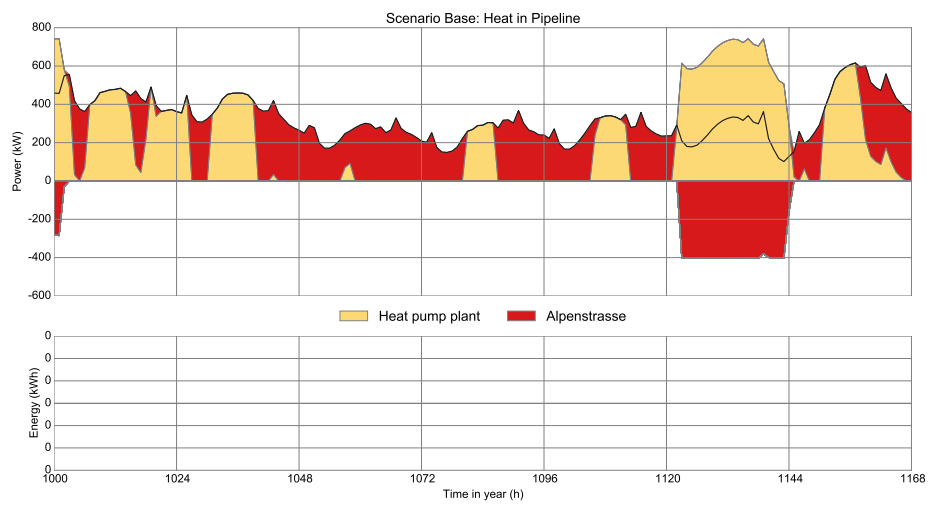
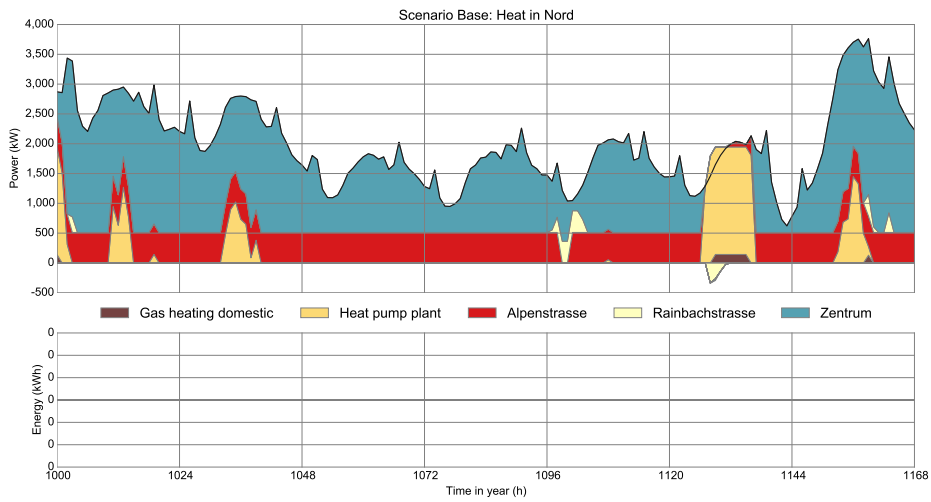
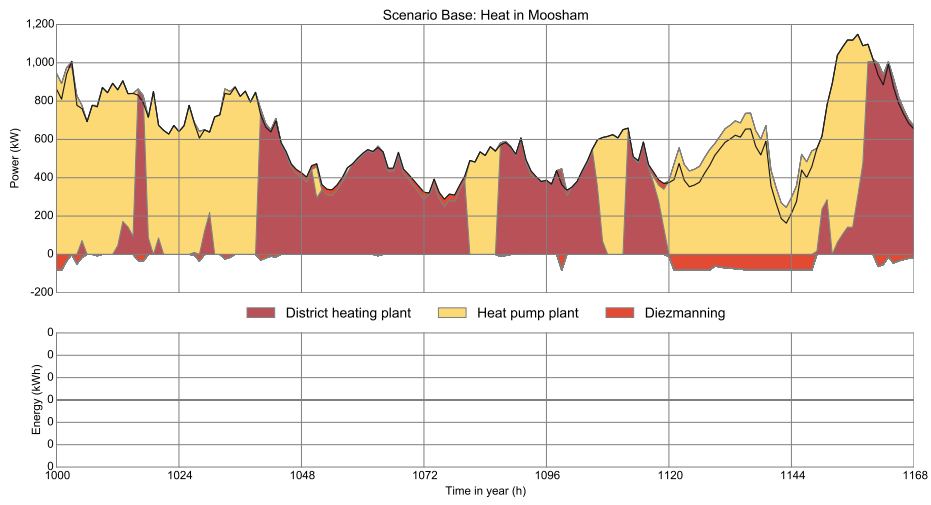


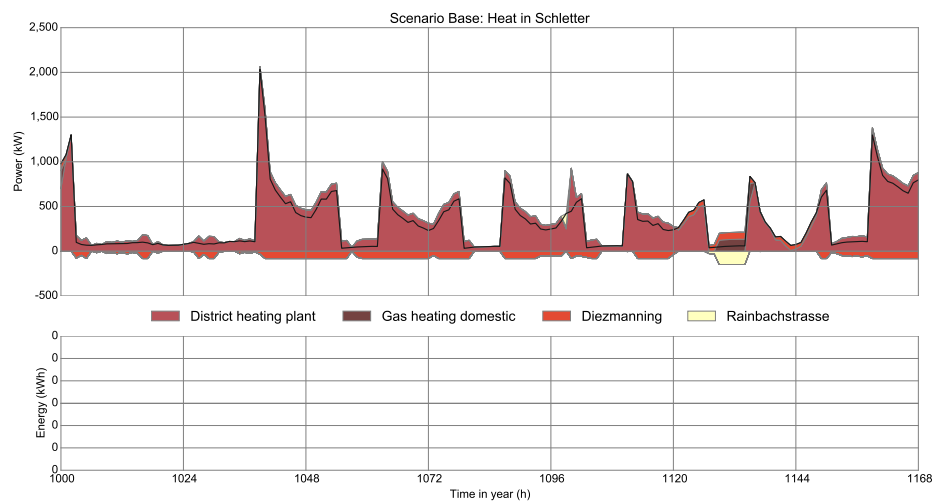
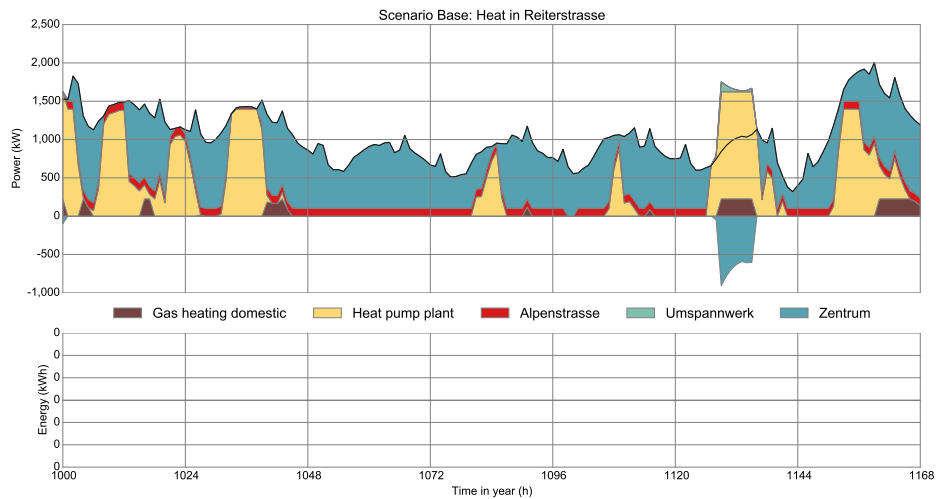
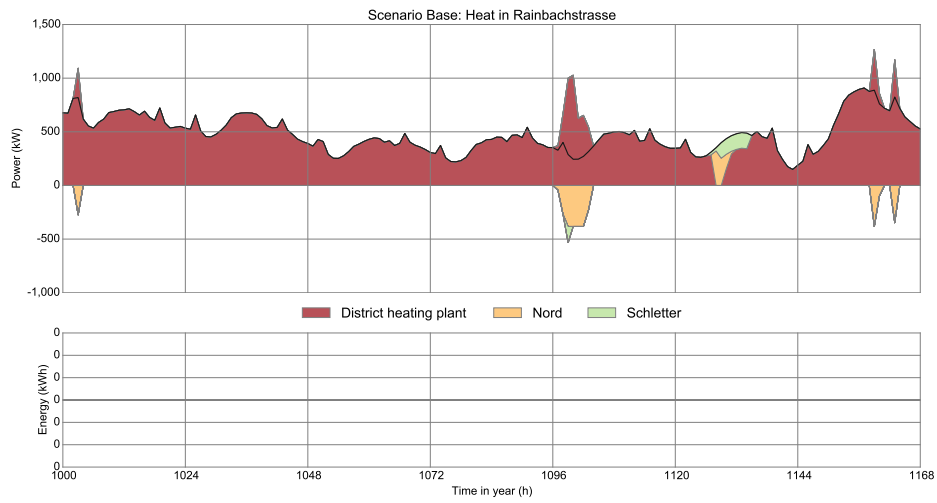


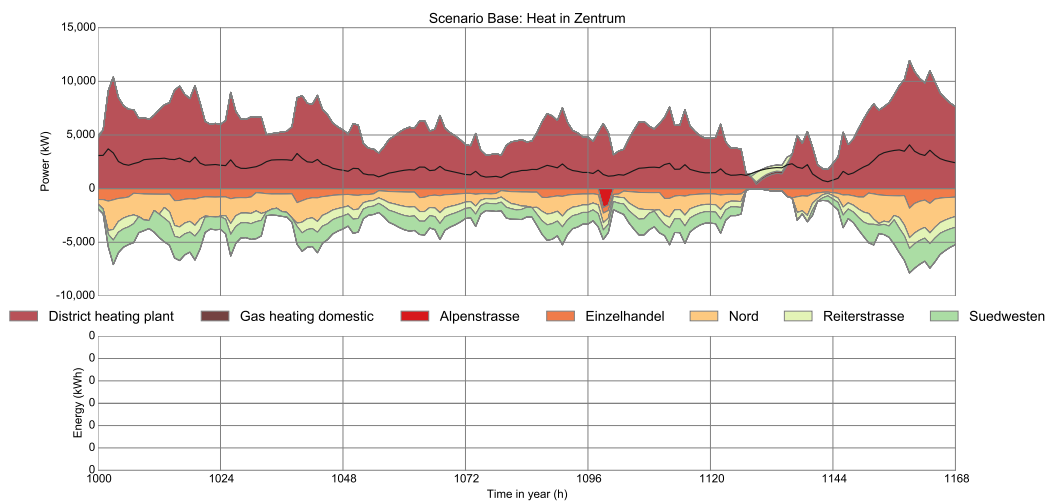
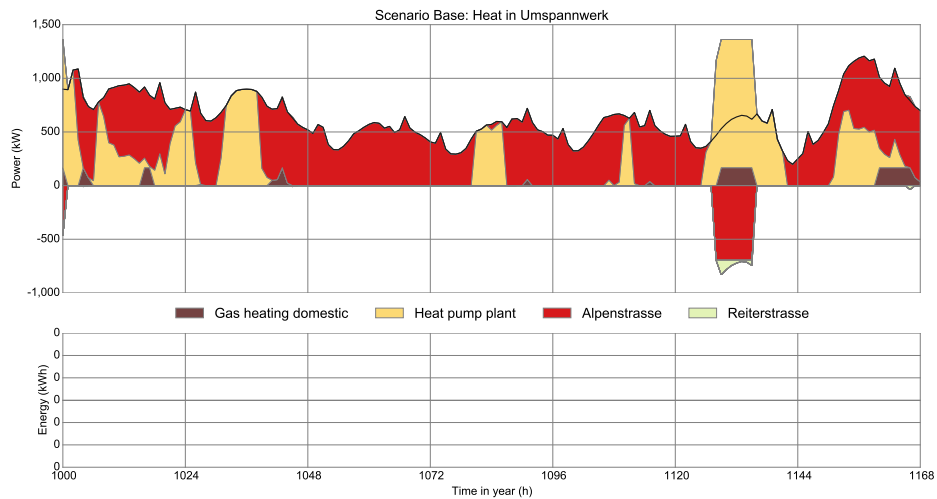
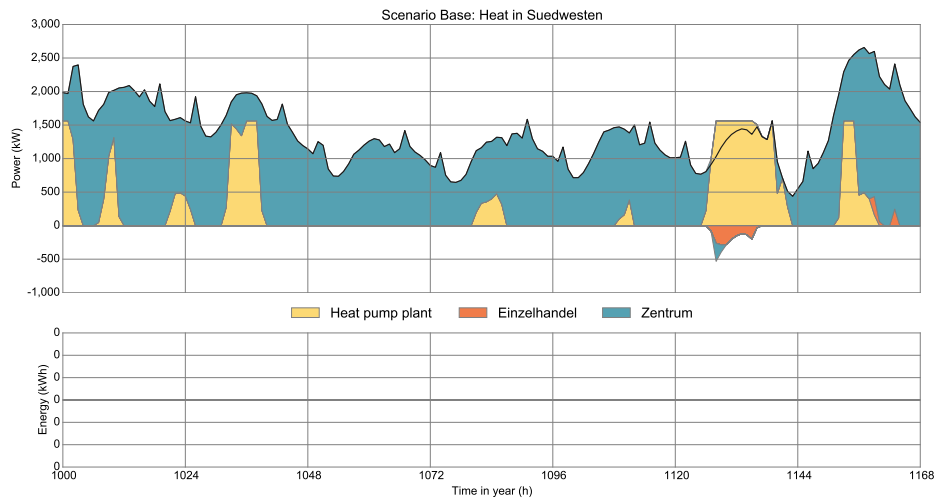
**C.1.2 Heat**





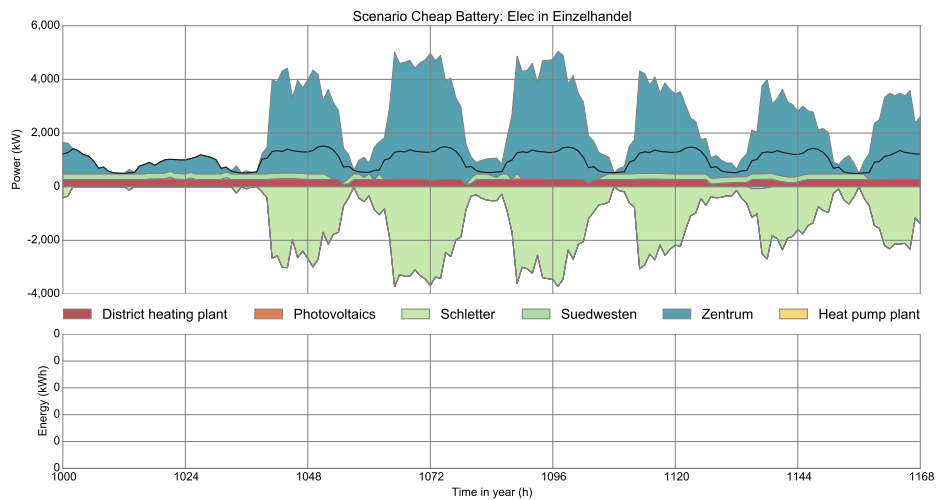
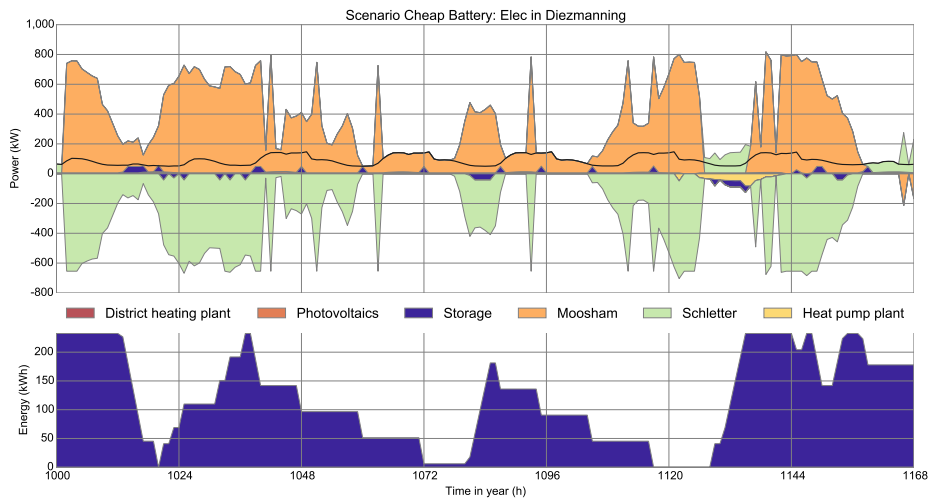
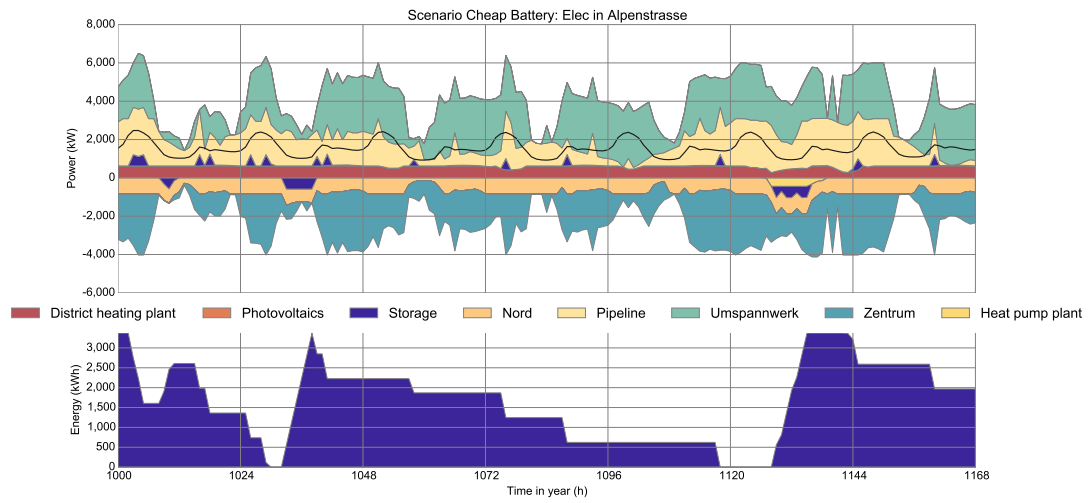


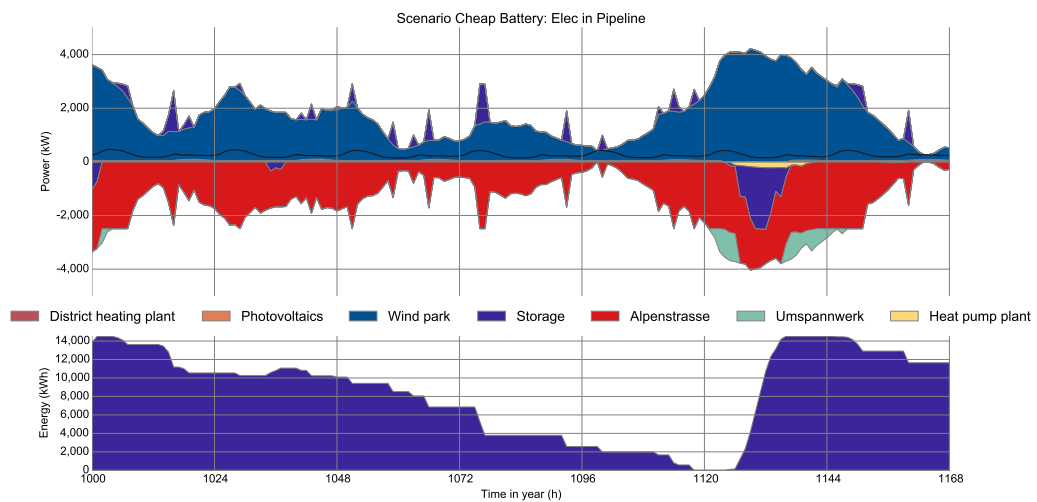
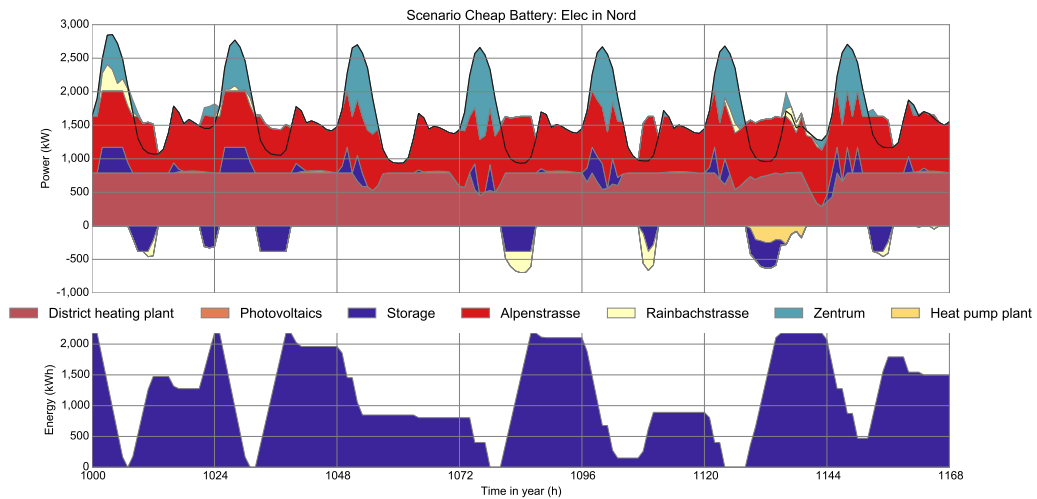
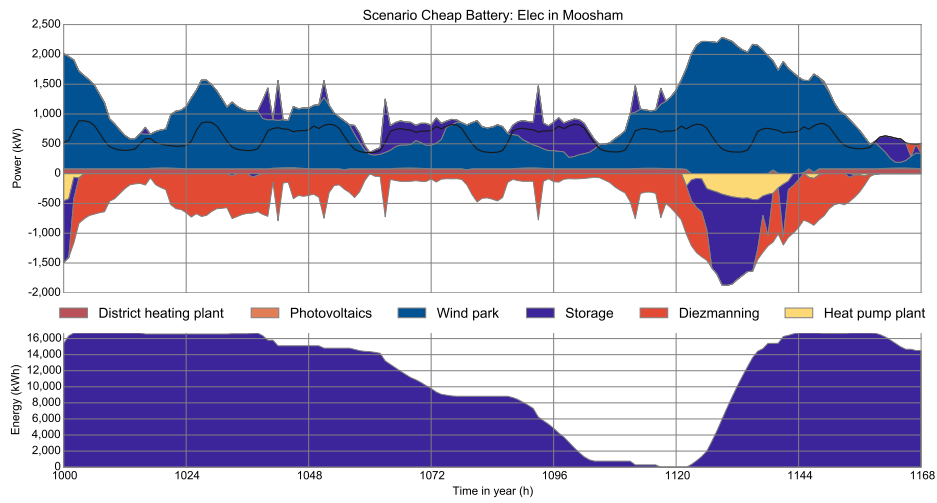


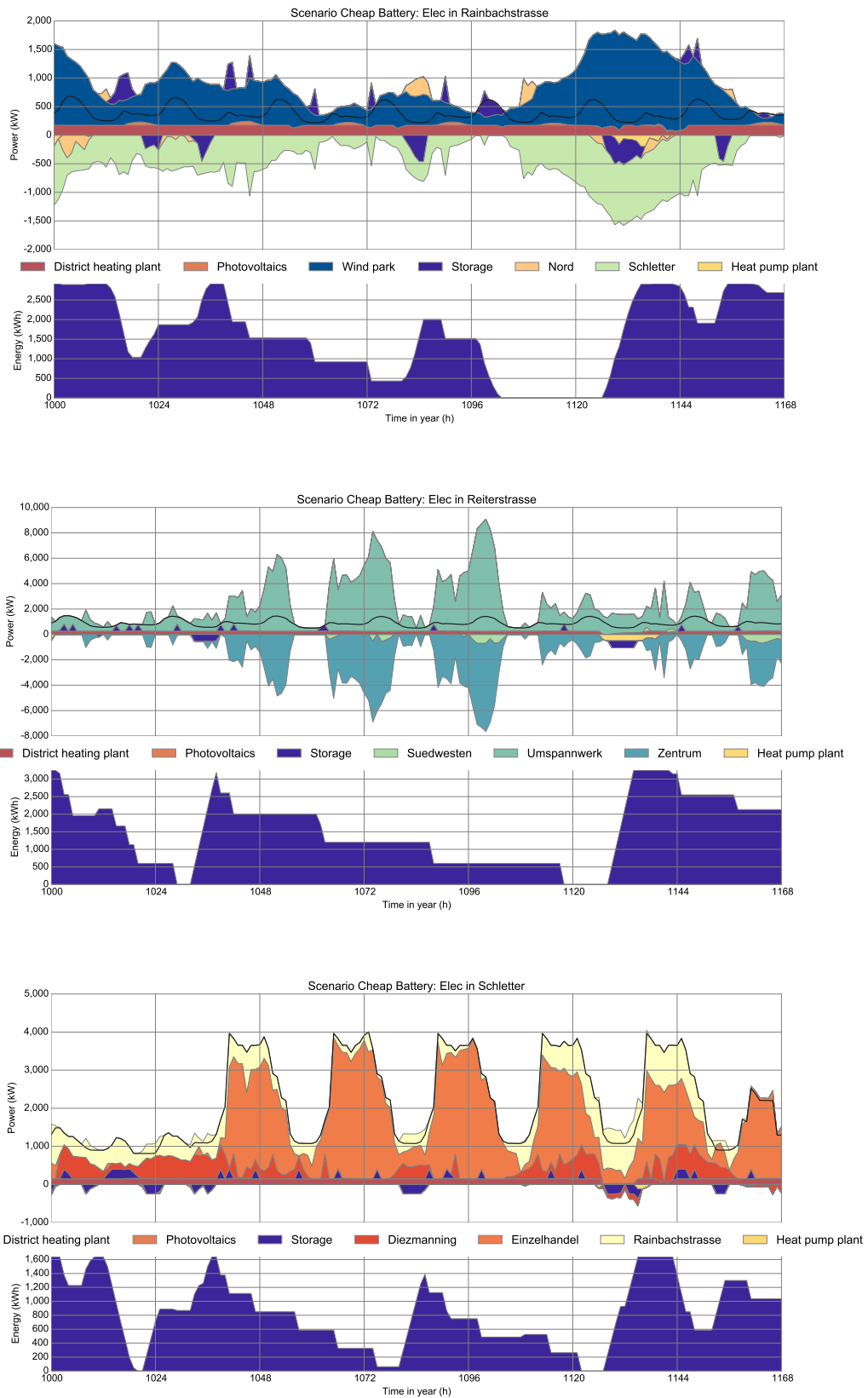


## **C.2 Scenario BATTERY CHEAP**

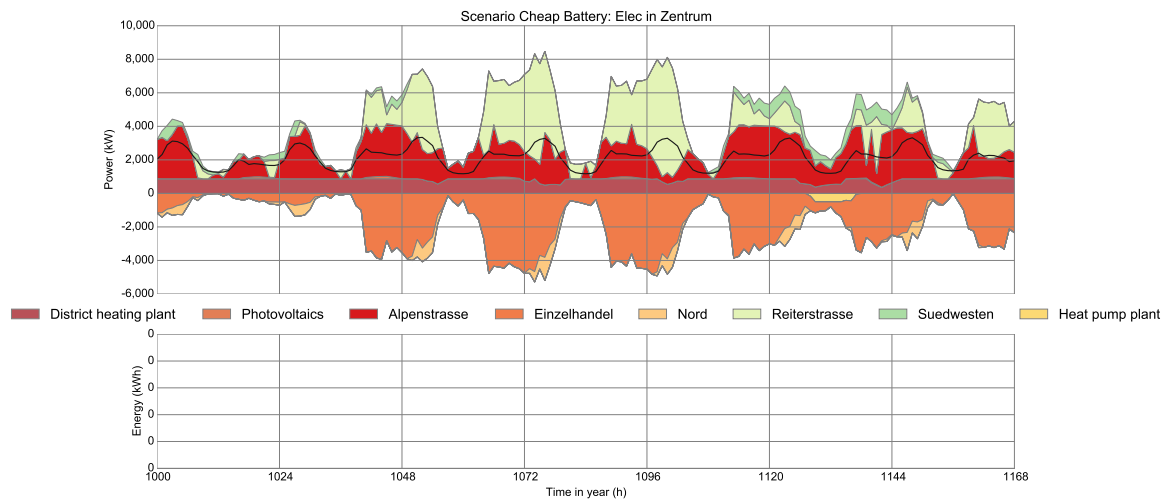
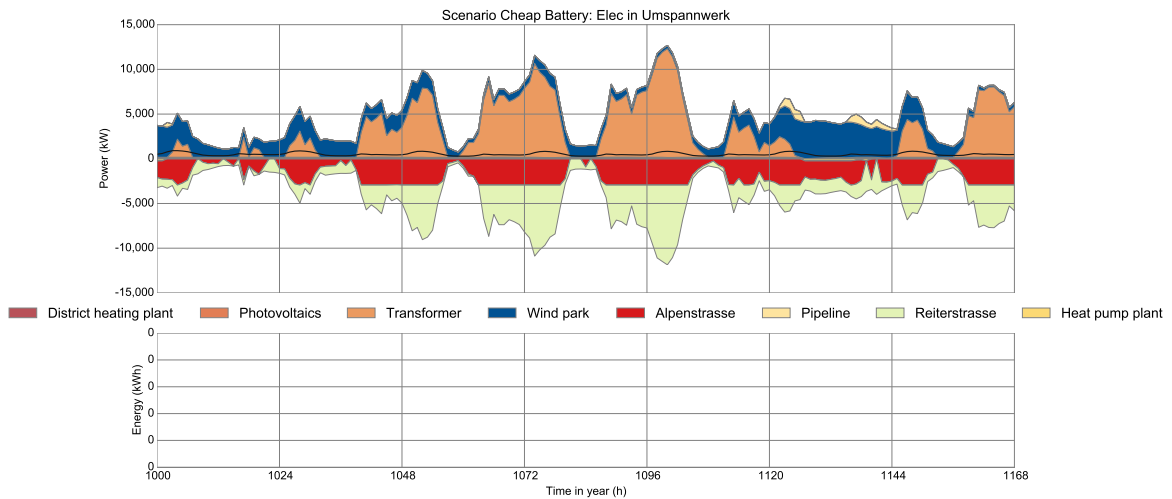
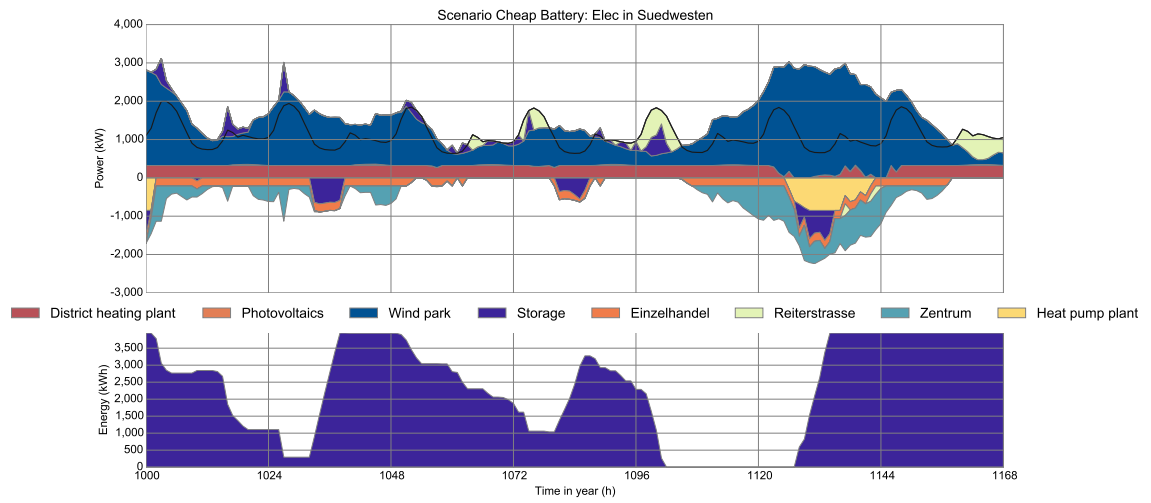
### **C.2.1 Electricity**



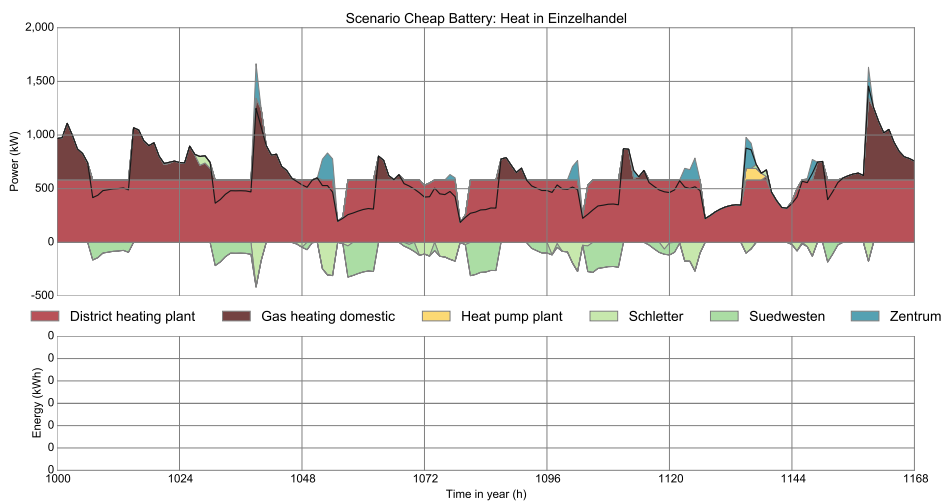
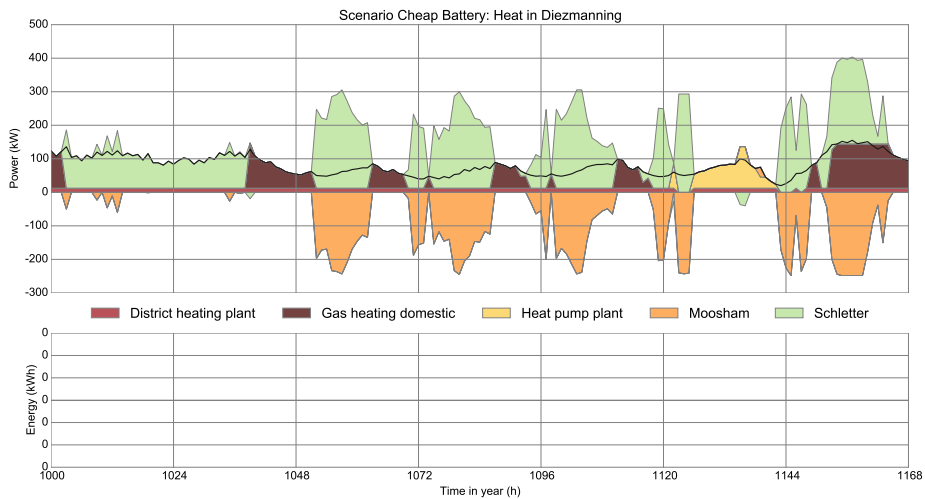
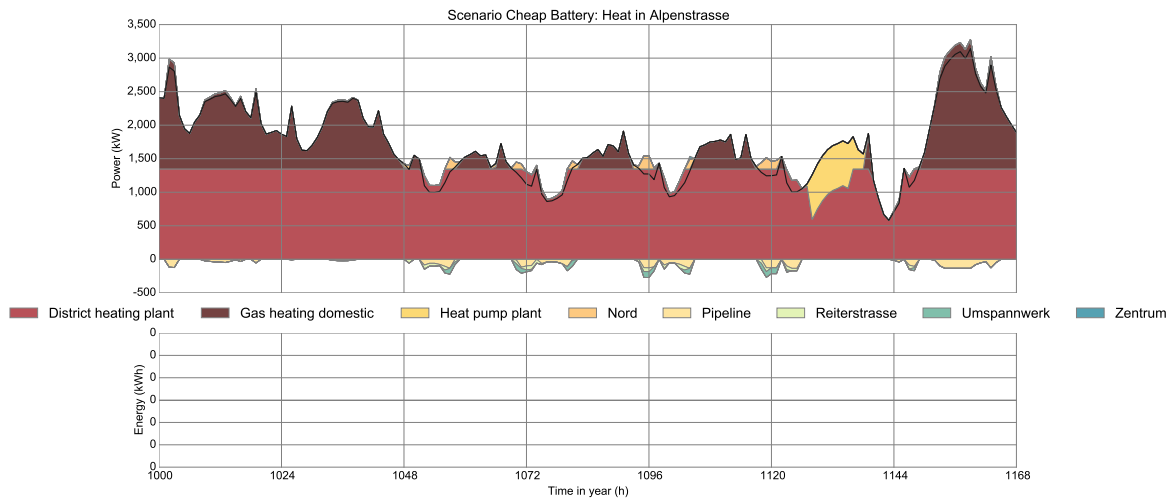


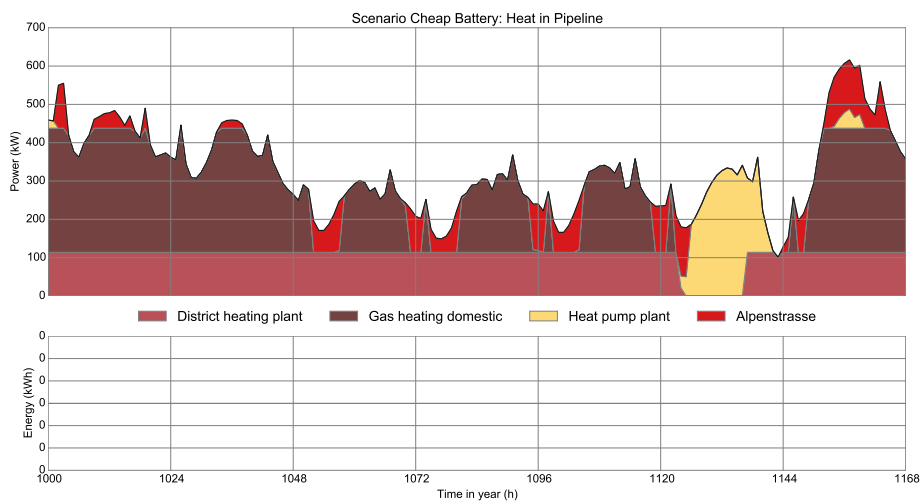
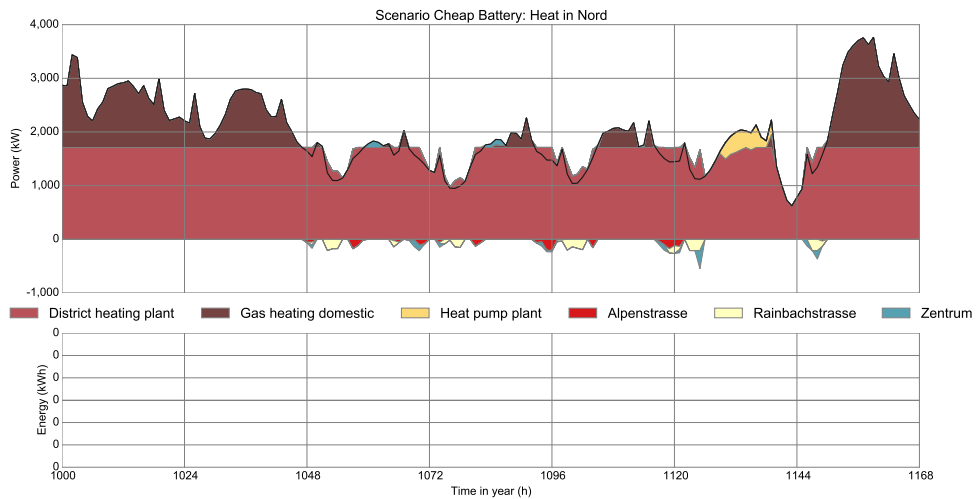
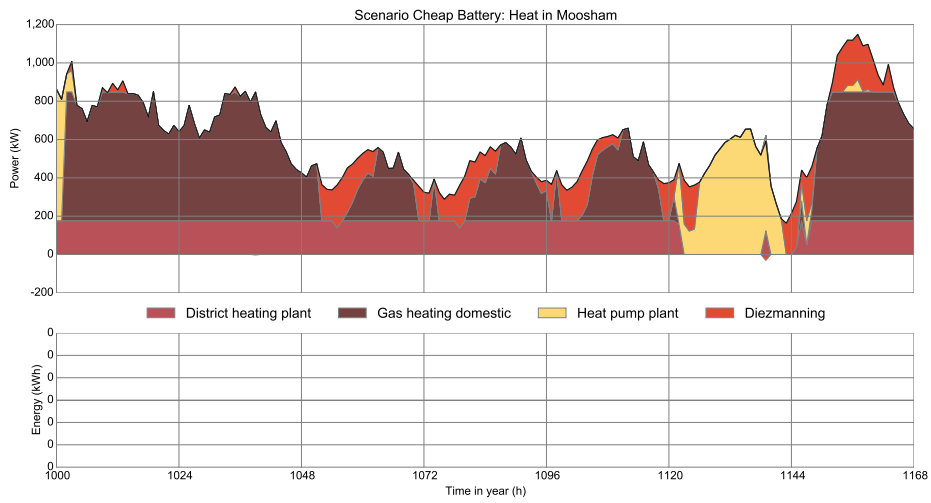


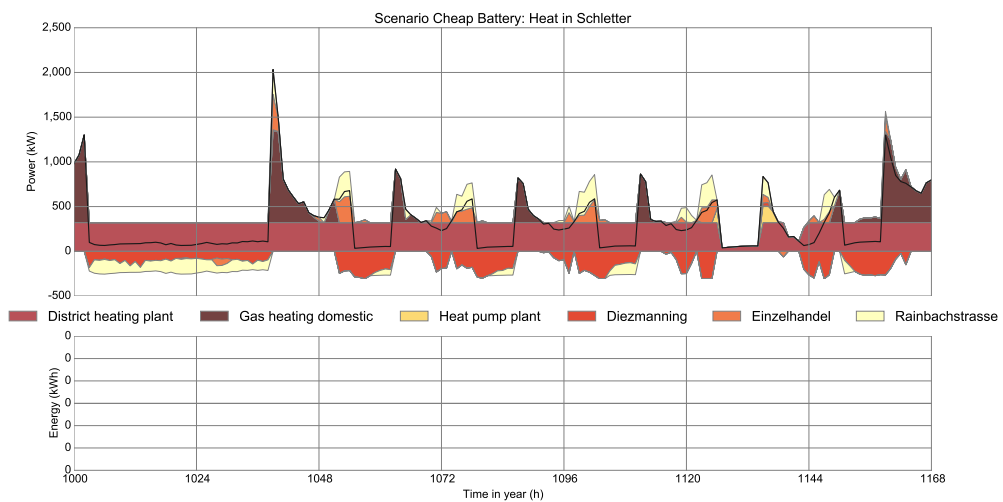
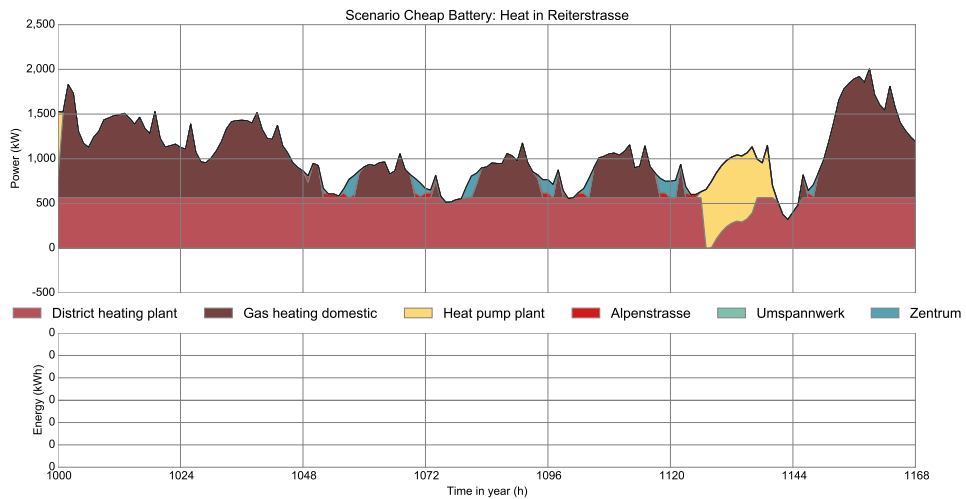
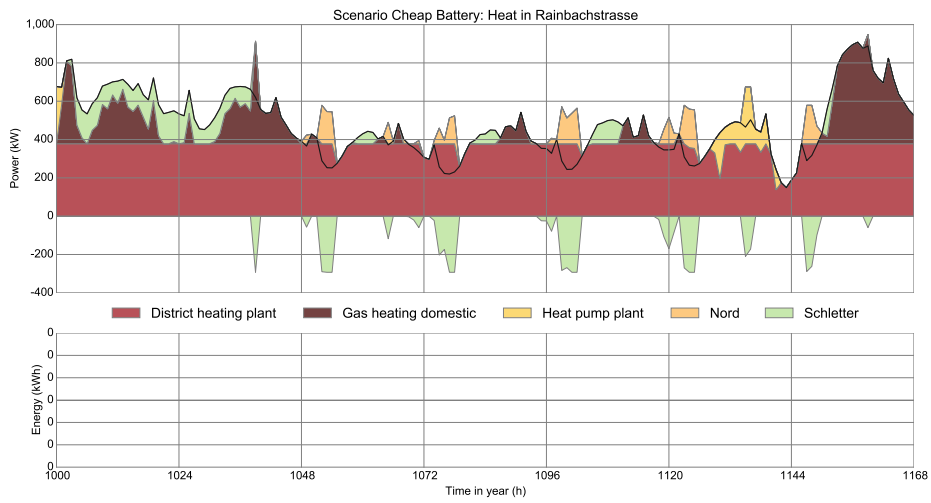


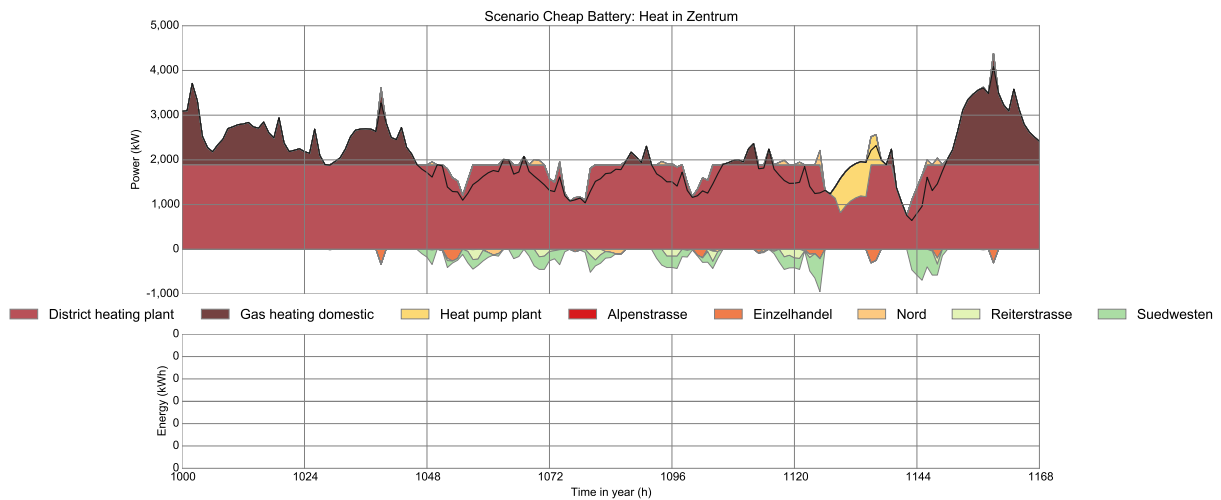
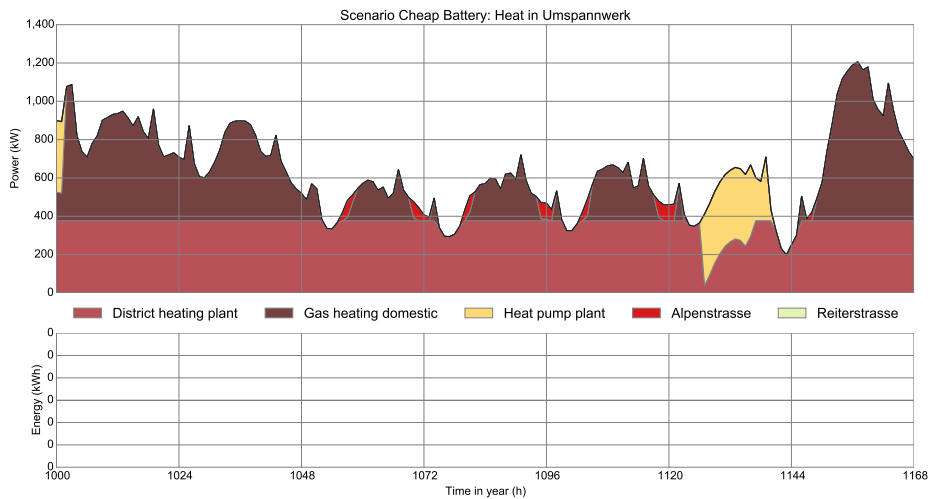
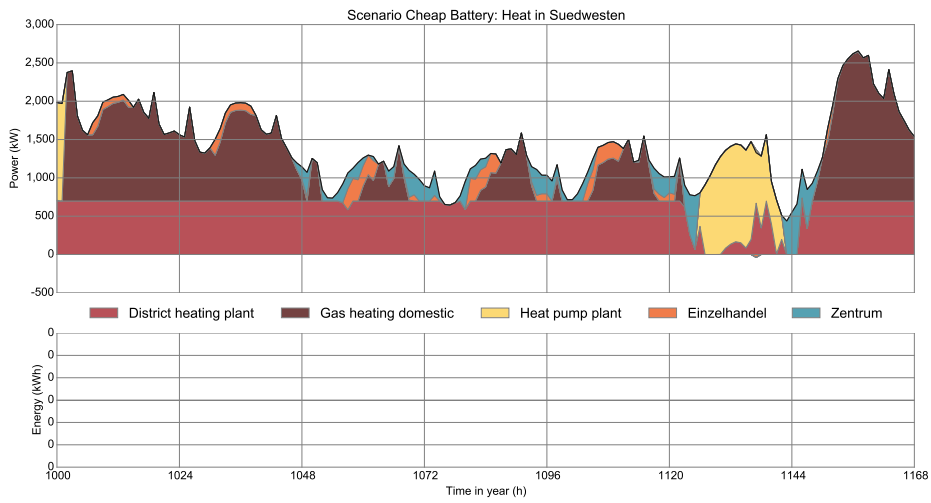


**C.2.2 Heat**









# Appendix D

## Symbols

Convention: greek letters ( $\kappa$ ,  $\pi$ ,  $\zeta$ ) represent variables. Roman uppercase letters ( $A$ ,  $C$ ) represent sets, their elements are denoted by the corresponding lowercase letters ( $a$ ,  $c$ ). Parameters are roman letters with subscripts ( $\underline{K}_{vp}$ ,  $g_{ij}$ ); the subscripts are always set element letters; lower and upper bound parameters are typeset with an underline ( $\underline{K}$ ) or and overline ( $\overline{G}$ ).

$a \in A$	<b>arc</b> (directed) from vertex $v_i$ to $v_j$ , set of arcs
$A_v^p$	arriving arcs into vertex $v$ ( <i>predecessor</i> )
$A_v^s$	leaving arcs from vertex $v$ ( <i>successor</i> )
$A$	building area (parameter)
$b \in B_e$	building, set of buildings along edge $e$
$b$	concurrence effect (parameter)
$c \in C$	<b>commodity</b> (mainly energy carrier), set of commodities
$C_{\text{dem}}$	demand commodity (required, like heat)
$C_{\text{env}}$	environmental commodity (created, like CO <sub>2</sub> )
$C_{\text{st}}$	stock commodity (storable, like coal)
$C_{\text{sup}}$	intermittent commodity (supply, like wind)
$C_{vp}^{\text{in}}$	input commodities of process $p$
$C_{vp}^{\text{out}}$	output commodities of process $p$
$c$	cost parameters (in models <b>DHMIN</b> and <b>dhmin</b> )
$d$	demand time series (parameter)
$\Delta t$	time step length (parameter)
$e \in E$	<b>edge</b> (undirected) between $v_i$ and $v_j$ , set of edges
$e$	efficiency (parameter)
$e^{\text{in}}$	input efficiency of storage (parameter)
$e^{\text{out}}$	output efficiency of storage (parameter)
$\epsilon$	energy/power flow (variable)
$\epsilon^{\text{in}}$	energy input for conversion and storage
$\epsilon^{\text{out}}$	energy output of conversion and storage
$f \in F$	<b>transmission</b> process, set of transmission processes

$F_{vc}^{\text{exp}}$	set of exporting transmission processes
$F_{vc}^{\text{imp}}$	set of importing transmission processes
$g_{ij}$	pipe existence (1=yes, 0=no) (parameter)
$\bar{G}$	greenhouse gas emissions, upper limit (parameter)
$\gamma$	greenhouse gas emissions (variable)
$h \in H$	<b>hub</b> process, set of hub processes
$i, j$	running indices, mostly for vertices
$l$	initial and final storage content (parameter)
$K$	capacity, already installed (parameter)
$\underline{K}$	capacity, lower bound (parameter)
$\bar{K}$	capacity, upper bound (parameter)
$k$	costs (parameter)
$k^{\text{fix}}$	fixed costs (parameter)
$k^{\text{fuel}}$	fuel costs (parameter)
$k^{\text{inv}}$	investment costs (parameter)
$k^{\text{inv,fix}}$	capacity-independent investment costs (parameter)
$k^{\text{inv,var}}$	capacity-dependent investment costs (parameter)
$k^{\text{var}}$	variable costs (parameter)
$k^{\text{c},\circ}$	capacity-dependent costs (parameter)
$k^{\text{p},\circ}$	power-dependent costs (parameter)
$\kappa$	capacity, total (variable)
$\hat{\kappa}$	capacity, new (variable)
$l$	length (parameter)
$m, n$	running indices
$n_b$	number of building floors (parameter)
$p \in P$	<b>process</b> (energy conversion), set of processes
$P_c^{\text{in}}$	set of processes consuming commodity $c$
$P_c^{\text{out}}$	set of processes producing commodity $c$
$\pi$	power flow, transmission (variable)
$\psi$	pipe usage decision (variable)
$q$	connection quota (parameter)
$Q^{\text{max}}$	source vertex capacity (parameter)
$r^{\text{heat}}$	revenue for heat (parameter)
$r^{\text{in}}$	input ratio (parameter)
$r^{\text{out}}$	output ratio (parameter)
$\mathbb{R}$	real numbers (set)
$\mathbb{R}_0^+$	non-negative real numbers (set)
$\rho$	source vertex flow (variable)
$s \in S$	<b>storage</b> (technology), set of storage processes



$s_{vct}$	intermittent supply capacity factor (parameter)
$s_{ct}$	peak demand scaling factor (parameter)
$t \in T$	time step, set of timesteps $\{t_0, t_1 \dots t_N\}$
$T_m$	set of modelled time steps $T \setminus \{t_0\}$
$t_0$	initialisation time step
$t_N$	final time step
$T_{flh}$	full load hours (parameter)
$\tau$	throughput, through MIMO process (variable)
$u$	annuity factor (parameter)
$v \in V$	<b>vertex</b> , set of vertices
$w^{\text{fix}}$	fixed thermal losses (parameter)
$w^{\text{var}}$	variable thermal losses (parameter)
$w$	time step weight (parameter)
$y$	availability (parameter)
$\xi$	pipe building decision (variable)
$z_b$	building area correction factor (parameter)
$\mathbb{Z}^1$	binary digit $\{0, 1\}$
$\zeta$	total costs, objective function value (variable)
$\zeta_{\text{fix}}$	total annual fixed (operation & maintenance) costs (variable)
$\zeta_{\text{fuel}}$	total annual fuel costs (variable)
$\zeta_{\text{inv}}$	total annual investment costs (variable)
$\zeta_{\text{var}}$	total annual variable costs (variable)



# Appendix E

## Acronyms

**AMPL** A Mathematical Programming Language 41, 52

**API** Application programming interface 48, 55

**BCE** Before the Common Era 11

**CAD** Computer aided design 39

**CAPEX** Capacity expenditure 60, 129

**CBC** COIN-OR branch and Cut 56

**CHP** Combined heat and power 13, 33, 65, 109, 122

**COIN-OR** Computational Infrastructure for Operations Research 51, 56

**Coopr** COmmon Optimization Python Repository 42

**COP** Coefficient of performance 141

**CPLEX** IBM ILOG CPLEX Optimization Studio 57

**CRS** Coordinate reference system 56, 62

**CSV** Comma separated values 55, 56

**DBMS** Database management system 56

**DC** District cooling 24

**DH** District heating 24, 125

**dhmin** District heating minimisation (seasonal) (optimisation model) 31, 39, 40, 60, 98, 106, 129, 132

**DHMIN** District heating minimisation (optimisation model) 15, 60, 101, 119, 121, 122, 129

**DOE** Department of Energy 144, 145

**EAC** Equivalent annual cost 35, 60

**EPGS** European Petroleum Survey Group 62

**ETSAP** Energy Technology Systems Analysis Programme 37

**EU** European Union 37

**GAMS** General Algebraic Modeling System 52, 55

**GDX** GAMS data exchange file 55

**GHG** Greenhouse gas 38, 141

**GIS** Geographic information system 39, 47

- GLPK** GNU Linear Programming Kit 41, 56
- GPS** Global positioning system 20
- GUI** Graphical user interface 40, 41
- HDF** Hierarchical Data Format 56, 156, 157
- IEA** International Energy Agency 37
- IIASA** International Institute for Applied Systems Analysis 38
- IP** Integer programming 27, 28
- LP** Linear programming 16, 27, 28, 29, 43, 56, 57
- MARKAL** MARKet ALocation 36, 37, 38, 42
- MERRA** Modern-era retrospective analysis for research and applications 61
- MES** Multi-energy system 13, 14, 16
- MESSAGE** Model for Energy Supply Strategy Alternatives and their General Environmental Impact 38
- MILP** Mixed integer linear programming 16, 27, 28, 29, 40, 55, 56, 57, 127, 153
- MIMO** Multiple input multiple output 33, 38, 77, 185
- MIP** Mixed integer programming 55
- NLP** Non-linear programming 36
- NPV** Net present value 36, 37, 60
- O&M** Operation & maintenance 59, 94, 95, 101, 103
- OPEX** Operational expenditure 60, 129
- OSM** OpenStreetMap 62, 122, 123, 138, 145
- Pandas** Python Data Analysis Library 41
- POI** Point of interest 129
- PRISM** Puget Sound Regional Integrated Synthesis Model 12
- PU** Polyurethane 23
- Pyomo** Python Optimization Modeling Objects 41, 42, 55
- RES** Reference energy system 14, 36, 38, 41, 42, 65, 66, 77
- rivus** Network optimisation model (optimisation model) 31, 34, 39, 46, 60, 121, 135, 138, 141, 145, 154
- SCIP** Solving Constraint Integer Programs 56
- SISO** Single input single output 32, 38, 65, 77
- SOC** State of charge 74
- SWM** Stadtwerke München 117, 122, 129
- TIMES** The Integrated MARKAL-EFOM System 37
- TMY** Typical meteorological year 144, 145
- UC** Unit commitment 22

**urbs** Distributed energy system (optimisation model) 34, 38, 60, 77, 82, 135, 141, 145, 148, 150, 154

**URBS** Urban Research Toolbox: Energy Systems 15, 38, 39, 60, 65, 82

**USAF** United States Air Force 144

**WACC** Weighted average cost of capital 60

**XML** Extensible markup language 62

**YAML** Yet Another Markup Language 41

**ZIB** Zuse Institute Berlin 56



## Appendix F

# Bibliography

- [1] T. Achterberg. *Constraint Integer Programming*. Doctoral thesis, Technische Universität Berlin, Jan 2009. URL: <http://nbn-resolving.de/urn:nbn:de:0297-zib-11129.44>
- [2] M. Agnew, L. Schrattenholzer, and A. Voss. *User's Guide for the MESSAGE Computer Program*. International Institute for Applied Systems Analysis (IIASA), 1978. IISASA Research Memorandum RM-78-026. URL: <http://webarchive.iiasa.ac.at/cgi-bin/pub/pubsrchKK?ID:RM78026>. 38
- [3] C. M. Agudelo-Vera, A. R. Mels, K. J. Keesman, and H. H. Rijnaarts. Resource management as a key factor for sustainable urban planning. *Journal of Environmental Management*, 92(10):2295–2303, Oct 2011. doi:10.1016/j.jenvman.2011.05.016. 13
- [4] M. Alberti and P. Waddell. An integrated urban development and ecological simulation model. *Integrated Assessment*, 1(3):215–227, 2000. doi:10.1023/a:1019140101212. 13
- [5] B. H. Bakken and H. I. Skjelbred. Planning of distributed energy supply to suburb. In *Power Engineering Society General Meeting, IEEE*, pages 1–8, Tampa, FL, Jun 2007. 12
- [6] C. Barber, D. Dobkin, and H. Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Trans. on Mathematical Software*, 22(4):469–483, Dec 1996. URL: <http://www.qhull.org>. 48, 195
- [7] M. Beller. Reference energy system methodology. *NASA STI/Recon Technical Report N*, 77:17574, Apr. 1976. 36
- [8] Bentley sisHYD. District Heating Network Modeling and Analysis, 2009. URL: <http://www.bentley.com/en-US/Products/Bentley+sisHYD>. 39
- [9] J. Bezanson, S. Karpinski, V. B. Shah, and A. Edelman. Julia: A fast dynamic language for technical computing. *arXiv preprint arXiv:1209.5145*, 2012. URL: <http://arxiv.org/abs/1209.5145>. 55

- [10] S. C. Bhattacharyya and G. R. Timilsina. A review of energy system models. *International Journal of Energy Sector Management*, 4(4):494–518, Nov 2010. doi: 10.1108/17506221011092742. 36
- [11] M. Blesl. *Räumlich hoch aufgelöste Modellierung leitungsgebundener Energieversorgungssysteme zur Deckung des Niedertemperaturwärmebedarfs*. Doctoral thesis, Institut für Energiewirtschaft und Rationelle Energieanwendung, Universität Stuttgart, Jul 2002. URL: <http://elib.uni-stuttgart.de/opus/volltexte/2002/1193/>. 40
- [12] F. Botzenhart and T. Hamacher. Optimierung urbaner Energiesysteme – Analysen zur Struktur der Wärmeversorgung in Salzburg. In *17. Münchner Fortbildungsseminar Geoinformationssysteme*. Max-Planck-Institut für Plasmaphysik, Mar 2012. URL: [http://gispoint.de/fileadmin/user\\_upload/paper\\_gis\\_open/537514005.pdf](http://gispoint.de/fileadmin/user_upload/paper_gis_open/537514005.pdf). 15
- [13] M. T. Brown. Systems diagram of a city. Image, Jun 2010. License: CC BY-SA 3.0. URL: [https://commons.wikimedia.org/wiki/File:Systems\\_diagram\\_of\\_a\\_city.png](https://commons.wikimedia.org/wiki/File:Systems_diagram_of_a_city.png). 13
- [14] T. Bruckner. High-resolution modeling of energy-services supply systems using deeco: Overview and application to policy development. *Annals of Operations Research*, 121(1/4):151–180, 2003. doi:10.1023/a:1023359303704. 38
- [15] T. Bruckner, H.-M. Groscurth, and R. Kümmel. Competition and synergy between energy technologies in municipal energy systems. *Energy*, 22(10):1005–1014, Oct 1997. doi:10.1016/s0360-5442(97)00037-6. 38
- [16] W. Buhr. What is infrastructure? Technical report, Department of Economics, School of Economic Disciplines, University of Siegen, 2003. URL: <http://www.wiwi.uni-siegen.de/vwl/research/diskussionsbeitraege/pdf/107-03.pdf>. 20
- [17] P. Böhme. *Wärmeverbrauchsanalyse auf Basis einer raumbezogenen Zusammenführung von Gebäudedaten*. Doctoral thesis, Technische Universität München, Munich, Dec 2013. URL: <http://nbn-resolving.de/urn/resolver.pl?urn:nbn:de:bvb:91-diss-20131219-1162472-0-3>. 62, 138
- [18] D. Ciechanowicz, H. Ayd, M. Lees, A. Knoll, and T. Hamacher. A universal scheme for modeling energy systems. Technical Report TUM-I1341, Technische Universität München, 2013. URL: <https://mediatum.ub.tum.de/?id=1172962>. 36
- [19] C. Coffrin and P. Van Hentenryck. A linear-programming approximation of AC power flows. *INFORMS Journal on Computing*, 26(4):718–734, 2014. URL: <http://arxiv.org/pdf/1206.3614.pdf>. 23, 154
- [20] G. B. Dantzig. *Linear programming and extensions*. Princeton University Press, 1998. ISBN: 9780691059136. URL links to the original (1963) version of the identically named report that would become this book. URL: <https://www.rand.org/content/dam/rand/pubs/reports/2007/R366part1.pdf>. 43



- [21] J. DeCarolis, K. Hunter, and S. Sreepathi. The TEMOA project: Tools for energy model optimization and analysis. In *International Energy Workshop*, Stockholm, Sweden, Jun 21-23 2010. URL: [http://www.temoaproject.org/publications/DeCarolis\\_IEW2010\\_paper.pdf](http://www.temoaproject.org/publications/DeCarolis_IEW2010_paper.pdf). 42
- [22] J. Dorfner. `pandashp.py`. GitHub, May 2014. Finds closest edge for centroid of polygons. URL: <https://github.com/ojdo/python-tools/blob/master/pandashp.py>. 47
- [23] J. Dorfner. `skeletrontools.py`. GitHub, Feb 2014. Uses `qhull` [6] to find simplified road network for given DataFrame of roads. URL: <https://github.com/ojdo/python-tools/blob/master/skeletrontools.py>. 49
- [24] J. Dorfner and T. Hamacher. Large-scale district heating network optimization. *IEEE Transactions on Smart Grid*, 5(4):1884–1891, Jul 2014. doi:10.1109/TSG.2013.2295856. 91, 122
- [25] I. Dunning, J. Huchette, and M. Lubin. JuMP: A modeling language for mathematical optimization. *arXiv preprint arXiv:1508.01982*, 2015. URL: <http://arxiv.org/abs/1508.01982>. 55
- [26] ESRI. ESRI Shapefile Technical Description. White Paper, Jul 1998. URL: <https://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>. 62
- [27] J. W. Forrester. Systems analysis as a tool for urban planning. *IEEE Transactions On Systems Science and Cybernetics*, 6(4):258–265, Oct 1970. doi:10.1109/TSSC.1970.300299. 13
- [28] M. Frondel, P. Grösche, N. Ritter, H. Tauchmann, C. Vance, P. Matuschek, and U. Müller. Erhebung des Energieverbrauchs der privaten Haushalte für die Jahre 2006–2008. Technical report, RWI, forsa, Oct 2011. URL: <http://www.bmwi.de/DE/Mediathek/Publikationen/publikationen-archiv,did=452032.html>. 123
- [29] H. Gadd and S. Werner. Heat load patterns in district heating substations. *Applied Energy*, 108:176–183, Aug 2013. doi:10.1016/j.apenergy.2013.02.062. 61
- [30] M. Geidl. *Integrated Modeling and Optimization of Multi-Carrier Energy Systems*. Dissertation, ETH Zürich, 2007. URL: [http://www.eeh.ee.ethz.ch/uploads/tx\\_ethpublications/eth-diss-17141.pdf](http://www.eeh.ee.ethz.ch/uploads/tx_ethpublications/eth-diss-17141.pdf). 38
- [31] M. Geidl, G. Koeppel, P. Favre-Perrod, B. Klockl, G. Andersson, and K. Frohlich. Energy hubs for the future. *IEEE Power and Energy Mag.*, 5(1):24–30, Jan 2007. doi:10.1109/mpae.2007.264850. 38
- [32] C. Geiß, H. Taubenböck, M. Wurm, T. Esch, M. Nast, C. Schillings, and T. Blaschke. Remote sensing-based characterization of settlement structures for assessing local potential of district heat. *Remote Sensing*, 3:1447–1471, 2011. doi:10.3390/rs3071447. 123
- [33] S. Gillies and Toblerity. Shapely, 2010. Python package for manipulation and analysis of features in the Cartesian plane. URL: <https://github.com/Toblerity/Shapely>. 58

- [34] L. Girardin, F. Marechal, M. Dubuis, N. Calame-Darbellay, and D. Favrat. Energis: A geographical information based system for the evaluation of integrated energy conversion systems in urban areas. *Energy*, 35(2):830–840, Feb 2010. doi:10.1016/j.energy.2009.08.018. 14
- [35] T. H. Gobmaier. *Entwicklung und Anwendung einer Methodik zur Synthese zukünftiger Verbraucherlastgänge*. Doctoral thesis, Technische Universität München, Dec 2013. URL: <https://mediatum.ub.tum.de/?id=1166299>. 61
- [36] Google Books Team. Google Ngram Viewer, 2010. URL: <http://books.google.com/ngrams>. 20
- [37] Google Maps, Accessed Feb 2013. URL: <http://maps.google.com/>. 62, 123, 124
- [38] P. E. Grohnheit and H. V. Larsen. Balmorel – data and calibration, Mar 2001. Version 2.05. URL: <http://www.eabalmorel.dk/files/download/Balmorel%20Data%20and%20Calibration%20Version%202.05.pdf>. 41
- [39] S. Gronemann and R. Döring. Nachhaltigkeit und Diskontierung. *Zeitschrift für Wirtschafts- und Unternehmensethik*, 2:233–256, 2001. URL: [http://www.zfwu.de/fileadmin/pdf/2\\_2001/Gronemann\\_Doering.pdf](http://www.zfwu.de/fileadmin/pdf/2_2001/Gronemann_Doering.pdf). 60
- [40] H.-M. Groscurth, T. Bruckner, and R. Kümmel. Modeling of energy-services supply systems. *Energy*, 20(9):941–958, Sep 1995. doi:10.1016/0360-5442(95)00067-q. 39
- [41] T. Harmat. *Verallgemeinerung der Diskretisierung von Jahresdauerlinien für mehrere Energieträger*. Bachelor thesis, Technische Universität München, Feb 2015. URL: <https://mediatum.ub.tum.de/?id=1278678>. 32, 45, 46
- [42] C. Heilek. *Modellgestützte Optimierung des Neubaus und Einsatzes von Erzeugungsanlagen und Speichern für elektrische und thermische Energie im deutschen Energiesystem*. Doctoral thesis, Technische Universität München, Mar 2015. URL: <https://mediatum.ub.tum.de/?id=1230817>. 40
- [43] N. Heitmann. *Lösung energiewirtschaftlicher Probleme mit Hilfe linearer Programmierung*. Diploma thesis, Universität Augsburg, Oct 2005. URL: <http://hdl.handle.net/11858/00-001M-0000-0027-1AC2-0>. 15
- [44] M. Hellwig. *Entwicklung und Anwendung parametrisierter Standard-Lastprofile*. Doctoral thesis, Technische Universität München, Sep 2003. URL: <https://mediatum.ub.tum.de/?id=601557>. 61
- [45] R. Holden, D. V. Val, R. Burkhard, and S. Nodwell. A network flow model for interdependent infrastructures at the local scale. *Safety Science*, 53:51–60, 2012. 39
- [46] Homer Energy LLC. Homer (hybrid optimization of multiple energy resources). Version 3.3.1, Aug 2015. URL: <http://www.homerenergy.com>. 41

- [47] Hovhannest. Gas network simulation software. Image, 2011. License: CC BY-SA 3.0. URL: [https://upload.wikimedia.org/wikipedia/commons/e/ee/Gas\\_network\\_simulation\\_software.PNG](https://upload.wikimedia.org/wikipedia/commons/e/ee/Gas_network_simulation_software.PNG). 22
- [48] B. Howard, L. Parshall, J. Thompson, S. Hammer, J. Dickinson, and V. Modi. Spatial distribution of urban building energy consumption by end use. *Energy and Buildings*, 45:141–151, Feb 2012. doi:10.1016/j.enbuild.2011.10.061. 62
- [49] M. Howells, H. Rogner, N. Strachan, C. Heaps, H. Huntington, S. Kypreos, A. Hughes, S. Silveira, J. DeCarolis, M. Bazillian, and et al. OSeMOSYS: The open source energy modeling system. *Energy Policy*, 39(10):5850–5870, Oct 2011. doi:10.1016/j.enpol.2011.06.033. 41
- [50] M. Huber, J. Dorfner, and T. Hamacher. Electricity system optimization in the EUMENA region. Technical report, Technische Universität München, Jan 2012. doi:10.14459/2013md1171502. 39
- [51] M. Huber, A. Roger, and T. Hamacher. Optimizing long-term investments for a sustainable development of the ASEAN power system. *Energy*, 88:180–193, Aug 2015. doi:10.1016/j.energy.2015.04.065. 39
- [52] M. Huber, F. Sanger, and T. Hamacher. Coordinating smart homes in microgrids: A quantification of benefits. *IEEE PES ISGT Europe 2013*, Oct 2013. doi:10.1109/isgteurope.2013.6695357. 39
- [53] International Institute for Applied System Analysis (IIASA). Greenhouse gas initiative (GGI) scenario database, version 2.0. Web application. URL: <http://www.iiasa.ac.at/Research/GGI/DB/>. 38
- [54] K. A. Janker. *Aufbau und Bewertung einer für die Energiemodellierung verwendbaren Datenbasis an Zeitreihen erneuerbarer Erzeugung und sonstiger Daten*. Doctoral thesis, Technische Universität München, Jan 2015. URL: <http://mediatum.ub.tum.de/?id=1207265>. 61, 145
- [55] M. Jennings, D. Fisk, and N. Shah. Modelling and optimization of retrofitting residential energy systems at the urban scale. *Energy*, 64:220–233, Jan 2014. doi:10.1016/j.energy.2013.10.076. 40
- [56] J. Jonnes. *Empires Of Light: Edison, Tesla, Westinghouse, And The Race To Electrify The World*. Random House, 2003. ISBN 978-0375507397. URL: <http://www.jjonnes.com/b03-eol.html>. 22
- [57] W. Kaufmann. *Planung öffentlicher Elektrizitätsverteilungs-Systeme*. VDE-Verlag, 1995. ISBN: 978-3800721412. 15
- [58] J. Keirstead, M. Jennings, and A. Sivakumar. A review of urban energy system models: Approaches, challenges and opportunities. *Renewable and Sustainable Energy Reviews*, 16(6):3847–3866, Aug 2012. doi:10.1016/j.rser.2012.02.047. 40

- [59] C. Kennedy, S. Pincetl, and P. Bunje. The study of urban metabolism and its applications to urban planning and design. *Environmental Pollution*, 159(8-9):1965–1973, Aug 2011. doi:10.1016/j.envpol.2010.10.022. 11
- [60] H. Kleeberger, P. Tzscheuschler, and B. Geiger. Erstellen der Anwendungsbilanz 2009 und 2010 für den Sektor Gewerbe, Handel, Dienstleistungen (GHD). Technical report, Lehrstuhl für Energiewirtschaft und Anwendungstechnik, Technische Universität München, Sep 2011. URL: [http://www.ag-energiebilanzen.de/index.php?article\\_id=29&fileName=ife\\_geiger\\_anwendungsbilanz\\_ghd\\_sektor\\_2009\\_2.pdf](http://www.ag-energiebilanzen.de/index.php?article_id=29&fileName=ife_geiger_anwendungsbilanz_ghd_sektor_2009_2.pdf). 123
- [61] P. Konstantin. *Praxisbuch Energiewirtschaft*. Springer Science + Business Media, 3rd edition, 2013. doi:10.1007/978-3-642-37265-0. 60
- [62] U. Krien, G. Pleßmann, and C. Möller. oemof – Open Energy System Modelling Framework. v0.0.1. License: GPLv3. URL: <https://github.com/oemof>. 41
- [63] P. Krystallas and J. Dorfner. Expansion planning for district cooling networks using mathematical optimization. *EuroHeat & Power International*, 3:35–37, Aug 2014. URL: <http://www.ew-online.de/fachzeitschriften/euroheatpower/ehp-international/>. 129
- [64] P. Kuhn. *Iteratives Modell zur Optimierung von Speicherausbau und -betrieb in einem Stromsystem mit zunehmend fluktuierender Erzeugung*. Doctoral thesis, Technische Universität München, Munich, Oct 2012. URL: <https://mediatum.ub.tum.de/?id=1271192>. 27, 44
- [65] A. Ladak and R. B. Martinez. Automated derivation of high accuracy road centrelines thiesen polygons technique. In *ESRI International User Conference*, 1996. URL: <http://proceedings.esri.com/library/userconf/proc96/T0400/PAP370/P370.HTM>. 48
- [66] T. Lambert, P. Gilman, and P. Lienthal. *Integration of Alternative Sources of Energy*, chapter Micropower system modeling with HOMER. John Wiley & Sons, 2006. URL: <http://homerenergy.com/documents/MicropowerSystemModelingWithHOMER.pdf>. 41
- [67] U. Leuschner. Die deutsche Gasversorgung von den Anfängen bis 1998, 2009. URL: <http://www.udo-leuschner.de/pdf/gasversorgung.pdf>. 22
- [68] P. A. Longley, M. F. Goodchild, D. J. Maguire, and D. W. Rhind. *Geographic information systems & science*. Wiley, Hoboken, NJ, 3 edition, 2011. 47
- [69] R. Loulou, G. Goldstein, and K. Noble. Documentation for the MARKAL Family of Models – Part II: MARKAL-MACRO. URL: [http://www.etsap.org/MrkLDoc-II\\_MARKALMACRO.pdf](http://www.etsap.org/MrkLDoc-II_MARKALMACRO.pdf). 37
- [70] R. Loulou, G. Goldstein, and K. Noble. Documentation for the MARKAL Family of Models – Part I: Standard MARKAL. Manual, Oct 2004. URL: [http://www.etsap.org/MrkLDoc-I\\_StdMARKAL.pdf](http://www.etsap.org/MrkLDoc-I_StdMARKAL.pdf). 36

- [71] R. Loulou, U. Remne, A. Kanudia, A. Lehtila, and G. Goldstein. *Documentation for the TIMES Model - Part I: General description*. Energy Technology Systems Analysis Programme, Apr 2005. URL: <http://www.iea-etsap.org/web/Docs/TIMESDoc-Intro.pdf>. 37
- [72] R. Loulou, U. Remne, A. Kanudia, A. Lehtilä, and G. Goldstein. *Documentation for the TIMES Model - Part II: Reference manual*. Energy Technology Systems Analysis Programme, Apr 2005. URL: <http://www.iea-etsap.org/web/Docs/TIMESDoc-Details.pdf>. 33, 34, 35
- [73] D. MacKay. *Sustainable Energy – without the hot air*. UIT Cambridge, 2008. URL: <http://www.withouthotair.com/>. 12
- [74] Mainova AG. Zielnetzplanung: Frankfurt für die Zukunft gerüstet. Press release, Oct 2013. Retrieved on Sep 17, 2015. URL: <http://www.mainova-nachhaltigkeit.de/technik/medienuebergreifendezielnetzplanung.html>. 15
- [75] P. Mancarella. MES (multi-energy systems): An overview of concepts and evaluation models. *Energy*, 65(0):1–17, Feb 2014. doi:10.1016/j.energy.2013.10.041. 13, 14, 16
- [76] A. Martin, K. Klamroth, J. Lang, G. Leugering, A. Morsi, M. Oberlack, M. Ostrowski, and R. Rosen. *Mathematical Optimization of Water Networks*. Springer, 2012. 40
- [77] W. McKinney, J. Reback, and contributors. pandas, 2008. Python package providing fast, flexible, and expressive data structures. URL: <http://pandas.pydata.org/>. 57
- [78] D. Meadows, W. W. B. III, D. L. Meadows, R. F. Naill, J. Randers, and E. K. O. Zahn. *Dynamics of growth in a finite world*. Wright-Allen Press, 1974. URL: <http://www.worldcat.org/oclc/1089854>. 13
- [79] D. H. Meadows, D. L. Meadows, J. Randers, and W. W. B. III. *The Limits to Growth*. Universe Books, 1972. URL: <https://www.worldcat.org/oclc/307838>. 13
- [80] W. Medjroubi, C. Matke, and D. Kleinhans. SciGRID: an open source model of the European transmission network. License: GPLv3. URL: <http://www.scigrid.de/>. 42
- [81] S. Messner. *User's Guide for the Matrix Generator of Message II*. International Institute for Applied Systems Analysis (IIASA), 1984. IIASA Working Paper WP-84-071. URL: <http://webarchive.iiasa.ac.at/cgi-bin/pub/pubsrchKK?ID:WP84071>. 38
- [82] S. Messner and M. Strubegger. *User's Guide for MESSAGE III*. International Institute for Applied Systems Analysis (IIASA), Vienna, Jul 1995. IIASA Working Paper WP-95-069. URL: <http://webarchive.iiasa.ac.at/cgi-bin/pub/pubsrchKK?ID:WP95069>. 38
- [83] M. Migurski. Skeletron. Computes straight skeletons of simple polygons. URL: <https://github.com/migurski/Skeletron>. 48, 49
- [84] D. L. Morton. Reviewing the history of electric power and electrification. *Endeavour*, 26(2):60–63, 2002. doi:10.1016/S0160-9327(02)01422-9. 22

- [85] National Renewable Energy Laboratory. Typical meteorological year 3 (1991- 2005 update), 2005. URL: [http://rredc.nrel.gov/solar/old\\_data/nsrdb/1991-2005/tmy3/](http://rredc.nrel.gov/solar/old_data/nsrdb/1991-2005/tmy3/). 144
- [86] Y. Nesterov, A. Nemirovskii, and Y. Ye. *Interior-point polynomial algorithms in convex programming*, volume 13. SIAM, 1994. ISBN: 978-0-89871-319-0. doi:10.1137/1.9781611970791. 43
- [87] D. Newcomb. Renewed interest in district heating and cooling. *IEEE Power Eng. Rev.*, PER-5(4):7–10, Apr 1985. doi:10.1109/mper.1985.5528789. 20
- [88] R. Niemi, J. Mikkola, and P. Lund. Urban energy systems with smart multi-carrier energy networks and renewable energy generation. *Renewable Energy*, 48:524–536, 2012. doi:10.1016/j.renene.2012.05.017. 12
- [89] D. Oeding and B. R. Oswald. *Elektrische Kraftwerke und Netze*. Springer, Berlin, 6 edition, 2011. URL: <http://link.springer.com/book/10.1007/978-3-662-06960-8>. 59
- [90] openmod. Open energy modelling initiative, Sep 2014. An initiative of modellers from universities and research institutes across Europe. URL: <http://openmod-initiative.org/>. 41
- [91] OpenStreetMap contributors. OpenStreetMap database, accessed 2012–2015. Licensed under the Open Data Commons Open Database License (ODbL) by the OpenStreetMap Foundation (OSMF). URL: <http://www.openstreetmap.org/>. 22, 62, 122, 124, 136
- [92] OpenStreetMap contributors. OpenStreetMap map tiles, Accessed 2012–2015. Licensed under the Creative Commons Attribution-ShareAlike 2.0 license (CC BY-SA). URL: [http://wiki.openstreetmap.org/wiki/Tile\\_usage\\_policy](http://wiki.openstreetmap.org/wiki/Tile_usage_policy). 130, 132, 133, 134
- [93] M. Owen, editor. *ASHRAE Handbook*. American Society of Heating, Refrigerating, and Air-Conditioning Engineers, 2013. URL: <https://www.ashrae.org/resources--publications/handbook>. 59
- [94] S. Pfenninger. calliope - a multi-scale energy systems (MUSES) modeling framework. Release v0.3.4, Apr 2015. URL: <http://www.callio.pe/>, doi:10.5281/zenodo.17173. 41
- [95] S. Pfenninger, A. Hawkes, and J. Keirstead. Energy systems modeling for twenty-first century energy challenges. *Renewable and Sustainable Energy Reviews*, 33:74–86, May 2014. doi:10.1016/j.rser.2014.02.003. 36
- [96] S. Pfenninger and J. Keirstead. Renewables, nuclear, or fossil fuels? Scenarios for Great Britain’s power system considering costs, emissions and energy security. *Applied Energy*, 152:83–93, Aug 2015. doi:10.1016/j.apenergy.2015.04.102. 41
- [97] H. Pálsson. *Methods for planning and operating decentralized combined heat and power plants*. Department of Energy, Technical University of Denmark, 2000. URL: <http://orbit.dtu.dk/files/51481519/Halldor.PDF>. 61



- [98] A. Ramaswami, A. Chavez, J. Ewing-Thiel, and K. E. Reeve. Two approaches to greenhouse gas emissions foot-printing at the city scale. *Environ. Sci. Technol.*, 45(10):4205–4206, May 2011. doi:10.1021/es201166n. 40
- [99] D. Reiter, F. Botzenhart, P. Mühlich, T. Hamacher, and A. Reuter. Developing a roadmap for the future energy infrastructure in Salzburg. In *10th IAEE European Conference*, Vienna, Sep 2009. 15
- [100] J. C. Richstein, E. J. Chappin, and L. J. de Vries. Cross-border electricity market effects due to price caps in an emission trading system: An agent-based approach. *Energy Policy*, 71:139–158, Aug 2014. doi:10.1016/j.enpol.2014.03.037. 41
- [101] S. Richter. *Entwicklung einer Methode zur integralen Beschreibung und Optimierung urbaner Energiesysteme*. Doctoral thesis, Universität Augsburg, Jan 2004. URL: <http://d-nb.info/973199415/34>. 15, 38, 39, 65
- [102] K. Schaber. *Integration of Variable Renewable Energies in the European power system: a model-based analysis of transmission grid extensions and energy sector coupling*. Doctoral thesis, Technische Universität München, Dec 2013. URL: <https://mediatum.ub.tum.de/?id=1163646>. 15, 39
- [103] K. Schaber, F. Steinke, and T. Hamacher. Transmission grid extensions for the integration of variable renewable energies in Europe: Who benefits where? *Energy Policy*, 43:123–135, Apr 2012. doi:10.1016/j.enpol.2011.12.040. 15, 39
- [104] K. Schaber, F. Steinke, P. Mühlich, and T. Hamacher. Parametric study of variable renewable energy integration in Europe: Advantages and costs of transmission grid extensions. *Energy Policy*, 42:498–508, Mar 2012. doi:10.1016/j.enpol.2011.12.016. 15, 39
- [105] K. Schmidt. Göbekli Tepe – the stone age sanctuaries: New results of ongoing excavations with a special focus on sculptures and high reliefs. *Documenta Praehistorica*, XXXVII:239–256, Dec 2010. doi:10.4312/dp.37.21. 11
- [106] A. Schrijver. *Theory of linear and integer programming*. John Wiley & Sons, 1998. ISBN 978-0-471-98232-6. 44
- [107] K. Schönsteiner, M. Wagner, S. Dierkes, K. Münch, M. Hans, T. Hamacher, A. Moser, and M. Fiedeldej. Veränderte Anforderungen an städtische Energieinfrastrukturen durch die Energiewende. *Energiewirtschaftliche Tagesfragen*, 64(10):75–78, Oct 2014. URL: <http://www.et-energie-online.de/AktuellesHeft/Inhalt/tabid/69/Year/2014/Month/10/NewsModule/416/201410.aspx>. 119
- [108] A. J. Seebregts. Energy/environmental modeling with the MARKAL family of models. Presentation, Sep 2001. URL: <http://www.gams.com/presentations/or01/marcal.pdf>. 37
- [109] H. D. Sherali and P. J. Driscoll. Evolution and state-of-the-art in integer programming. *Journal of Computational and Applied Mathematics*, 124(1-2):319–340, Dec 2000. doi:10.1016/s0377-0427(00)00431-3. 44

- [110] A. H. Simmons. *The Neolithic Revolution in the Near East – Transforming the Human Landscape*. The University of Arizona Press, 2007. URL: <http://www.uapress.arizona.edu/Books/bid1808.htm>. 11
- [111] J. Snow. Map showing cases of cholera in the London epidemics of 1854. Image, 1854. License: Public domain. URL: <https://commons.wikimedia.org/wiki/File:Snow-cholera-map.jpg>. 61
- [112] Stadtwerke München. Map of district heating network extension in Munich Pasing-Laim and Westkreuz. Accessed Feb 28, 2013. URL: <http://www.swm.de/dms/swm/dokumente/m-fernwaerme/plan-fernwarmausbau-west.pdf>. 122, 124
- [113] Stadtwerke München. Homepage, section M-Fernwärme, Accessed Feb 2013. URL: <http://www.swm.de/>. 122
- [114] N. Stern. Stern review on the economics of climate change. Technical report, HM Treasury, 2006. URL: <http://www.webcitation.org/5nCeyEYJr>. 60
- [115] J. Söderman. Optimisation of structure and operation of district cooling networks in urban regions. *Applied Thermal Engineering*, 27:2665–2676, May 2007. doi:10.1016/j.applthermaleng.2007.05.004. 40
- [116] Technische Universität München. EEBatt – Dezentrale stationäre Batteriespeicher zur effizienten Nutzung erneuerbarer Energien und Unterstützung der Netzstabilität. Project website, Jan 2013 – Dec 2016. Retrieved 17 Sep 2015. URL: <http://www.eebatt.tum.de>. 16, 135
- [117] H.-P. Tietz and T. Hühner. Zukunftsfähige Infrastruktur und Raumentwicklung: Handlungserfordernisse für Ver- und Entsorgungssysteme. Forschungs- und Sitzungsberichte der ARL 235, Akademie für Raumforschung und Landesplanung (ARL) – Leibniz-Forum für Raumwissenschaften, Hannover, 2011. URL: <http://hdl.handle.net/10419/60112>. 12
- [118] UNO. World urbanization prospects, the 2014 revision. United Nations, 2014. URL: <http://esa.un.org/unpd/wup/>. 12
- [119] UNO, Department of Public Information. UN finds world’s population is increasingly urban with more than half living in urban areas today and another 2.5 billion expected by 2050. Press release, Jul 2014. URL: [http://esa.un.org/unpd/wup/PressRelease/WUP2014\\_PressRelease.pdf](http://esa.un.org/unpd/wup/PressRelease/WUP2014_PressRelease.pdf). 12
- [120] U.S. Department of Energy (DOE). Commercial and residential hourly load profiles for all TMY3 locations in the United States. URL: <http://en.openei.org/datasets/dataset/commercial-and-residential-hourly-load-profiles-for-all-tmy3-locations-in-the-united-states>. 144
- [121] J. Utoft and H. Thomsen. The history of gas. URL: <http://www.gashistory.org/Files/gashistoryWGC06.pdf>. 22



- [122] P. Voll. *Automated Optimization-Based Synthesis of Distributed Energy Supply Systems*. Doctoral thesis, RWTH Aachen, Jun 2013. URL: <http://publications.rwth-aachen.de/record/228954>. 40
- [123] A. von Meier. *Electric Power Systems – A Conceptual Introduction*. Wiley-IEEE Press, Aug 2006. 328 pages. URL: <http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0471178594,miniSiteCd-IEEE2.html>. 14
- [124] A. Voß, C. Schlenzig, and A. Reuter. MESAP-III: A tool for energy planning and environmental management. history and new developments. Technical report, Institut für Energiewirtschaft und Rationelle Energieanwendung (IER), Universität Stuttgart, Sep 1994. URL: <http://elib.uni-stuttgart.de/opus/volltexte/2013/8529/pdf/vos280.pdf>. 38
- [125] M. Welsch, M. Howells, M. Bazilian, J. DeCarolis, S. Hermann, and H. Rogner. Modelling elements of smart grids – enhancing the OSeMOSYS (open source energy modelling system) code. *Energy*, 46(1):337–350, Oct 2012. doi:10.1016/j.energy.2012.08.017. 41
- [126] C. Wheeler. Sewers – ancient and modern; compiled from official sources of the sewerage system of the city of Auburn, N.Y. In *a paper read before the Cayuga County Historical Society, December 14, 1886*, published 1887. URL: <https://archive.org/details/sewersancientmod02whee>. 11
- [127] H. Wickham et al. The split-apply-combine strategy for data analysis. *Journal of Statistical Software*, 40(1):1–29, 2011. doi:10.18637/jss.v040.i01. 56
- [128] F. Wiese. *renpass: Renewable Energy Pathways Simulation System*. Doctoral thesis, University of Flensburg, Apr 2015. Subtitle: Open Source as an approach to meet challenges in energy modeling. URL: [http://renpass.eu/files/public-docs/renpass\\_publications/Dissertation\\_Frauke\\_Wiese\\_April2015\\_digitalversion.pdf](http://renpass.eu/files/public-docs/renpass_publications/Dissertation_Frauke_Wiese_April2015_digitalversion.pdf). 42
- [129] M. Wolter. *Infrastrukturmärkte: eine betriebswirtschaftliche Analyse*. Doctoral thesis, Universität München, 1997. ISBN: 978-3631319703. 20, 21
- [130] Q. Zhang, Y. Ban, J. Liu, and Y. Hu. Simulation and analysis of urban growth scenarios for the greater shanghai area, China. *Computers, Environment and Urban Systems*, 35(2):126–139, Mar 2011. doi:10.1016/j.compenvurbsys.2010.12.002. 13